Journal of Non-Newtonian Fluid Mechanics 211 (2014) 16–30

Contents lists available at ScienceDirect

Journal of Non-Newtonian Fluid Mechanics

journal homepage: http://www.elsevier.com/locate/jnnfm

On a modified non-singular log-conformation formulation for Johnson–Segalman viscoelastic fluids

Pierre Saramito *

CNRS, Lab. J. Kuntzmann, B.P. 53, 38041 Grenoble cedex 9, France

ARTICLE INFO

Article history:
Received 15 March 2014
Received in revised form 15 May 2014
Accepted 23 June 2014
Available online 1 July 2014

Keywords:
Johnson–Segalman viscoelastic fluid
Matrix logarithm
Newton method
Incompressible finite elements
Adaptive mesh
Lid-driven cavity

ABSTRACT

A modified log-conformation formulation of viscoelastic fluid flows is presented in this paper. This new formulation is non-singular for vanishing Weissenberg numbers and allows a direct steady numerical resolution by a Newton method. Moreover, an exact computation of all the terms of the linearized problem is provided. The use of an exact divergence-free finite element method for velocity–pressure approximation and a discontinuous Galerkin upwinding treatment for stresses leads to a robust discretization. A demonstration is provided by the computation of steady solutions at high Weissenberg numbers for the difficult benchmark case of the lid driven cavity flow. Numerical results are in good agreement, qualitatively with experiment measurements on real viscoelastic flows, and quantitatively with computations performed by others authors. The numerical algorithm is both robust and very efficient, as it requires a low mesh-invariant number of linear systems resolution to obtain solutions at high Weissenberg number. An adaptive mesh procedure is also presented: it allows representing accurately both boundary layers and the main and secondary vortices.

1. Introduction

The Johnson–Segalman [34] model is considered here, i.e. the upper convective derivative in the Oldroyd-B [47] model is replaced by a mixed Gordon–Schowalter [25] derivative with a parameter $a \in [-1,1]$:

$$\frac{\mathscr{D}_a\boldsymbol{\tau}}{\mathscr{D}t} = \frac{\partial\boldsymbol{\tau}}{\partial t} + (\mathbf{u}\cdot\nabla)\boldsymbol{\tau} - \boldsymbol{\tau}\mathbf{g}_a(\mathbf{u})^T - \mathbf{g}_a(\mathbf{u})\boldsymbol{\tau}, \tag{1}$$

for all symmetric tensor $\boldsymbol{\tau}$, where $\mathbf{u}$ is the velocity field and

$$\mathbf{g}_a(\mathbf{u}) = \frac{1+a}{2}\nabla\mathbf{u} - \frac{1-a}{2}\nabla\mathbf{u}^T \tag{2}$$

denotes a generalized gradient, with the convention $\nabla\mathbf{u} = \left(\frac{\partial u_i}{\partial x_j}\right)_{1\leqslant i,j\leqslant d}$ and $d = 2,3$. Here, $a \in [-1,1]$ denotes the parameter of the tensor derivative. Remark that $\mathbf{g}_a(\mathbf{u}) = W(\mathbf{u}) + aD(\mathbf{u})$ where $D(\mathbf{u}) = (\nabla\mathbf{u} + \nabla\mathbf{u}^T)/2$ and $W(\mathbf{u}) = (\nabla\mathbf{u} - \nabla\mathbf{u}^T)/2$ are the symmetric and skew-symmetric parts of the velocity gradient, respectively. When $a = 1$, then $\mathbf{g}_1(\mathbf{u}) = \nabla\mathbf{u}$ and the Gordon–Schowalter derivative coincides with the usual upper-convected tensor derivative. When $a = 0$, we obtain the corotational derivative and when $a = -1$, the lower convected derivative. Problems involving such tensor derivatives appear in non-Newtonian viscoelastic polymer melt flow problems (see e.g. [57,63]), in turbulence modeling with the $R_{ij} - \epsilon$ Reynolds stress turbulence models, for liquid crystals [20], fiber suspension [42] or active fluids [20], where a is related to the aspect ratio of the particles in suspension. All these models introduce a relaxation time that multiplies the tensor derivative. The corresponding dimensionless quantity is called the Weissenberg number.

The numerical computation of viscoelastic flows in non-trivial geometries has been founded as a very challenging enterprise. The failure of numerical methods when the Weissenberg number becomes large is known as the *high Weissenberg number problem*. In 1986, Keunings [36] observed that the maximal Weissenberg number reached by all algorithms was mesh-dependent: he deduced that this failure was due to a wrong numerical methodology. In 1987, Marchal and Crochet [45] presented numerical simulations at high Weissenberg numbers obtained with an improved numerical approach and pointed out the need of a mixed finite element methods and of an upwind treatment of the stress transport terms. This was the starting point of many numerical computations of viscoelastic flows at higher Weissenberg numbers (see e.g. [22,58,49] for a review of this period). In 2000, in a review paper, Keunings [36] observed the progress accomplished, but pointed out that the maximal Weissenberg number reached in most reported simulations was still clearly decreasing with mesh refinement. He also concluded on the crucial role of benchmark flow problems for future works.

* Tel.: +33 476 51 46 10.
E-mail address: Pierre.Saramito@imag.fr

http://dx.doi.org/10.1016/j.jnnfm.2014.06.008
0377-0257/

In 2004, Fattal and Kupferman [17,18] observed that some numerical instabilities are caused by the failure of polynomial functions to approximate accurately the exponential growth of the stress tensor, due to the presence of the deformation as a source term in the tensor transport equation. The deformation source term takes its origin in the two last terms on the right-hand side of the tensor derivative (1). The solution proposed by these authors was a change of unknown that scale logarithmically with the stress tensor. Exploiting the fact that the conformation tensor is symmetric positive definite, the stress transport equation was rewritten as an equation for the matrix logarithm of the conformation tensor (the so-called *log-conformation formulation*). Numerical experiments for the driven cavity benchmark showed that the maximal Weissenberg number reached was no longer mesh-dependent [18]. This idea was a new starting point and many improved numerical computations of viscoelastic flows were then performed (see e.g. [28,15,1,10] for a recent review).

The main objective of the present paper is to bring some novelties in the challenging field of numerical methods for viscoelastic fluid flows. These novelties develop in three main axes:

1. A new and different log-conformation formulation for viscoelastic models is proposed. This formulation is *non-singular* when the Weissenberg number vanishes, while the original one, as proposed by Fattal and Kupferman [17,18], degenerates, due to the apparition of the inverse of the Weissenberg number in the set of equations. With the present formulation, the problem reduces nicely to the Navier–Stokes equations at zero Weissenberg number. This is a major advantage, as it opens the door to continuation methods, starting smoothly at zero Weissenberg number and progressively increasing.
2. The *steady* problem is directly solved by a Newton method, while, to our knowledge, all previous approaches with the log-conformation formulation were time-dependent (see e.g. [15]). The derivation of a robust Newton solver for the strongly nonlinear steady log-formulation is based on an *exact* computation of all the derivatives. Notice that previous Newton methods relied on some finite difference methods for computing the Jacobian matrix, as the strong non-linearities were considered non-differentiable [35,15].
3. The discretization is based on an *incompressible* finite element method: the discrete velocity field satisfies the divergence free relation exactly. This is a major advantage when dealing with a transport equation: Recall that a non-zero divergence velocity field $\mathbf{u}$ introduces an additional source term $\operatorname{div}\mathbf{u}$ in the stress transport equation: this term can then generate an exponential growth of the stresses, and then overrules all the benefits of using a log-conformation formulation.

The outline of the paper is as follows. Section 1 starts with a presentation of the Johnson–Segalman model and its reformulation using the conformation tensor. Then the new non-singular log-conformation formulation is presented. The first section ends with the variational formulation of the steady problem. Section 2 begins with the space discretization using an incompressible finite element method. Then, the upwind discretization of the transport term, based on the discontinuous Galerkin method is presented. Then, the Newton method applied to the discrete problem is introduced with special care to the derivation of highly nonlinear terms produced by the log-conformation formulation. Section 2 ends with the Euler–Newton continuation algorithm. Section 3 presents numerical computations on the bi-dimensional driven cavity flow. A specific adaptive mesh procedure allows resolving accurately both boundary layers and secondary vortices. Finally, two appendices elaborate on the computation of some strongly nonlinear terms and their derivatives.

