Numerical analysis of unsteady viscoelastic flow

Baaijens, F.P.T.

Published in:
Computer Methods in Applied Mechanics and Engineering

Published: 01/01/1992

Document Version
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Numerical analysis of unsteady viscoelastic flow

Frank P.T. Baaijens
Philips Research Laboratories, P.O. Box 80000, 5600 JA Eindhoven, The Netherlands
and
Eindhoven University of Technology, Faculty of Mechanical Engineering, Section of Engineering Fundamentals, P.O. Box 513, 5600 MB Eindhoven, The Netherlands

Received 23 November 1990

A numerical method is presented to analyse unsteady viscoelastic flow. Unsteady calculations may be used to find steady-state solutions as a limiting case. The solution scheme employs a mixed method with a decoupled Picard iteration solver. The elastic extra stress tensor is integrated with a space-time finite element method: the time discontinuous/streamline upwind Petrov-Galerkin (TD/SUPG) method. The Jeffreys, Phan-Thien/Tanner and Leonov models are used. The method is consistent, contrary to previously defined techniques.

1. Introduction

Many algorithms have been proposed to solve the viscoelastic flow of various model fluids through the so-called $4-1$ contraction, i.e. [1]. One of the most successful methods appears to be the streamline-upwind (SU) algorithm of Marchal and Crochet [2]. This algorithm converges for very high values of the Deborah number (De). However, it is subject to criticism (see [3]) because the method is inconsistent. The exact solution does not satisfy the weak form.

The purpose of this paper is to propose an algorithm that is consistent. It employs a space–time finite element method, a technique now widely used in other areas of computational fluid dynamics (CFD). Particularly, the time discontinuous/Galerkin least-squares (TD/GLS) method is very popular in solving the compressible Navier–Stokes equations, i.e. [4, 5]. Viscoelastic flows share many of the features of these problems, and it may be expected that the TD/GLS method is a good candidate to solve viscoelastic flow problems. However, it was found that the TD/GLS form could not be applied fully. A slight modification must be performed: rather than using a least-squares form, a streamline upwind Petrov–Galerkin (SUPG) form needs to be used. It must be noticed though that consistency is maintained.

The structure of the paper is as follows. First the problem is defined accurately, then a rough outline of the solution schemes is presented. Thereafter, the formal definitions of the problems mentioned in the outline are specified in detail. Next, the discretization employed is defined and some computational aspects are discussed. Finally, the performance of the algorithms is investigated for the $4-1$ contraction problem.

2. Problem definition

Let $\Omega$ be a spatial domain in $\mathbb{R}^2$ with a smooth boundary $\Gamma$. $I$ represents the open time interval $[0, T]$. The issue is to solve the unsteady viscoelastic plane flow with three constitutive equations: the Jeffreys, Phan–Thien/Tanner (PTT) and the Leonov model.
The formal definition of the problem is given by

(P) Given \( v^0 : \Gamma_v \rightarrow \mathbb{R}^2 \) and \( \tau^0 : \Gamma_r \rightarrow \mathbb{R}^4 \), find the plane velocity field \( v(x, t) : \Omega \times I \rightarrow \mathbb{R}^2 \) and the pressure field \( p(x, t) : \Omega \times I \rightarrow \mathbb{R} \) for all \( (x, t) \in \Omega \times I \), such that

\[
\nabla \cdot (-pI + \dot{\tau} + \tau) = 0, \tag{1}
\]

\[
\nabla \cdot v = 0, \tag{2}
\]

with

(A) Jeffreys model:

\[
\dot{\tau} = 2(\eta_0 + \eta_1)D, \tag{3}
\]

\[
\dot{\tau} + \frac{1}{\theta} \tau = -2\eta_0 D, \tag{4}
\]

\[
\dot{\tau} = \dot{\tau} - L \cdot \tau - \tau \cdot L^t. \tag{5}
\]

(B) PTT model:

\[
\dot{\tau} = 2\eta_0 D, \tag{6}
\]

\[
\dot{\tau} + \left( \frac{1}{\theta} + \frac{\varepsilon}{\eta_1} \text{tr}(\tau) \right) \tau = 2 \frac{\eta_1}{\theta} D. \tag{7}
\]

(C) Leonov model:

\[
\dot{\tau} = 2\eta_0 D, \quad \tau = \frac{\eta_1}{\theta} (B - I), \tag{8}
\]

\[
\dot{B} = L \cdot B + B \cdot L^t - \frac{1}{2\theta} (B^2 - I), \tag{9}
\]

where \( L = (\nabla v)^t \) and \( D = \frac{1}{2} (L + L^t) \), while the following boundary conditions are specified on \( \Gamma \):

\[
v(x, t) = v^0(x, t) \quad \text{on} \ \Gamma_v, \tag{10}
\]

\[
\tau(x, t) = \tau^0(x, t) \quad \text{on} \ \Gamma_r, \tag{11}
\]

with \( \Gamma = \Gamma_v \cup \Gamma_r \).

3. Outline of the decoupled solution scheme, Algorithm 1

Problem P consists of a set of coupled partial differential equations. In this paper a mixed formulation is used, where the momentum, continuity and constitutive equations are cast in a weak form. In this section an algorithm is roughly outlined that works quite well for moderate Deborah numbers in the case of the Jeffreys model when cast in the form of eq. (4), and gives good results for the PTT model and the Leonov model at high Deborah numbers.

In a space–time finite element method, not only the spatial variables, but also time are
discretized with a finite element scheme. In the technique used in this paper, time is discretized with piecewise continuous functions. Let the time domain $I$ be partitioned in $n$ time slabs: $0 < t_0 < t_1 < \cdots < t_n = T$. One time slab is denoted by $I_n = [t_n^-, t_n^+]$. At the interface between two time slabs, the solution is, in general, discontinuous. Therefore, one has to distinguish between the state of a variable at $t_n^-$ and $t_n^+$.

Assume the solution to be known at $t_n^-$. Then the solution at $t_n^+$ is found by iteratively solving a sequence of weak problems. Essentially, the algorithm is a decoupled Picard iteration solution scheme. The algorithm applies to all three models investigated, but at first only reference is made to the Jeffreys model. The modifications for the PTT and Leonov model are indicated later. First, it is assumed that the extra stress tensor $\tau$ is known. Then an estimate for the current velocity and pressure field is obtained by solving $\text{PW}_1$ (weak form of the momentum (1) and continuity equation (2)). Next the discontinuous velocity gradient as obtained from differentiation of the velocity field is projected onto the space of continuous velocity gradient tensors (step 2: problem $\text{PW}_2$). Finally, at the end of each iteration, the extra stress tensor is calculated from problem $\text{PAW}_3$ (weak form of the Jeffreys model). This process is repeated until convergence of the solution is obtained. The precise definition of the weak problems $\text{PW}_1$, $\text{PW}_2$ and $\text{PAW}_3$ is given later. In the case of the PTT or Leonov model, $\text{PAW}_3$ is replaced by $\text{PBW}_3$ and $\text{PCW}_3$, respectively.

Denote by $[\mathcal{C}^{-1}(\Omega)]^d$ the space of piecewise continuous functions on $\Omega$, and by $[\mathcal{C}^0(\Omega)]^d$ the space of continuous functions on $\Omega$, both of dimension $d$, e.g.

$$[\mathcal{C}^{-1}(\Omega)]^d = \{f : \Omega \rightarrow \mathbb{R}^d \mid f \text{ piecewise continuous}\}.$$  \hspace{1cm} (12)

The algorithm can be summarized as follows.