2. Continuous setup

2.1. Conformation tensor formulation

Let $\Omega \subset \mathbb{R}^d$ be a bounded open domain, $d = 2, 3$, and $t_f > 0$ a final time. The Jonson–Segalman problem writes :

(P_1): *find* $\tau, \mathbf{u}$ *and* p, *defined in* $]0, t_f[\times\Omega$, *such that*

$$\lambda\frac{\mathscr{D}_a\tau}{\mathscr{D}t} + \tau - 2\eta_p D(\mathbf{u}) = 0 \text{ in }]0,t_f[\times\Omega$$
$$\rho\frac{D\mathbf{u}}{Dt} - \mathbf{div}(\tau + 2\eta_v D(\mathbf{u}) - p\cdot I) = 0 \text{ in }]0,t_f[\times\Omega$$
$$\operatorname{div}\mathbf{u} = 0 \text{ in }]0,t_f[\times\Omega$$
$$\lambda(\tau - \tau_\Gamma) = 0 \text{ on}]0,t_f[\times\partial\Omega_-$$
$$\mathbf{u} = \mathbf{u}_\Gamma \text{ on }]0,t_f[\times\partial\Omega$$
$$\lambda\tau(0) = \lambda\tau_0 \quad \text{and} \quad \mathbf{u}(0) = \mathbf{u}_0 \text{ in } \Omega$$

where $\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}$ denotes the Lagrange derivative, $\lambda \geqslant 0$ is the relaxation time, $\eta_p > 0$ and $\eta_v \geqslant 0$ are the polymer and solvent viscosities, respectively, $\rho \geqslant 0$ is the constant density, τ_0, $\mathbf{u}_0$, τ_Γ and $\mathbf{u}_\Gamma$ are given initial and boundary conditions and $\partial\Omega_-$ is the upstream boundary, defined by $\partial\Omega_- = \{x \in \partial\Omega; \mathbf{u}_\Gamma\cdot\mathbf{n}(x) < 0\}$. The total Cauchy stress tensor is $\boldsymbol{\sigma}_{\text{tot}} = -pI + 2\eta_v D(\mathbf{u}) + \tau$. When $\lambda = 0$, the problem reduces to the Navier–Stokes equations with a total viscosity $\eta_v + \eta_p$ and $\boldsymbol{\tau} = 2\eta_p D(\mathbf{u})$.

Several studies published are concerned with the existence, uniqueness and regularity of a solution for this problem. In 1985, Renardy [55] proved the existence of a solution to the steady problem. In 1987, Guillopé and Saut [26] proved that there exists a unique strong solution local in time and that this solution is globally defined when the data are small and the fluid is not too elastic (see also [40] for a different approach). In 1998, Fernández-Cara et al. [19] extended this result to the large elasticity case and for an arbitrarily large but finite final time t_f. In 2000, Lions and Masmoudi [41] obtained an improved result (t_f could be infinite) in the particular case of a corotational derivative ($a = 0$). In 2005, Kupferman et al. [37] obtained a more explicit condition for the global in time existence of the solution when $\Omega = \mathbb{R}^3$.

The solution $(\tau, \mathbf{u}, p)$ exhibits some important properties. Hulsen [32, p. 6, Eq. (30)] (see also Kwon and Leonov [38, p. 31, Eq. (10)], Lee et al. [39, p. 383]) introduced the following definition of the conformation tensor when both a and λ are non-zero:

$$\mathbf{c} = \boldsymbol{\tau} + \frac{\eta_p}{a\lambda}I$$

where I denotes the $d \times d$ identity matrix. Notice that $\mathbf{c}$ has the dimension of a stress and is undefined when $\lambda = 0$ or $a = 0$. Also, the conformation tensor $\mathbf{c}$ admits an explicit expression in integral form involving the unknown velocity field $\mathbf{u}$ [39, p. 385, Eq. (3.29)]. From (1) we get $\frac{\mathscr{D}_a I}{\mathscr{D}t} = -2aD(\mathbf{u})$ then $\frac{\mathscr{D}_a\boldsymbol{\tau}}{\mathscr{D}t} = \frac{\mathscr{D}_a\mathbf{c}}{\mathscr{D}t} - \frac{2\eta_p}{\lambda}D(\mathbf{u})$. Then, the Johnson–Segalman problem becomes:

(P_2): *find* $\mathbf{c}, \mathbf{u}$ *and* p, *defined in* $]0, t_f[\times\Omega$, *such that*

$$\lambda\frac{\mathscr{D}_a\mathbf{c}}{\mathscr{D}t} + \mathbf{c} = \frac{\eta_p}{a\lambda}I \text{ in }]0,t_f[\times\Omega \tag{3a}$$
$$\rho\frac{D\mathbf{u}}{Dt} - \mathbf{div}(\mathbf{c} + 2\eta_v D(\mathbf{u}) - p\cdot I) = 0 \text{ in }]0,t_f[\times\Omega \tag{3b}$$
$$\operatorname{div}\mathbf{u} = 0 \text{ in }]0,t_f[\times\Omega \tag{3c}$$
$$\lambda(\mathbf{c} - \mathbf{c}_\Gamma) = 0 \text{ on }]0,t_f[\times\partial\Omega_- \tag{3d}$$
$$\mathbf{u} = \mathbf{u}_\Gamma \text{ on }]0,t_f[\times\partial\Omega \tag{3e}$$
$$\lambda\mathbf{c}(0) = \lambda\mathbf{c}_0 \quad \text{and} \quad \mathbf{u}(0) = \mathbf{u}_0 \text{ in } \Omega \tag{3f}$$

In 1990, Hulsen [32] showed that when the initial condition $\mathbf{c}_0$ in (3f) is symmetric definite positive, and when $\nabla\mathbf{u}$ is bounded, then $\mathbf{c}$ remains symmetric definite positive at any time $t > 0$. Notice that when $\lambda = 0$, the problem degenerates, due to the appa-

rition of $a\lambda$ in the denominator in the constitutive Eq. (3a). In the case of the upper convected derivative ($a=1$), Fattal and Kupferman [17,18] introduced a change of variable $\boldsymbol{\psi}=\log\left(\frac{a\lambda}{\eta_p}\mathbf{c}\right)$, the so-called log-conformation formulation. Nevertheless, when $\lambda=0$ the expected solution is simply $\boldsymbol{\psi}=0$, while the reformulated problem still degenerates due to a division by λ in the conservation of momentum equation (see [17, p. 283], the last equation of the page). This change of variable was then applied in [11] to a generalized constitutive model covering the present Johnson–Segalman model ($a\in[-1,1]$), but the obtained problem still degenerated when $\lambda=0$. The next paragraph presents a slightly different change of variable that solves this degeneracy nicely. The new formulation will no longer be singular when $\lambda=0$. This opens the possibility of building a robust steady solver based on the Newton method and a continuation algorithm, starting smoothly at $\lambda=0$.

2.2. A new log-conformation formulation

In order to obtain a non-singular formulation, a different change of the logarithmic variable is introduced:

$$\boldsymbol{\chi}=\frac{\eta_p}{a\lambda}\log\left(\frac{a\lambda}{\eta_p}\mathbf{c}\right)=\frac{\eta_p}{a\lambda}\log\left(I+\frac{a\lambda}{\eta_p}\boldsymbol{\tau}\right) \tag{4a}$$

$$\Longleftrightarrow \mathbf{c}=\frac{\eta_p}{a\lambda}\exp\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)\quad\text{and}\quad\boldsymbol{\tau}=\frac{\eta_p}{a\lambda}\left(\exp\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)-I\right) \tag{4b}$$

Notice that both $\boldsymbol{\chi},\mathbf{c}$ and $\boldsymbol{\tau}$ have the dimension of stress. At the limit case $a\lambda=0$, the new variable $\boldsymbol{\chi}$ is still well defined, as showed by a simple Taylor expansion of the matrix logarithm in the vicinity of $a\lambda=0$:

$$\boldsymbol{\chi}=\boldsymbol{\tau}-\left(\frac{a\lambda}{\eta_p}\right)\boldsymbol{\tau}^2+\cdots+\frac{1}{n}\left(\frac{a\lambda}{\eta_p}\right)^{n-1}\boldsymbol{\tau}^n+\cdots$$

For instance, when $\lambda=0$, then $\boldsymbol{\tau}=2\eta_p D(\mathbf{u})$ is the solution of the Johnson–Segalman model which reduces to a Newtonian fluid and $\boldsymbol{\chi}=\boldsymbol{\tau}=2\eta_p D(\mathbf{u})$. For completeness, the derivation of the present log-conformation formulation is provided here, since there are some subtle modifications due to the different change of variable and the introduction of the Gordon–Schowalter derivative parameter a. The impatient reader can jump directly to the new formulation (8a)–(8f)

The main technique of the present derivation of the log-conformation is related to the evolution of the principal axes of the conformation tensor [33]. Recall that the symmetric positive definite matrix $\mathbf{c}$ can always be diagonalized as $\mathbf{c}=\mathbf{q}\tilde{\mathbf{c}}\mathbf{q}^T$ where $\tilde{\mathbf{c}}$ is diagonal and $\mathbf{q}$ is an orthogonal tensor, i.e. $\mathbf{q}\mathbf{q}^T=\mathbf{q}^T\mathbf{q}=I$. For convenience, let us denote by $\dot{\boldsymbol{\tau}}=\frac{D\boldsymbol{\tau}}{Dt}=\frac{\partial\boldsymbol{\tau}}{\partial t}+\mathbf{u}\cdot\nabla\boldsymbol{\tau}$ the Lagrangian derivative of any tensor $\boldsymbol{\tau}$. Then $\dot{\mathbf{c}}=\dot{\mathbf{q}}\tilde{\mathbf{c}}\mathbf{q}^T+\mathbf{q}\tilde{\mathbf{c}}\dot{\mathbf{q}}^T+\dot{\mathbf{q}}\tilde{\mathbf{c}}\mathbf{q}^T$. Let us introduce $\mathbf{r}=\dot{\mathbf{q}}\mathbf{q}^T$. By differentiating $\mathbf{q}\mathbf{q}^T=I$ we get $\dot{\mathbf{q}}\mathbf{q}^T+\mathbf{q}\dot{\mathbf{q}}^T=0$ which also writes $\mathbf{r}^T=-\mathbf{r}$. Then $\mathbf{r}$ is skew-symmetric. We also obtain $\dot{\mathbf{q}}=-\mathbf{r}\mathbf{q}$ and $\dot{\mathbf{q}}^T=-\mathbf{q}^T\mathbf{r}^T=\mathbf{q}^T\mathbf{r}$ and then $\dot{\mathbf{c}}=\mathbf{q}\dot{\tilde{\mathbf{c}}}\mathbf{q}^T-\mathbf{q}\tilde{\mathbf{c}}\mathbf{q}^T\mathbf{r}^T-\mathbf{r}\mathbf{q}\tilde{\mathbf{c}}\mathbf{q}^T$. Next, let $\tilde{\mathbf{r}}=\mathbf{q}^T\mathbf{r}\mathbf{q}$. Note that $\tilde{\mathbf{r}}$ is also skew-symmetric. We have $\mathbf{r}=\mathbf{q}\tilde{\mathbf{r}}\mathbf{q}^T$ and then

$$\dot{\mathbf{c}}=\mathbf{q}\left(\dot{\tilde{\mathbf{c}}}-\tilde{\mathbf{c}}\tilde{\mathbf{r}}^T-\tilde{\mathbf{r}}\tilde{\mathbf{c}}\right)\mathbf{q}^T \tag{5a}$$

The generalized gradient introduced in (2) decomposes in this eigensystem as $\mathbf{g}_a(\mathbf{u})=\mathbf{q}\tilde{\mathbf{g}}_a\mathbf{q}^T$ where $\tilde{\mathbf{g}}_a$ is not *a priori* diagonal, since in general $\mathbf{c}$ and $\mathbf{g}_a(\mathbf{u})$ are not expected to share the same eigenvectors. Then $-\mathbf{c}\mathbf{g}_a(\mathbf{u})^T-\mathbf{g}_a(\mathbf{u})\mathbf{c}=\mathbf{q}(-\tilde{\mathbf{c}}\tilde{\mathbf{g}}_a^T-\tilde{\mathbf{g}}_a\tilde{\mathbf{c}})\mathbf{q}^T$ and the Gordon–Schowalter derivative (1) writes:

$$\frac{\mathscr{D}_a\mathbf{c}}{\mathscr{D}t}=\mathbf{q}\left(\dot{\tilde{\mathbf{c}}}-\tilde{\mathbf{c}}(\tilde{\mathbf{r}}+\tilde{\mathbf{g}}_a)^T-(\tilde{\mathbf{r}}+\tilde{\mathbf{g}}_a)\tilde{\mathbf{c}}\right)\mathbf{q}^T \tag{5b}$$

The constitutive Eq. (3a) leads to

$$\dot{\tilde{\mathbf{c}}}=\frac{\eta_p}{a\lambda^2}I-\frac{1}{\lambda}\tilde{\mathbf{c}}+(\tilde{\mathbf{r}}+\tilde{\mathbf{g}}_a)\tilde{\mathbf{c}}+\tilde{\mathbf{c}}(\tilde{\mathbf{r}}+\tilde{\mathbf{g}}_a)^T \tag{5c}$$

This is a tensorial equation. Let $(c_i)_{1\leqslant i\leqslant d}$ be the eigenvalues of $\mathbf{c}$. Recall that $\tilde{\mathbf{c}}$ is diagonal and that its i-th diagonal entry is c_i. Then, by taking the diagonal entries $i=j,\ 1\leqslant i\leqslant d$, of the tensorial Eq. (5c), we get a differential equation for c_i :

$$\dot{c}_i=\frac{\eta_p}{a\lambda^2}-\left(\frac{1}{\lambda}-2a\,\tilde{d}_{i,i}\right)c_i$$

where we have set $\tilde{\mathbf{d}}=\mathbf{q}^TD(\mathbf{u})\mathbf{q}=(\tilde{d}_{i,j})_{1\leqslant i,j\leqslant d}$. The previous relation also writes

$$\dot{\tilde{\mathbf{c}}}=\frac{\eta_p}{a\lambda^2}I-\left(\frac{I}{\lambda}-2a\ \mathrm{diag}(\tilde{\mathbf{d}})\right)\tilde{\mathbf{c}} \tag{5d}$$

Now, let us perform the change of variable (4a) and (4b). The problem will be rewritten in terms of the variable

$$\boldsymbol{\chi}=\frac{\eta_p}{a\lambda}\log\left(\frac{a\lambda}{\eta_p}\mathbf{c}\right)=\mathbf{q}\frac{\eta_p}{a\lambda}\log\left(\frac{a\lambda}{\eta_p}\tilde{\mathbf{c}}\right)\mathbf{q}^T=\mathbf{q}\tilde{\boldsymbol{\chi}}\mathbf{q}^T$$

where $\tilde{\boldsymbol{\chi}}$ is the diagonal tensor with diagonal entries

$$\chi_i=\frac{\eta_p}{a\lambda}\log\left(\frac{a\lambda}{\eta_p}c_i\right)\Longleftrightarrow c_i=\frac{\eta_p}{a\lambda}\exp\left(\frac{a\lambda}{\eta_p}\chi_i\right),\quad 1\leqslant i\leqslant d$$

Thus $\dot{\tilde{\chi}}_i=\frac{\eta_p}{a\lambda}\frac{\dot{\tilde{c}}_i}{\tilde{c}_i}$ and, since $\tilde{\mathbf{c}}$, $\tilde{\chi}$, $\dot{\tilde{\mathbf{c}}}$ and $\dot{\tilde{\chi}}$ are all diagonal:

$$\begin{aligned}\dot{\tilde{\boldsymbol{\chi}}}&=\frac{\eta_p}{a\lambda}\dot{\tilde{\mathbf{c}}}\tilde{\mathbf{c}}^{-1}\\&=\frac{\eta_p}{a\lambda}\left(\frac{\eta_p}{a\lambda^2}I-\left(\frac{I}{\lambda}-2a\,\mathrm{diag}(\tilde{\mathbf{d}})\right)\tilde{\mathbf{c}}\right)\tilde{\mathbf{c}}^{-1},\quad\text{from (5d)}\\&=-\frac{1}{\lambda}\left\{\frac{\eta_p}{a\lambda}\left(I-\frac{a\lambda}{\eta_p}\tilde{\mathbf{c}}^{-1}\right)\right\}+\frac{2\eta_p}{\lambda}\mathrm{diag}(\tilde{\mathbf{d}})\end{aligned}$$

From (4a), we have $\mathbf{c}^{-1}=\frac{a\lambda}{\eta_p}\exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)$ and then

$$\dot{\tilde{\boldsymbol{\chi}}}=-\frac{1}{\lambda}\left\{\frac{\eta_p}{a\lambda}\left(I-\exp\left(-\frac{a\lambda}{\eta_p}\tilde{\boldsymbol{\chi}}\right)\right)\right\}+\frac{2\eta_p}{\lambda}\mathrm{diag}(\tilde{\mathbf{d}})$$

Then, replacing in (5a) the instances of $\mathbf{c}$ and $\tilde{\mathbf{c}}$ by $\boldsymbol{\chi}$ and $\tilde{\boldsymbol{\chi}}$ respectively, the Lagrangian derivative of $\boldsymbol{\chi}$ reads:

$$\begin{aligned}\dot{\boldsymbol{\chi}}=&\mathbf{q}\left(\dot{\tilde{\boldsymbol{\chi}}}-\tilde{\boldsymbol{\chi}}\tilde{\mathbf{r}}^T-\tilde{\mathbf{r}}\tilde{\boldsymbol{\chi}}\right)\mathbf{q}^T\\=&-\frac{1}{\lambda}\left\{\frac{\eta_p}{a\lambda}\left(I-\exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)\right)\right\}+\mathbf{q}\left(\frac{2\eta_p}{\lambda}\mathrm{diag}(\tilde{\mathbf{d}})-\tilde{\boldsymbol{\chi}}\tilde{\mathbf{r}}^T-\tilde{\mathbf{r}}\tilde{\boldsymbol{\chi}}\right)\mathbf{q}^T\end{aligned} \tag{5e}$$