**ALGORITHM A1**

**Step 0.** Initiation:

$$i = 1, \quad \tau^i = \tau(x, t_n^-).$$

**Step 1.** Equilibrium and continuity iteration:

$$\mathbf{v}^i, p^i, \tau^i \mapsto \text{PW}_1 \mapsto \mathbf{v}^{i+1}, p^{i+1}.$$ 

**Step 2.** Projection of the (discontinuous) velocity gradient:

$$L(v^{i+1}) \in [\mathcal{C}^{-1}]^d \mapsto \text{PW}_2 \mapsto \hat{L}^{i+1} \in [\mathcal{C}^0]^d.$$ 

**Step 3.** Solution of extra stress tensor:

$$\mathbf{v}^{i+1}, \hat{L}^{i+1} \mapsto \text{PAW}_3 \mapsto \tau^{i+1}.$$ 

**Step 4.** Continuation:

$$i \leftarrow i + 1.$$

Repeat from Step 1 until convergence.

The symbol $\mapsto$ signifies that the preceding or trailing variable is either used as an input variable for a certain problem or is an output variable.
4. Outline of the decoupled solution scheme Leonov model

In the case of the Leonov model another algorithm has been experimented with. The crucial difference between the algorithm proposed in this section and Algorithm A1 is that an operator splitting technique is employed to determine the extra stress tensor $\tau$. For the Leonov model, $\tau$ depends solely on $B$, defined by (9). The operator splitting method employed is as follows.

First the formal definition of the material time derivative $\dot{B}$ is exploited:

$$\dot{B} = \lim_{\Delta t \to 0} \frac{B(x, t) - B(p(x), t - \Delta t)}{\Delta t}, \quad (13)$$

where $p$ denotes the position at time $t - \Delta t$ of the particle currently located at $x$. Define $B_p(x, t) = B(p(x), t - \Delta t)$. The field $B_p(x, t)$ can be found by solving the transport problem:

Find $C(x, t) \in \Omega \times I_n$ such that

$$\dot{C} = \left( \frac{\partial}{\partial t} + u_j \frac{\partial}{\partial x_j} \right) C = 0 , \quad (14)$$

$$C(x, t_n) = B(x, t_n^-) .$$

Hence $B_p(x, t_{n+1}^-) = C(x, t_{n+1}^-)$.

Secondly, according to (13), on $I_n$, $\dot{B}$ can be approximated by

$$\dot{B} \approx \frac{B(x, t_{n+1}) - B_p(x, t_{n+1})}{\Delta t} . \quad (15)$$

So, $B(x, t_{n+1})$ is found by solving (19), given in Step 4 below.

Finally, the continuous strain field thus obtained, is projected on a discontinuous approximation (of lower order than the original field).

**ALGORITHM A2**

**Step 0. Initiation:**

$$i = 1 , \quad \tau^i = \tau(x, t_n^-) , \quad B^i = B(x, t_n^-) .$$

**Step 1. Equilibrium and continuity iteration:**

$$\nu^i , p^i , \tau^i \mapsto PW_1 \mapsto \nu^{i+1} , p^{i+1} . \quad (16)$$

**Step 2. Projection of the (discontinuous) velocity gradient:**

$$L(\nu^{i+1}) \in [\mathbb{G}^{-1}]^4 \mapsto PW_2 \mapsto \dot{L}^{i+1} \in [\mathbb{G}^0]^4 . \quad (17)$$

**Step 3. Find $B_p(x, t_{n+1}^-) = B(p(x, t_n^-))$ by solving $\dot{C} = 0$ on $I_n$:**

$$B(x, t_n^-) , \nu^{i+1} \mapsto PCW_{32} \mapsto B_p . \quad (18)$$
Step 4. Calculate $B_{n+1} = B(x, t_{n+1})$ at nodes:

$$\frac{B_{n+1} - B_p}{\Delta t} = \hat{L}^{i+1} \cdot B_{n+1} + B_{n+1} \cdot (\hat{L}^{i+1})^T - \frac{1}{2\theta} (B_{n+1}^2 - I).$$

(19)

Step 5. Projection of $B_{n+1}$ on discontinuous 'element' field:

$$B_{n+1} \in [\mathcal{C}^0]^4 \mapsto \text{PW}_{4} \mapsto \tilde{B}_{n+1} \in [\mathcal{C}^{-1}]^4 \mapsto \tau^{i+1}.$$  

(20)

Step 6. Continuation:

$$i \leftarrow i + 1.$$

Repeat from Step 1 until convergence.