Equivalently, this relation can be written

$$\dot{\boldsymbol{\chi}}+\phi_a(\boldsymbol{\chi},\nabla\mathbf{u})+\frac{1}{\lambda}\left\{\frac{\eta_p}{a\lambda}\left(I-\exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)\right)\right\}=0 \tag{5f}$$

where we have introduced the notation:

$$\begin{aligned}\phi_a(\boldsymbol{\chi},\nabla\mathbf{u})=&\mathbf{q}\tilde{\phi}_a\mathbf{q}^T\\\tilde{\phi}_a=&\tilde{\boldsymbol{\chi}}\tilde{\mathbf{r}}^T+\tilde{\mathbf{r}}\tilde{\boldsymbol{\chi}}-\frac{2\eta_p}{\lambda}\mathrm{diag}(\tilde{\mathbf{d}})\end{aligned}$$

Note that $\phi_a(\boldsymbol{\chi},\nabla\mathbf{u})$ is symmetric since $\mathrm{diag}(\tilde{\mathbf{d}})$ is diagonal and $\tilde{\mathbf{r}}\tilde{\boldsymbol{\chi}}+\tilde{\boldsymbol{\chi}}\tilde{\mathbf{r}}^T$ is symmetric. Notice also that both $\mathbf{q}$ and $\mathrm{diag}(\tilde{\mathbf{d}})$ are directly computable from $\boldsymbol{\chi}$ and $\nabla\mathbf{u}$ since $\mathbf{q}$ is the tensor containing the eigenvectors of $\boldsymbol{\chi}$ and $\tilde{\mathbf{d}}=\mathbf{q}^TD(\mathbf{u})\mathbf{q}$. The computation of $\tilde{\mathbf{r}}\tilde{\boldsymbol{\chi}}+\tilde{\boldsymbol{\chi}}\tilde{\mathbf{r}}^T$ is more technical. Let us compute $\tilde{\mathbf{r}}=(\tilde{r}_{i,j})_{1\leqslant i,j\leqslant d}$. Since $\tilde{\mathbf{r}}^T=-\tilde{\mathbf{r}}$ we have $\tilde{r}_{i,i}=0$. By taking the off-diagonal entries $i\neq j, 1\leqslant i,j\leqslant d$, in (5c), we get

$$0=c_i(-\tilde{r}_{i,j}+\tilde{g}_{a;j,i})+(\tilde{r}_{i,j}+\tilde{g}_{a;i,j})c_j$$

where we have used $\tilde{r}_{j,i} = -\tilde{r}_{i,j}$ and set $\tilde{g}_{a;i,j} = \tilde{w}_{i,j} + a\tilde{d}_{i,j}$ and $\tilde{\mathbf{w}} = \mathbf{q}^T W(\mathbf{u})\mathbf{q} = (\tilde{w}_{i,j})_{1\leqslant i,j\leqslant d}$. When $c_i \neq c_j$ we obtain by simple development:

$$\tilde{r}_{i,j} = \frac{\tilde{g}_{a;i,j}c_j + \tilde{g}_{a;j,i}c_i}{c_i - c_j} = -\tilde{w}_{i,j} + a\left(\frac{c_i + c_j}{c_i - c_j}\right)\tilde{d}_{i,j}$$
$$= -\tilde{w}_{i,j} + \frac{a\tilde{d}_{i,j}}{\tanh\left(\frac{a\lambda}{\eta_p}\left(\frac{\chi_i - \chi_j}{2}\right)\right)}$$

Note that now $\tilde{r}$ is directly computable from $\boldsymbol{\chi}$ and $\nabla\mathbf{u}$ but is still undetermined when $\chi_i = \chi_j$ or $a\lambda = 0$. However, always when $i \neq j$ and $c_i \neq c_j$:

$$\tilde{\phi}_{a;i,j} = \chi_i\tilde{r}_{j,i} + \tilde{r}_{i,j}\chi_j = -\left(\chi_i - \chi_j\right)\tilde{r}_{i,j}$$
$$= \left(\chi_i - \chi_j\right)\tilde{w}_{i,j} - \left(\frac{\frac{a\lambda}{\eta_p}\left(\frac{\chi_i - \chi_j}{2}\right)}{\tanh\left(\frac{a\lambda}{\eta_p}\left(\frac{\chi_i - \chi_j}{2}\right)\right)}\right)\frac{2\eta_p\tilde{d}_{i,j}}{\lambda}$$

As $\lim_{x\to 0}\frac{x}{\tanh(x)} = 1$, we obtain:

$$\tilde{\phi}_{a;i,j} = -\frac{2\eta_p}{\lambda}\tilde{d}_{i,j} \quad \text{when} \quad \chi_i = \chi_j \quad \text{or} \quad a\lambda = 0$$

Finally $\phi_a(\boldsymbol{\chi}, \nabla\mathbf{u})$ is directly computable from $\boldsymbol{\chi}$ and $\nabla\mathbf{u}$, in all cases. Note that when all the eigenvalues of $\boldsymbol{\chi}$ are equal, then $\phi_a(\boldsymbol{\chi}, \nabla\mathbf{u}) = -\frac{2\eta_p}{\lambda}D(\mathbf{u})$. Also, when $a = 0$, we have $\phi_a(\boldsymbol{\chi}, \nabla\mathbf{u}) = \boldsymbol{\chi}W(\mathbf{u}) - W(\mathbf{u})\boldsymbol{\chi} - \frac{2\eta_p}{\lambda}D(\mathbf{u})$. This suggests the introduction of a nonlinear form $\kappa(\cdot,\cdot)$ such that

$$\phi_a(\boldsymbol{\chi}, \nabla\mathbf{u}) = \boldsymbol{\chi}W(\mathbf{u}) - W(\mathbf{u})\boldsymbol{\chi} + \frac{\eta_p}{\lambda}\kappa\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}, 2D(\mathbf{u})\right) - \frac{2\eta_p}{\lambda}D(\mathbf{u})$$

Then (5f) becomes

$$\lambda(\dot{\boldsymbol{\chi}} + \boldsymbol{\chi}W(\mathbf{u}) - W(\mathbf{u})\boldsymbol{\chi}) + \eta_p\kappa\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}, 2D(\mathbf{u})\right)$$
$$+\frac{\eta_p}{a\lambda}\left(I - \exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)\right) = 2\eta_p D(\mathbf{u}) \tag{5g}$$

Let $\mathbb{R}^{d\times d}_s$ denotes the set of $d\times d$ symmetric real matrix. The form $\kappa(\boldsymbol{\beta}, \boldsymbol{\gamma})$ is defined for all $\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^{d\times d}_s$ by

$$\kappa(\boldsymbol{\beta}, \boldsymbol{\gamma}) = \mathbf{q}\tilde{\kappa}\mathbf{q}^T \tag{6a}$$
$$\tilde{\kappa}_{i,j} = \hat{\kappa}\left(\frac{\beta_i - \beta_j}{2}\right)\tilde{\gamma}_{i,j}, \quad 1 \leqslant i,j \leqslant d \tag{6b}$$

where $\tilde{\boldsymbol{\beta}} = \mathbf{q}^T\boldsymbol{\beta}\mathbf{q} = \mathrm{diag}(\beta_i)$ and $\tilde{\boldsymbol{\gamma}} = \mathbf{q}^T\boldsymbol{\gamma}\mathbf{q}$. Here, $\hat{\kappa}(x)$ is defined for all $x \in \mathbb{R}$ by

$$\hat{\kappa}(x) = \begin{cases} 0 & \text{when } x = 0 \\ 1 - \frac{x}{\tanh(x)} & \text{otherwise} \end{cases} \tag{6c}$$

For convenience, let us define the following function, for all real $\mu \geqslant 0$ and $\boldsymbol{\chi} \in \mathbb{R}^{d\times d}_s$

$$f(\mu, \boldsymbol{\chi}) = \begin{cases} 0 & \text{when } \mu = 0 \\ \frac{\exp(\mu\boldsymbol{\chi}) - I}{\mu} - \boldsymbol{\chi} & \text{otherwise} \end{cases} \tag{7}$$

Note that the trace of $\lambda f\left(\frac{a\lambda}{\eta_p}, \boldsymbol{\chi}\right)$ represents a local *free energy* stored by the micro-scale mechanical system (see e.g. [5,69,48]). This concept was recently used by Hu and Lelièvre [31,7] for stability estimates and is employed later in this paper for an automatic adaptive meshing purpose.

The new log-conformation formulation of the Johnson–Segalman problem is:

(P_3): *find* $\boldsymbol{\chi}, \mathbf{u}$ *and* p, *defined in* $]0, t_f[\times\Omega$, *such that*

$$\lambda\frac{\mathscr{D}_0\boldsymbol{\chi}}{\mathscr{D}t} + \boldsymbol{\chi} - f\left(\frac{a\lambda}{\eta_p}, -\boldsymbol{\chi}\right) + \eta_p\kappa\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}, 2D(\mathbf{u})\right) - 2\eta_p D(\mathbf{u}) = 0 \text{ in }]0, t_f[\times\Omega \tag{8a}$$
$$\rho\frac{D\mathbf{u}}{Dt} - \mathbf{div}\left(\boldsymbol{\chi} + f\left(\frac{a\lambda}{\eta_p}, \boldsymbol{\chi}\right) + 2\eta_v D(\mathbf{u}) - p\cdot I\right) = 0 \text{ in }]0, t_f[\times\Omega \tag{8b}$$
$$\mathrm{div}\,\mathbf{u} = 0 \text{ in }]0, t_f[\times\Omega \tag{8c}$$
$$\lambda(\boldsymbol{\chi} - \boldsymbol{\chi}_\Gamma) = 0 \text{ on }]0, t_f[\times\partial\Omega_- \tag{8d}$$
$$\mathbf{u} = \mathbf{u}_\Gamma \text{ on }]0, t_f[\times\partial\Omega \tag{8e}$$
$$\lambda\boldsymbol{\chi}(0) = \lambda\boldsymbol{\chi}_0 \quad \text{and} \quad \mathbf{u}(0) = \mathbf{u}_0 \text{ in } \Omega \tag{8f}$$

where $\boldsymbol{\chi}_0$, $\boldsymbol{\chi}_\Gamma$, $\mathbf{u}_0$ and $\mathbf{u}_\Gamma$ are given. The functions f and κ are defined by (7) and (6a)–(6c), respectively. The elastic stress $\boldsymbol{\tau}$ is computed explicitly as $\boldsymbol{\tau} = \boldsymbol{\chi} + f\left(\frac{a\lambda}{\eta_p}, \boldsymbol{\chi}\right)$ and the total Cauchy stress tensor is $\boldsymbol{\sigma}_{\mathrm{tot}} = -pI + 2\eta_v D(\mathbf{u}) + \tau$.

Remark 1 ($\lambda = 0$: *Newtonian fluid*). When $\lambda = 0$, the problem reduces as expected to the Navier–Stokes equations with a total viscosity $\eta_v + \eta_p$ and $\boldsymbol{\chi} = 2\eta_p D(\mathbf{u})$.

Remark 2 ($a = 0$: *corotational Johnson–Segalman fluid*). When $a = 0$, the present log-formulation of the problem reduces nicely to the corotational Johnson–Segalman problem with $\boldsymbol{\tau} = \boldsymbol{\chi}$ is the elastic stress, since $f(0,.)$ and $\kappa(0,.)$ both are zero, as showed in the next two properties.

Remark 3. [Corotational versus Gordon–Schowalter tensor derivatives] Note the corotational derivative for $\boldsymbol{\chi}$ in (8a). Thus, the present log-conformation formulation of the general Johnson–Segalman model can be interpreted as a nonlinear perturbation with nonlinear terms (f and κ) of the corotational Johnson–Segalman model. In the following section, we show that the numerical treatment of corotational derivative is much simpler than in the general Gordon–Schowalter case.

Property 1 (*Regularity of f*). *The function f defined by (7) is continuously differentiable and*

$$f(0, \boldsymbol{\chi}) = f(\mu, 0) = 0, \quad \forall\mu \in \mathbb{R}^+, \quad \forall\boldsymbol{\chi} \in \mathbb{R}^{d\times d}_s$$

Proof. This result follows from a simple Taylor expansion:

$$f(\mu, \boldsymbol{\chi}) = \mu\boldsymbol{\chi}^2\left(\frac{I}{2} + \frac{\mu\boldsymbol{\chi}}{6} + \cdots + \frac{(\mu\boldsymbol{\chi})^n}{(n+2)!} + \cdots +\right) \qquad \square$$

Property 2 (*Skew symmetry of* κ). *The function* κ *defined by (6a)–(6c) satisfies the following properties:*

1. $\kappa(\cdot,\cdot)$ *is nonlinear with respect to its first variable* $\boldsymbol{\beta}$ *and is linear with respect to its second variable* $\boldsymbol{\gamma}$.
2. κ *is traceless:*
$$\mathrm{tr}\,\kappa(\boldsymbol{\beta}, \boldsymbol{\gamma}) = 0, \quad \forall\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^{d\times d}_s$$
3. $\kappa(\boldsymbol{\beta}, \boldsymbol{\gamma}) = 0$ *when* $\boldsymbol{\beta}$ *and* $\boldsymbol{\gamma}$ *are aligned, i.e. when they share the same eigensystem. As a special case,* $\kappa(\boldsymbol{\beta}, \boldsymbol{\gamma}) = 0$ *when all eigenvalues of* $\boldsymbol{\beta}$ *are equal.*
4. κ *is skew-symmetric with respect to its first variable:*
$$\kappa(\boldsymbol{\beta}, \boldsymbol{\gamma}) : \boldsymbol{\beta} = 0, \quad \forall\boldsymbol{\beta}, \boldsymbol{\gamma} \in \mathbb{R}^{d\times d}_s$$

The skew symmetry extends to
$$\kappa(\boldsymbol{\beta}, \boldsymbol{\gamma}) : \boldsymbol{\sigma} = 0, \quad \forall\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\sigma} \in \mathbb{R}^{d\times d}_s \text{ and } \boldsymbol{\beta} \text{ and } \boldsymbol{\sigma} \text{ share the same eigensystem} \tag{9}$$

5. κ *is continuously differentiable everywhere and*

$$\kappa(0,\boldsymbol{\gamma})=0 \quad \text{and} \quad \frac{\partial\kappa}{\partial\boldsymbol{\beta}}(0,\boldsymbol{\gamma})=0, \quad \forall\boldsymbol{\gamma}\in\mathbb{R}^{d\times d}_s$$

Proof. Remark that, from (6a)–(6c), we have $\tilde{\kappa}_{k,k}=0$ for all $k, 1\leqslant k\leqslant d$ which leads to the traceless property. When $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are aligned then $\tilde{\boldsymbol{\gamma}}$ is diagonal and then $\tilde{\kappa}_{i,j}=0$. The skew-symmetry result can be shown by the following development:

$$\begin{aligned}\kappa(\boldsymbol{\beta},\boldsymbol{\gamma}):\boldsymbol{\beta} &= (\mathbf{q}\tilde{\kappa}\mathbf{q}^T):(\mathbf{q}\tilde{\boldsymbol{\beta}}\mathbf{q}^T)\\ &= \sum_{i,j,k,l,m=0}^{d-1}\mathbf{q}_{i,j}\tilde{\kappa}_{j,k}\mathbf{q}_{l,k}\mathbf{q}_{i,m}\beta_m\mathbf{q}_{l,m}\\ &= \sum_{j,k,m=0}^{d-1}\delta_{j,m}\delta_{k,m}\tilde{\kappa}_{j,k}\beta_m \quad \text{since} \sum_{i=0}^{d-1}\mathbf{q}_{i,j}\mathbf{q}_{i,m}=\delta_{j,m}\\ &\quad \text{and} \quad \sum_{l=0}^{d-1}\mathbf{q}_{l,k}\mathbf{q}_{l,m}=\delta_{k,m}=\sum_{k=1}^{d-1}\tilde{\kappa}_{k,k}\boldsymbol{\beta}_k=0 \quad \text{since } \tilde{\kappa}_{k,k}=0\end{aligned}$$

When $\boldsymbol{\beta}=0$, then all eigenvalues of $\boldsymbol{\beta}$ are equal and then $\kappa(0,\boldsymbol{\gamma})$ (point 3). The proof of the differentiability of κ is postponed to Appendix A.2, as it requires some technical developments. □