REMARK. When solving (19), explicit use is made of the incompressibility constraint $\det(B) = 1$. If this constraint is not enforced, $\det(B)$ slowly drifts away from 1 and the solution becomes unstable.

5. Definition of the weak problems

5.1. Definition of problem PW$_1$

The problem PW$_1$ is the classical weighted residual formulation of the momentum equation (1) and continuity equation (2) for an incompressible viscous fluid with a given extra stress tensor $\tau$. A penalty function method is employed with discontinuous approximation of the pressure field that allows the elimination of the pressure variables on the element level.

The trial solutions and weighting functions are approximated by $k$th-order interpolation polynomials, $\mathcal{P}_k$. Define the space of velocity trial solutions as

$$\mathcal{V} = \{v \mid v \in [\mathcal{C}^0(\Omega)]^2, v|_{\Gamma_o} \in [\mathcal{P}_k(\Omega^e)]^2, v = v^0 \text{ on } \Gamma_v\},$$

(21)

the space of velocity weighting functions as

$$\mathcal{W} = \{w \mid w \in [\mathcal{C}^0(\Omega)]^2, w|_{\Gamma_o} \in [\mathcal{P}_k(\Omega^e)]^2, w = 0 \text{ on } \Gamma_v\},$$

(22)

and the space of pressure trial solutions and weighting functions as

$$\mathcal{Z} = \{q \mid q \in \mathcal{C}^{-1}(\Omega), q|_{\Gamma_o} \in \mathcal{P}_k(\Omega^e)\}.$$  

(23)

By omitting reference to time- and iteration level, PW$_1$ can be defined as:

(PW$_1$) Given $\tau(x, t)$, find $v(x, t) \in \mathcal{V}$ and $p(x, t) \in \mathcal{Z}$ such that for all $(w, q) \in (\mathcal{W}, \mathcal{Z})$

$$\int_{\Omega} (\nabla w)^T : (-pI + \hat{\gamma}(v) + \tau(v)) \, d\Omega = 0,$$

(24)

$$\int_{\Omega} q \left( \frac{1}{\delta} p + \nabla \cdot v \right) \, d\Omega = 0.$$  

(25)

The penalty parameter is denoted by $\delta$. 
5.2. Definition of problem PW₂

The velocity gradient \( \dot{L} \) as calculated from the velocity field is discontinuous across element borders, e.g. \( \dot{L} \in [\mathcal{C}^{-1}]^4 \). It is made continuous by projection onto \([\mathcal{C}^0]^4\). Along certain parts of the boundary, say \( \Gamma_3 \), \( \dot{L} \) may be known a priori: \( \dot{L}(x, t) = \dot{L}_0(x, t) \) on \( \Gamma_3 \subseteq \Gamma \).

Define the space of trial solutions as
\[
\mathcal{L} = \{ \dot{L} | \dot{L} \in [\mathcal{C}^0(\Omega)]^4, \dot{L}|_{\alpha e} \in [\mathcal{P}_k(\Omega^e)]^4, \dot{L} = \dot{L}_0 \text{ on } \Gamma_3 \} \tag{26}
\]
and weighting functions as
\[
\mathcal{K} = \{ K | K \in [\mathcal{C}^0(\Omega)]^4, K|_{\alpha e} \in [\mathcal{P}_k(\Omega^e)]^4, K = 0 \text{ on } \Gamma_3 \} . \tag{27}
\]

\( \text{(PW}_2 \text{)} \) Given \( \dot{L} \in [\mathcal{C}^{-1}]^4 \), find \( \dot{L} \in \mathcal{L} \) such that for all \( K \in \mathcal{K} \),
\[
\int_{\Omega} K : (\dot{L} - \dot{L}) \, d\Omega = 0 . \tag{28}
\]
Furthermore, by adding \( \alpha I \) to \( \dot{L} \) the constraint \( \text{tr}(\dot{L} + \alpha I) = 0 \) is enforced at the nodes.

5.3. Definition of problem PAW₃

The time-discontinuous/streamline upwind Petrov–Galerkin (TD/SUPG) method has proven to be very successful for the solution of advective diffusion equations (see [5]). The Jeffreys equation (4) contains an advective part due to the presence of the material time derivative of the extra stress tensor \( \tau \). The central idea of this paper is to apply the TD/SUPG technique to this equation.

The Jeffreys model is written in the following operator form:
\[
\mathcal{L}_d = \mathcal{L}_d d . \tag{29}
\]
Where the following definitions have been used:
\[
\tau^t = \begin{bmatrix} \tau_{11} & \tau_{22} & \tau_{12} \end{bmatrix} , \tag{30}
\]
\[
d^t = \begin{bmatrix} D_{11} & D_{22} & D_{12} \end{bmatrix} , \tag{31}
\]
\[
\dot{\mathcal{P}} = \left( \frac{\partial}{\partial t} + \nu_x \frac{\partial}{\partial x_t} \right) \mathcal{I} , \tag{32}
\]
\[
\mathcal{L} = \dot{\mathcal{P}} + \left( \mathcal{C} + \frac{1}{\theta} \mathcal{I} \right) , \tag{33}
\]
\[
\mathcal{L}_d = -2\eta_t (\dot{\mathcal{P}} + \mathcal{C}) , \tag{34}
\]
\[
\mathcal{C} = \begin{bmatrix} -2L_{11} & 0 & -2L_{12} \\ 0 & -2L_{22} & -2L_{21} \\ -L_{21} & -L_{12} & -L_{11} - L_{22} \end{bmatrix} . \tag{35}
\]

In a space–time finite element method, the domain \( \Omega \times I \) is divided into space–time slabs, defined by
\[ Q_n = \Omega \times I_n \]  

with boundary

\[ P_n = \Gamma \times I_n. \]

A space–time element can be defined as

\[ Q_e^\varepsilon = \Omega^\varepsilon \times I_n, \quad e = 1, \ldots, n_{el}. \]

The trial solutions and weighting functions are approximated by \( k \)-th order interpolation polynomials \( \mathcal{P}_k(Q_n) \). These are chosen \( C^0 \) continuous within each space–time slab, but are discontinuous across the interfaces of each time slab. Now, define the space of stress trial solutions as

\[ \mathcal{S}_n = \{ \sigma \mid \sigma \in [C^0(Q_n)]^3, \sigma|_{Q_e^\varepsilon} \in [\mathcal{P}_k(Q_e^\varepsilon)]^3, \sigma = \tau^0 \text{ on } P_n \} \]

and the space of stress weighting functions as

\[ \mathcal{V}_n = \{ w \mid w \in [C^0(Q_n)]^3, \ w|_{Q_e^\varepsilon} \in [\mathcal{P}_k(Q_e^\varepsilon)]^3, \ w = 0 \text{ on } P_n \}. \]

Within the context of the finite element theory, application of the TD/SUPG method to the Jeffreys model results in the following weak form: (PAW) Given \( L(x, t) \) and \( v \), find \( \tau(x, t) \in \mathcal{S}_n \) such that for all \( w(x, t) \in \mathcal{V}_n \)

\[
\int_{Q_n} w^i \mathcal{L}_T \, dQ_n + \sum_{e=1}^{n_{el}} \int_{Q_e^\varepsilon} (\mathcal{P}_w)^i \mathcal{L}_T \, dQ_n \\
Galiner \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad
\]

\[
\lambda = \lambda I, \quad \lambda = \frac{\Delta t}{2 + \left[ \frac{\partial \xi_j}{\partial x_i} \frac{\partial \xi_i}{\partial x_j} v_k^u \right]^{1/2}}, \quad i, j, k = 1, 2, \]

\[
\lambda_D = 2 \frac{||\mathcal{L}_T - \mathcal{L}_D||_A^2}{||\mathcal{L}_E||^2} ,
\]