2.3. Variational formulation of the steady problem

Here, the variational formulation of the steady version of the log-conformation formulation (8a) and (8f) of the Johnson–Segalman problem is considered. The inertia term $(\mathbf{u}\cdot\nabla)\mathbf{u}$ in the conservation of momentum (8b) is neglected. This is a common assumption in such flow simulations and the main difficulty is associated with the nonlinear terms related to the elasticity $\lambda\geqslant 0$. Let us introduce three functional spaces: $T=L^2(\Omega)^{d\times d}_s$ for square summable symmetric tensors, $V(\mathbf{u}_\Gamma)=\{\mathbf{v}\in H^1(\Omega)^d;\ \mathbf{v}=\mathbf{u}_\Gamma \text{ on } \partial\Omega\}$ for velocities with square summable gradients which satisfies the boundary condition, and $Q=L^2_0(\Omega)=\{q\in L^2(\Omega);\int_\Omega q\,\mathrm{d}x=0\}$ for square summable pressures with zero average value. Next, let the following forms be defined for all $\boldsymbol{\chi},\xi\in T$, $\mathbf{u},\mathbf{v}\in H^1(\Omega)^d$ $p,q\in L^2(\Omega)$:

$$\begin{aligned}t(\mathbf{u};\boldsymbol{\chi},\xi) &= \lambda\int_\Omega((\mathbf{u}\cdot\nabla)\boldsymbol{\chi}):\xi\,\mathrm{d}x+\lambda\int_{\partial\Omega}\max(0,-\mathbf{u}_\Gamma.\mathbf{n})\boldsymbol{\chi}:\xi\,\mathrm{d}s\\ &\quad +\lambda\int_\Omega(\boldsymbol{\chi}W(\mathbf{u})-W(\mathbf{u})\boldsymbol{\chi}):\xi\,\mathrm{d}x+\eta_p\int_\Omega\kappa\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi},2D(\mathbf{u})\right):\xi\,\mathrm{d}x \quad (10a)\\ l(\xi) &= \lambda\int_{\partial\Omega}\max(0,-\mathbf{u}_\Gamma.\mathbf{n})\boldsymbol{\chi}_\Gamma:\xi\,\mathrm{d}s \quad (10b)\\ a_0(\boldsymbol{\chi},\xi) &= \int_\Omega\boldsymbol{\chi}:\xi\,\mathrm{d}x-\int_\Omega f\left(\frac{a\lambda}{\eta_p},-\boldsymbol{\chi}\right):\xi\,\mathrm{d}x \quad (10c)\\ b_1(\xi,\mathbf{u}) &= -2\eta_p\int_\Omega\xi:D(\mathbf{u})\mathrm{d}x \quad (10d)\\ b_2(\boldsymbol{\chi},\mathbf{v}) &= -\int_\Omega\boldsymbol{\chi}:D(\mathbf{v})\mathrm{d}x-\int_\Omega f\left(\frac{a\lambda}{\eta_p},\boldsymbol{\chi}\right):D(\mathbf{v})\mathrm{d}x \quad (10e)\\ c(\mathbf{u},\mathbf{v}) &= 2\eta_v\int_\Omega D(\mathbf{u}):D(\mathbf{v})\mathrm{d}x \quad (10f)\\ d(\mathbf{u},q) &= -\int_\Omega q\,\mathrm{div}\,\mathbf{u}\,\mathrm{d}x \quad (10g)\end{aligned}$$

The variational formulation writes

(*FV*): *find* $(\boldsymbol{\chi},\mathbf{u},p)\in T\times V(\mathbf{u}_\Gamma)\times Q$ *such that*

$$\begin{aligned}t(\mathbf{u};\boldsymbol{\chi},\xi)+a_0(\boldsymbol{\chi},\xi)+b_1(\xi,\mathbf{u}) &= l(\xi), \quad \forall\xi\in T \quad (11a)\\ b_2(\boldsymbol{\chi},\mathbf{v})-c(\mathbf{u},\mathbf{v})-d(\mathbf{v},p) &= 0, \quad \forall\mathbf{v}\in V(0) \quad (11b)\\ -d(\mathbf{v},p) &= 0, \quad \forall q\in Q \quad (11c)\end{aligned}$$

The following properties are fundamental for solving the discrete problem and will admit a finite-dimensional counterpart in the next section.

Property 3 (*Skew-symmetry of t*). *For all* $\boldsymbol{\chi}\in T$ *and* $\mathbf{u}\in V(0)$ *such that* $\mathrm{div}\mathbf{u}=0$, *we have*

$$t(\mathbf{u};\boldsymbol{\chi},\boldsymbol{\chi}) = 0 \quad (12)$$

Proof. Integrating by part, we have

$$\begin{aligned}\int_\Omega((\mathbf{u}\cdot\nabla)\boldsymbol{\chi}):\boldsymbol{\chi}\,\mathrm{d}x &= -\int_\Omega((\mathbf{u}\cdot\nabla)\boldsymbol{\chi})\\ &\quad :\boldsymbol{\chi}\,\mathrm{d}x-\int_\Omega|\boldsymbol{\chi}|^2\mathrm{div}\,\mathbf{u}\,\mathrm{d}x+\int_{\partial\Omega}|\boldsymbol{\chi}|^2\mathbf{u}\cdot\mathbf{n}\,\mathrm{d}s\end{aligned}$$

and since $\mathbf{u}$ is divergence-free and vanishes on the boundary, the first term of the right-hand side of (10a), which gives the expression of $t(\mathbf{u};\boldsymbol{\chi},\boldsymbol{\chi})$, is zero. Next, from the skew-symmetry of $W(\mathbf{u})$ and the symmetry of $\boldsymbol{\chi}$ we have $(\boldsymbol{\chi}W(\mathbf{u})-W(\mathbf{u})\boldsymbol{\chi}):\boldsymbol{\chi}=0$. From property (9) we have $\kappa\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi},2D(\mathbf{u})\right):\boldsymbol{\chi}=0$ and hence get (12). □

Property 4 (*Positivity of* a_0). *For all* $\boldsymbol{\chi}\in L^2(\Omega)^{d\times d}_s$, *we have*

$$a_0(\boldsymbol{\chi},\boldsymbol{\chi})\geqslant 0 \quad (13)$$

Proof. From (10c) and the definition (7) of f, we have

$$a_0(\boldsymbol{\chi},\boldsymbol{\chi})=\frac{\eta_p}{a\lambda}\int_\Omega\left(I-\exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)\right):\boldsymbol{\chi}\,\mathrm{d}x$$

Note that $\boldsymbol{\sigma}:\boldsymbol{\tau}=\mathrm{tr}(\boldsymbol{\sigma}\boldsymbol{\tau})$ for all symmetric tensors $\boldsymbol{\sigma}$ and $\boldsymbol{\tau}$. Then, when $a\lambda\neq 0$:

$$a_0(\boldsymbol{\chi},\boldsymbol{\chi})=\frac{\eta_p}{a\lambda}\int_\Omega\mathrm{tr}\left(\left(I-\exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)\right)\boldsymbol{\chi}\right)\mathrm{d}x$$

As $\boldsymbol{\chi}$ and $\exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)$ share the same eigensystem, if μ is an eigenvalue of $\boldsymbol{\chi}$ then $g(\mu)$ is an eigenvalue of $\boldsymbol{\chi}-\exp\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right)\boldsymbol{\chi}$ where $g(\mu)=\mu-\exp(-a\lambda\mu/\eta_p)\mu$. An easy inspection of the variation of g shows that $g(\mu)\geqslant 0$ for all $\mu\in\mathbb{R}$. We then obtain the result (13). □

Remark 4. Corotational versus Gordon–Schowalter tensor derivatives (cont.)Multiplying (1) by $\boldsymbol{\tau}$, integrating and rearranging we get

$$\int_\Omega\frac{\mathscr{D}_a\boldsymbol{\tau}}{\mathscr{D}t}:\boldsymbol{\tau}\,\mathrm{d}x=\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\left(\int_\Omega|\boldsymbol{\tau}|^2\mathrm{d}x\right)-2a\int_\Omega\mathrm{tr}(\boldsymbol{\tau}D(\mathbf{u})\boldsymbol{\tau})\mathrm{d}x$$

When $a\neq 0$, there is no way of determining the sign of the last term on the right-hand side of the previous relation. The corresponding term $-a(\boldsymbol{\tau}D(\mathbf{u})+D(\mathbf{u})\boldsymbol{\tau})$, supported by $a\neq 0$ in the tensor derivative (1), acts as a source term in any tensorial transport problem involving the general Gordon–Schowalter derivative, such as the initial and conformation formulations of the Johnson–Segalman problem. This term is responsible for the observed exponential growth of the stress tensor $\boldsymbol{\tau}$ and the failure of numerical methods. There is no hope of obtaining either a skew-symmetry or a positivity property for such problems, and this is especially true for the popular upper-convected derivative ($a=1$). On the contrary, the log-conformation formulation involves a corotational tensor derivative of $\boldsymbol{\chi}$ in (8a), leading to the skew-symmetry of t and the positivity of a_0. Properties 3 and 4 are definitive advantages of the present log-conformation formulation over the initial and the conformation ones.

Note also that, when initially using a corotational derivative ($a = 0$), there is no need anymore for the log-conformation formulation. In that case, the new formulation presented here nicely reduces to the initial one with $\boldsymbol{\chi} = \boldsymbol{\tau}$ while all the extra nonlinear terms disappear.

3. Discretization and numerical resolution

3.1. Space approximation

The main idea here is to replace the three spaces T, X, Q by some finite dimensional counterparts $T_h \subset T, X_h \subset X, Q_h \subset Q$ in the variational formulation (11a)–(11c). When $\lambda = 0$ (a Newtonian fluid), a finite dimensional linear system is obtained, whose matrix has the following bloc structure:

$$\begin{pmatrix} A & B^T & 0 \\ B & -C & -D^T \\ 0 & -D & 0 \end{pmatrix}$$

This linear system is often called the *three field Stokes problem* and the approximation space pairs (T_h, X_h) and (X_h, Q_h) both need to satisfy a compatibility condition, known as the inf-sup or Babuška–Brezzi condition [8,23,9]. There are many possible choices for T_h, X_h and Q_h, each of them having some advantages and drawbacks. In this paper, we consider the following choice (see Fig. 1):

$$\begin{aligned} T_h &= \{\tau_h \in T; \tau_{h|K} \in P_1, \ \forall K \in \mathscr{T}_h\} \\ X_h &= \{\mathbf{v}_h \in (H^1(\Omega) \cap C^0(\overline{\Omega}))^d; \ \mathbf{v}_{h|K} \in P_2, \ \forall K \in \mathscr{T}_h\} \\ V_h(\mathbf{g}) &= X_h \cap V(\pi_h(\mathbf{g})) \\ Q_h &= \{q_h \in Q; q_{h|K} \in P_1, \ \forall K \in \mathscr{T}_h\} \end{aligned}$$

where $\mathscr{T}_h$ is a finite element mesh of the flow domain Ω and $h > 0$ denotes the largest edge length of the mesh. Notice that both T_h and Q_h contain *discontinuous* piecewise polynomials while X_h elements are continuous. The discontinuous approximation of stresses has two advantages. On one hand, $D(X_h) \subset T_h$ and then the compatibility condition between T_h and X_h is satisfied for the bi-linear form b_1, as showed by Fortin and Pierre [23] and Fortin and Fortin [22]. On the other hand, it allows an efficient treatment of the stress transport term by the discontinuous Galerkin method.

The discontinuous approximation of pressure also presents a major advantage. As $\operatorname{div}(X_h) \subset Q_h$, it leads to an exact *divergence-free* approximation of the velocity: for any field $\mathbf{v}_h \in X_h$ satisfying $d(\mathbf{v}_h, q_h) = 0$ for all $q_h \in Q_h$, we have $\operatorname{div} \mathbf{v}_h = 0$ point-wise everywhere in Ω. The pair (X_h, Q_h) is known as the Scott–Vogelius lowest-order finite element approximation [65]. This is a major advantage when dealing with a transport equation. The only drawback is that the pair (X_h, Q_h) does not satisfy the inf-sup condition for an arbitrary mesh. There exists a solution to this however: Arnold and Qin [3] proposed a macro element technique applied to the mesh [3] that allows satisfying the inf-sup condition: for any triangular finite element mesh, it is sufficient to split each triangle in three elements from its barycenter (see Fig. 1). This technique has been implemented for the present computations. Notice that the macro element technique extends to quadrilateral meshes [3] and to the three-dimensional case [70].

Let us review and discuss some other possible choices for the discretization of the problem.

- A very popular choice consists in using staggered finite difference grids for the velocity–pressure pair. With this choice the approximate velocity fields are also exactly divergence-free, the finite-difference implementation is simple and this approximation also extends in the mixed finite element context with the incompressible Raviart–Thomas finite element [24, chap. 3]. For the stress approximation, there are several possibilities: in [57, p. 17], the present author also used a staggered grid approximation (diagonal stress components at cell centers and off-diagonal components at cell corners in the bi-dimensional case). With this choice, the stress-velocity pair satisfies the inf-sup condition [57] and leads to a robust scheme that is able to reach solutions at high Weissenberg numbers (see [58,64]). In [18, p. 27], Fattal and Kupferman used a cell-centered approximation for all the stress components. As they pointed out [18, p. 29]:

  > *"This implies that the rotational components of the system may be sensitive to numerical instabilities. A natural remedy would have been to store the off-diagonal elements of the stress tensor at cell corners. This cannot be done in our framework as the log-conformation tensor is an entity whose tensorial nature is essential."*

  See also point 6 of the discussion, p. 36 of the same paper. With this choice, it is not clear whether the stress-velocity pair satisfies the inf-sup condition or not. Nevertheless, using this approximation, Fattal and Kupferman were able to compute solutions for some high Weissenberg numbers. A possible explanation for this success can be found in a paper by Baranger and Sandri [4], who have shown that the stress-velocity compatibility condition exhibited by [23] is only required when $\eta_v/(\eta_v + \eta_p) \ll 1$, i.e. in the absence of the pure viscous contribution. Otherwise, when η_v is not too small with respect to η_p, a much larger class of discretization schemes is allowed. This last condition was fulfilled by Fattal and Kupferman: all the computations presented by these authors used $\eta_v = \eta_p$.
- A classical choice is to use the Taylor-Hood $P_2 - P_1$ continuous element [68] (see also [9, p. 252]) for the velocity–pressure spaces (X_h, Q_h) together with piecewise linear and discontinuous stresses. This choice of stress approximation was first introduced by Fortin and Fortin [22]. Damanik [14, p. 25] recently used a variant for quadrilaterals meshes, combining Q_2 continuous velocities with P_1 discontinuous pressures [9, p. 216] and a quadratic continuous approximation for the stresses. As $\operatorname{div}(X_h) \not\subset Q_h$, for these elements, the approximate velocity field are not exactly divergence-free. This is the main difference with the Scott–Volelius element used in the present computation. We performed tests for both the Taylor–Hood approximation and the $Q_2 - P_{1,d}$ variant for quadrilaterals, together with $P_{1,d}$

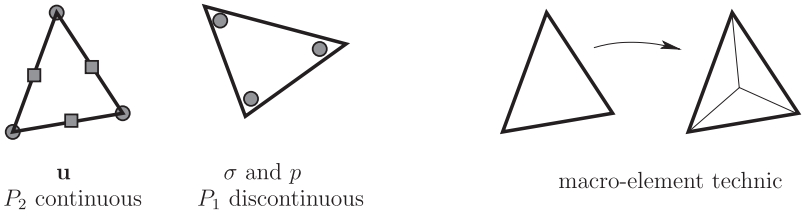

Fig. 1. Incompressible element for the three-field Stokes problem.

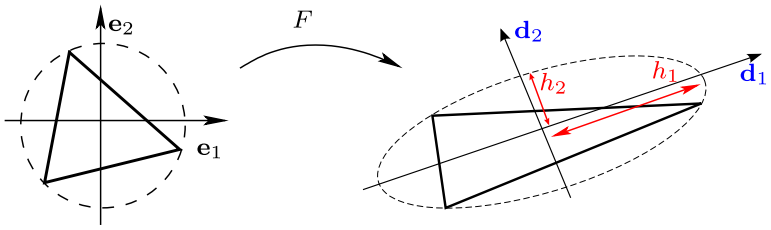

Fig. 2. Anisotropic adaptive mesh.

stresses, on the driven cavity problem and observed that the steady approximate solution presents some problems: for large dimensionless Weissenberg numbers (e.g. $We = 1$), the divergence $\mathrm{div}\,\mathbf{u}_h$ does not converge anymore to zero with mesh refinement ($h \to 0$). This could be due to the low regularity of the velocity field in the boundary layers of the cavity, especially at high Weissenberg numbers. This motivates the use the divergence-free approximation for the velocity–pressure pair.

- Another possible choice is to use constant and discontinuous P_0 stresses together with the divergence-free Scott–Vogelius velocity–pressure space pair. Notice that we have now $D(Xh) \not\subset Th$. This choice was first suggested by Mangoubi [44], Boyaval [6] and Boyaval et al. [7] in a theoretical paper, who were able to prove a nice stability property for the free energy. There were no numerical experiment with this combination. Moreover, we also performed some test on the driven cavity problem and observed that this choice is less robust and less precise than choosing piecewise linear and discontinuous stresses as done here.

This discretization review is far from complete and exhaustive: there are many other possible approaches and some of them have already been tested. Nevertheless, note that the exact divergence-free property for the approximation of the velocity field appears as an essential condition in such flow simulations. Both the staggered finite difference method and the Scott–Vogelius finite element one for the velocity–pressure pair satisfy this condition. The Scott–Vogelius finite element method presents two additional advantages compared to the finite difference one: (i) it allows the same approximation for all the stress components while satisfying the inf-sup condition and (ii) it is more flexible when dealing with complex geometries or adaptive meshes.