\[
\mathcal{L}_E = \begin{bmatrix} L & \frac{\partial}{\partial \xi_1} I & \frac{\partial}{\partial \xi_1} I & \frac{\partial}{\partial \xi_2} I \end{bmatrix}.
\]
REMARKS.
(1) Notice that the weak form is exactly satisfied when the exact solution is substituted, hence the method is consistent.
(2) Numerical experiments showed that replacement of \( \mathcal{L} \) by \( \mathcal{L}_d \), which would constitute a time discontinuous/Galerkin least-squares method, does not give reliable results. This might be due to the non-symmetry of the matrix \( \mathcal{C} \).
(3) The definition of \( \lambda_0 \) renders PAW\(_3\) non-linear even though \( \mathcal{L} \) is linear.
(4) The choices (42) and (43) may be non-optimal. Recent results of Franca and Stenberg [6] indicate that these choices may have to be revised.

5.4. Definition of problem PBW\(_3\)
In analogy with the previous section, the PTT model is written in the following operator form:
\[
\mathcal{L}d = 2 \frac{\eta}{\theta} d .
\]  
(45)
Where the operator \( \mathcal{L} \) is now defined by
\[
\mathcal{L} = \mathcal{L}_d + \mathcal{C} + \left( \frac{1}{\theta} + \frac{\epsilon}{\eta_1} \text{tr}(\tau) \right) I .
\]  
(46)
Hence, problem PBW\(_3\) has exactly the same format as problem PAW\(_3\) if the above definition of the operator \( \mathcal{L} \) is applied and \( \mathcal{L}_d = I \) is used.

5.5. Definition of problem PCW\(_{31}\)
Define
\[
b' = \begin{bmatrix} B_{11} & B_{22} & B_{12} \end{bmatrix} .
\]  
(47)
In analogy with the previous section, the Leonov model is written in the following operator form:
\[
\mathcal{L}b = \mathcal{L}_d d ,
\]  
(48)
where the operator \( \mathcal{L} \) is now defined by
\[
\mathcal{L} = \mathcal{L}_d + \mathcal{C} .
\]  
(49)
\( \mathcal{L}_d = I \) and \( d \) is defined by
\[
d = -\frac{1}{2\theta} \begin{bmatrix} B_{11}^2 + B_{12}^2 - 1 \\ B_{12}^2 + B_{22}^2 - 1 \\ B_{11}B_{12} + B_{12}B_{22} \end{bmatrix} .
\]  
(50)
Hence, problem PCW\(_{31}\) has exactly the same format as problem PAW\(_3\) if the above definitions are applied.

REMARK. Once \( B \) is calculated, the constraint \( \det(B) = 1 \) is enforced pointwise at the nodes, to avoid \( B \) slowly drifting away from this constraint.
5.6. Definition of problem PCW$_{32}$

The transport of $B$ can be solved with exactly the same technique as used in problem PAW$_3$. The problem to be solved is:

(PCW$_{32}$) Given $v(x, t)$, find $b(x, t) \in \mathcal{S}_n$ such that for all $w \in \mathcal{V}_n$,

$$
\int_{Q_n} w^T \mathcal{C}_2 b \, dQ_n + \sum_{e=1}^{n_e} \left[ \int_{Q_e} (\mathcal{C}_2 w)^T \Lambda \mathcal{C}_2 b \, dQ_e \right] \\
+ \sum_{e=1}^{n_e} \int_{Q_e} \lambda_D (\nabla \xi w)^T (\nabla \xi b) \, dQ_e + \int_{\Omega} [w(t^-_n)'(b(t^-_n) - b(t^-_n))] \, dQ
$$

Galerkin  Least-squares

Discontinuity capturing  Jump condition

\[ = 0 . \tag{51} \]

REMARKS.

(1) In this case, a least-squares form is used.

(2) The space $\mathcal{S}_n$ must be adjusted to accommodate the difference in boundary conditions between PAW$_3$ and PCW$_{32}$.

5.7. Definition of problem PW$_4$

Define the space of trial solutions and weighting functions by

$$
\mathcal{B} = \{ B \mid B \in [\mathcal{C}^{-1}(\Omega)]^4, B|_{\partial \Omega} \in [\mathcal{P}_h(\Omega')]^4 \} . \tag{52}
$$

The strain field $B \in [\mathcal{C}^0]^4$ is projected on a discontinuous field by the following weak form that is applied for each element.

(PW$_2$) Given $B \in [\mathcal{C}^0]^4$, find $\tilde{B} \in \mathcal{B}$ such that for all $K \in \mathcal{B}$,

$$
\int_{\Omega} K : (B - \tilde{B}) \, d\Omega = 0 . \tag{53}
$$

6. Discretisation

For the discretisation of $(v, p)$ in PW$_1$ the $P_2^+ - P_1$ Crouzeix–Raviart element is used. The components of $\bar{L}$ are discretized with quadratic triangular $P_2$ elements. The components of the extra stress tensor $\tau$ are discretized in space with quadratic triangular elements that are degenerated 8-node quadrilaterals by coalescing three points on one side (see [7]), and with a discontinuous linear field in time. To apply the operator $\mathcal{L}_d$ in PAW$_3$, $D$ was discretized in space with the same interpolation functions as $\tau$, but in time a piecewise constant approximation was used. For problem PBW$_3$, the same interpolations were used as for PAW$_3$, with the exception that $D$ is not interpolated separately. The strain field $B_p$ of problem PCW$_{32}$ was equally interpolated as $\tau$, both in space and time. The discontinuous projection $\tilde{B}$ of $B$ was interpolated with linear triangular $P_1$ elements. This is summarized in Table 1.
Table 1

<table>
<thead>
<tr>
<th>Problem</th>
<th>Variable(s)</th>
<th>Space</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>PW₁</td>
<td>(v; p)</td>
<td>P₂ ⁺ – P₁</td>
<td></td>
</tr>
<tr>
<td>PW₂</td>
<td>L</td>
<td>P₂</td>
<td></td>
</tr>
<tr>
<td>PAW₃</td>
<td>τ, D</td>
<td>P₂</td>
<td>P₀</td>
</tr>
<tr>
<td>PBW₃</td>
<td>τ</td>
<td>P₂</td>
<td>P₁</td>
</tr>
<tr>
<td>PCW₃₁</td>
<td>β</td>
<td>P₂</td>
<td>P₁</td>
</tr>
<tr>
<td>PCW₃₂</td>
<td>β</td>
<td>P₂</td>
<td>P₁</td>
</tr>
<tr>
<td>PW₄</td>
<td></td>
<td></td>
<td>P₁</td>
</tr>
</tbody>
</table>

7. Computational aspects

The parameter $\lambda_0$ in the discontinuity capturing operator is scaled with the $\Lambda$ norm of the residual $\mathcal{L}_τ - \mathcal{L}_Ω$. Therefore problem PAW₃ is non-linear even though the strong form is linear (at a given velocity field $v(x, t)$). This non-linearity is circumvented by evaluation of $\lambda_0$ at time $t^*$ when applied in time slab $I_0$. This is also done when solving PBW₃, PCW₃₁ and PCW₃₂.

However, the operators $\mathcal{L}$ of problems PBW₃ and PCW₃₁ are non-linear as well, hence PBW₃ and PCW₃₁ must be solved iteratively. At each iteration of Algorithm A1, PBW₃ and PCW₃₁ are solved with a modified Newton–Raphson iteration process: the tangent stiffness matrix is evaluated only once every time step.

8. Sample problem: 4–1 contraction

The performance of the algorithms is demonstrated for the 4–1 contraction problem. The geometry and the mesh used are depicted in Fig. 1.
Table 2

<table>
<thead>
<tr>
<th></th>
<th>Jeffreys</th>
<th>PTT</th>
<th>Leonov</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$</td>
<td>2.5, 5</td>
<td>2.5, 5, 10</td>
<td>2.5, 5, 10</td>
</tr>
<tr>
<td>$\eta_0$</td>
<td>0.11</td>
<td>0.11</td>
<td>0.11</td>
</tr>
<tr>
<td>$\eta_1$</td>
<td>0.89</td>
<td>0.89</td>
<td>0.89</td>
</tr>
<tr>
<td>$\dot{\gamma}_w$</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>0.1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Along the entry section ($F_1$) a parabolic velocity profile is prescribed:

$$v_1(x_1 = 0, x_2; t) = \frac{1}{64} (16 - x_2^2),$$

$$v_2(x_1 = 0, x_2; t) = 0,$$  \hspace{1cm} (54)

and along the exit boundary ($F_3$)

$$v_1(x_1 = 16, x_2; t) = 1 - x_2^2,$$

$$v_2(x_1 = 16, x_2; t) = 0.$$  \hspace{1cm} (55)

Obviously, this velocity profile does not correspond to the fully developed velocity profile of the PTT or Leonov model. No stress boundary conditions are prescribed on $F_1$ and $F_3$. Further, along $F_2$, $v_2 = 0$, and along $F_4$, $F_5$ and $F_6$, $v_1 = v_2 = 0$ (stick).

The model parameters that have been used are given in Table 2, where $\dot{\gamma}_w$ is the wall shear rate at the exit and the Deborah number is defined as $\text{De} = \theta \dot{\gamma}_w$. Further, a relative time parameter is defined as $s = t/\theta$. As a result, $\tau_{1_1}$ along the line $x_1 \in [6, 16]$, $x_2 = 1$ is shown for the various cases. In all calculations a time step of 0.1 is chosen.

Figures 2 and 3 show $\tau_{1_1}$ for the Jeffreys model at $\text{De} = 5$ and $\text{De} = 10$. Clearly, the solution is unstable, especially for $\text{De} = 10$. Conclusion: the algorithm does not produce good results for large values of the Deborah number when the Jeffreys model is applied.

Figures 4–9 show $\tau_{1_1}$ for the PTT and the Leonov model, respectively, if Algorithm A1 is used. Notice the sharp capturing of the singularity near the corner (at $x = 8$). This indicates
Fig. 3. $\tau_{ij}$ Jeffreys model, $De = 10$.

Fig. 4. $\tau_{ij}$ PTT model, $De = 5$.

Fig. 5. $\tau_{ij}$ PTT model, $De = 10$. 
Fig. 6. $\tau_{11}$ PTT model, $De = 20$.

Fig. 7. $\tau_{11}$ Leonov model, $De = 5$.

Fig. 8. $\tau_{11}$ Leonov model, $De = 10$. 
that the discontinuity capturing operator does not introduce excessive diffusion. In all cases a stable solution is found with a small secondary peak right behind the reentrant corner.

Figure 10 shows the performance of Algorithm A2 for the Leonov model at \( \text{De} = 20 \). Comparison with Fig. 9 shows that Algorithm A2 is slightly more diffusive than Algorithm A1.

9. Conclusions and discussion

The methods proposed are consistent. In the case of the Jeffreys model, the solution becomes unstable for high values of the Deborah number. If the PTT or Leonov model is used, stable results are obtained, at least up to \( \text{De} = 20 \).

The precise definitions of the upwind parameters must be studied more closely; another choice might give more accurate results. Clearly, the use of the discontinuity capturing operator is an essential feature to obtain stable solutions.

An obvious disadvantage of the path followed in this paper is that steady-state solutions are only obtained as a limiting case. Further, the linear in time approximation increases the
number of unknowns for the stress problems by a factor of two when compared with a constant in time approximation. A constant in time approximation would be feasible, but numerical experiments have shown that it is much more sensitive to discontinuities (such as singularities).

An important extension of the algorithm would be towards super-critical flows in which the acceleration terms in the momentum equation are no longer neglected. In that case it might be necessary to use so-called entropy variables, as in compressible flow [4].

Another extension of great practical importance would be the inclusion of the energy equation.

In all cases, there is a great need for more efficient solvers. In this paper a direct solver was used, requiring massive computer resources. Indirect solvers, such as GMRES [4], or non-symmetric updating schemes, such as the Broyden update, must be investigated to render an efficient numerical technique.

References