3.2. Approximation of the transport term

Let us turn to the discretization of the nonlinear stress transport term by the discontinuous Galerkin method. We introduce the following form, defined for all $\mathbf{u} \in H^1(\Omega)^d$ and $\boldsymbol{\chi}, \xi \in L^2(\Omega)_s^{d\times d}$ such that $\boldsymbol{\chi}_{|K}, \xi_{|K} \in H^1(K)_s^{d\times d}$ for all element $K \in \mathscr{T}_h$:

$$\begin{aligned} t_h(\mathbf{u};\boldsymbol{\chi},\xi) = {} & \lambda \sum_{K\in\mathscr{T}_h} \int_K ((\mathbf{u}\cdot\nabla)\boldsymbol{\chi}) : \xi \,\mathrm{d}x \\ & + \lambda \sum_{S\in\mathscr{S}_h^{(i)}} \int_S [[\boldsymbol{\chi}]] : \left(\frac{\theta}{2} |\mathbf{u}\cdot\mathbf{n}| [[\xi]] - (\mathbf{u}\cdot\mathbf{n})\{\{\xi\}\} \right) \mathrm{d}s \\ & + \lambda \int_{\partial\Omega} \max(0, -\mathbf{u}_\Gamma\cdot\mathbf{n}) \boldsymbol{\chi} : \xi \,\mathrm{d}s + \lambda \int_\Omega (\boldsymbol{\chi} W(\mathbf{u}) - W(\mathbf{u})\boldsymbol{\chi}) : \xi \,\mathrm{d}x \\ & + \int_\Omega \kappa \left(\frac{a\lambda}{\eta_p} \boldsymbol{\chi}, 2D(\mathbf{u}) \right) : \xi \,\mathrm{d}x \end{aligned} \tag{14}$$

where $\mathscr{S}_h^{(i)}$ denotes the set of internal sides of the mesh $\mathscr{T}_h$, $[[\xi]]$ is the jump of a piecewise discontinuous tensor across a side and $\{\{\xi\}\}$ is its average value [16,62]. The three first terms represent a discrete counterpart of the transport term that cannot be defined globally since the stress approximation is piecewise discontinuous. A term is weighted by a factor $\theta \geqslant 0$. Choosing $\theta = 0$ corresponds to the so-called centered flux approximation, while $\theta > 0$ is the *upwind* flux approximation. The case of $\theta = 1$, the most popular upwind discontinuous approximation scheme, is considered here. The upwind technique is known as an efficient approach to avoid spurious oscillations in the approximate solutions.

The discrete variational formulation of the problem writes:

$(FV)_h$: *find* $\boldsymbol{\chi}_h \in T_h, \mathbf{u}_h \in V_h(\mathbf{u}_\Gamma)$ and $p_h \in Q_h$ *such that*

$$\begin{aligned} t_h(\mathbf{u}_h;\boldsymbol{\chi}_h,\xi_h) + a_0(\boldsymbol{\chi}_h,\xi_h) + b_1(\xi_h,\mathbf{u}_h) &= l(\xi_h), \quad \forall \xi_h \in T_h \\ b_2(\boldsymbol{\chi}_h,\mathbf{v}_h) - c(\mathbf{u}_h,\mathbf{v}_h) - d(\mathbf{v}_h,p_h) &= 0, \quad \forall \mathbf{v}_h \in V_h(0) \\ -d(\mathbf{u}_h,q_h) &= 0, \quad \forall q_h \in Q_h \end{aligned}$$

where the continuous tri-linear form t has simply been replaced by t_h and the functional spaces by their finite dimensional counterpart.

Property 5 (*Discrete generalized skew-symmetry*). *For all* $\boldsymbol{\chi}_h \in T_h$ *and* $\mathbf{u}_h \in V_h(0)$ *such that* $\mathrm{div}\,\mathbf{u}_h = 0$, *we have*

$$t_h(\mathbf{u}_h;\boldsymbol{\chi}_h,\boldsymbol{\chi}_h) \geqslant 0 \tag{15}$$

with an equality when $\theta = 0$.

Proof. Integrating by part on an element K leads to

$$\begin{aligned} \int_K ((\mathbf{u}_h\cdot\nabla)\boldsymbol{\chi}_h) : \boldsymbol{\chi}_h \,\mathrm{d}x = {} & -\int_K \boldsymbol{\chi}_h \\ & : ((\mathbf{u}_h\cdot\nabla)\boldsymbol{\chi}_h)\mathrm{d}x + \int_K |\boldsymbol{\chi}_h|^2 (\mathbf{u}_h\cdot\mathbf{n})\mathrm{d}x \end{aligned}$$

since $\mathrm{div}\,\mathbf{u}_h = 0$. Then

$$\begin{aligned} \sum_{K\in\mathscr{T}_h} \int_K ((\mathbf{u}_h\cdot\nabla)\boldsymbol{\chi}_h) : \boldsymbol{\chi}_h \,\mathrm{d}x &= \frac{1}{2} \sum_{K\in\mathscr{T}_h} \int_{\partial K} |\boldsymbol{\chi}_h|^2 (\mathbf{u}_h\cdot\mathbf{n})\mathrm{d}s \\ &= \frac{1}{2} \sum_{S\in\mathscr{S}_h^{(i)}} \int_S [[|\boldsymbol{\chi}_h|^2]] (\mathbf{u}_h\cdot\mathbf{n})\mathrm{d}s \end{aligned}$$

Note that for any discontinuous scalar field ϕ and φ across a side S we have $[[\phi\varphi]] = [[\phi]]\{\{\psi\}\} + \{\{\phi\}\}[[\varphi]]$. Thus $[[|\boldsymbol{\chi}_h|^2]]/2 = [[\boldsymbol{\chi}_h]] : \{\{\boldsymbol{\chi}_h\}\}$ and

$$\sum_{K\in\mathscr{T}_h} \int_K ((\mathbf{u}_h\cdot\nabla)\boldsymbol{\chi}_h) : \boldsymbol{\chi}_h \,\mathrm{d}x = \sum_{S\in\mathscr{S}_h^{(i)}} \int_S [[\boldsymbol{\chi}_h]] : \{\{\boldsymbol{\chi}_h\}\} (\mathbf{u}_h\cdot\mathbf{n})\mathrm{d}s$$

Using $\mathbf{u}_h = 0$ on $\partial\Omega$ and dealing with the two last terms of (14) as in the proof of Property 3, we obtain:

$$t_h(\mathbf{u}_h;\boldsymbol{\chi}_h,\boldsymbol{\chi}_h) = \frac{\theta\lambda}{2} \sum_{S\in\mathscr{S}_h^{(i)}} \int_S |[[\boldsymbol{\chi}_h]]|^2 |\mathbf{u}_h\cdot\mathbf{n}| \,\mathrm{d}s$$

which completes the proof. □

Property 6 (*Discrete positivity*). *For all* $\boldsymbol{\chi}_h \in T_h$, *we have*

$$a_0(\boldsymbol{\chi}_h,\boldsymbol{\chi}_h) \geqslant 0 \tag{16}$$

Proof. As $T_h \subset L^2(\Omega)_s^{d\times d}$, this result is a direct consequence of Property 4. □

3.3. Newton method

Newton methods for the numerical resolution of steady viscoelastic flow problems started in the early eighties with the work of Crochet and Bézy [12] and Crochet and Keunings [13]. In 1992, Fortin and Zine [21] proposed a quasi-Newton variant, where the Jacobian matrix was approximated instead of being completely recomputed at each iteration. After a long time, the Newton method approach for viscoelastic fluid problems is now coming back. While previous works relied on some finite difference methods for computing the Jacobian matrix, in 2009, Howell [30] computed exactly the linearized steady Johnson–Segalman problem in its initial formulation. In 2010, Damanik et al. [15] and Damanik [14] turned to a Newton method for the time-dependent log-conformation formulation. They used a finite difference method for computing the Jacobian matrix. Let us quote a remark made in the 2009 paper by Kane et al. [35, p. 45] for the log-conformation formulation:

> "[...] *there is no hope to fully linearize the constitutive equations for the Newtons method without using some numerical tricks such as finite difference methods. The bulk of the computations relies indeed on the calculation, at each Gauss node, of the eigenvalues and eigenvectors of the conformation tensor which are not differentiable functions.*"

In the present paragraph, an exact expression of the derivatives for the fully linearized constitutive equation is presented for the log-conformation formulation, without any trick such as finite difference methods for computing the Jacobian matrix. The present Newton method directly treats the steady problem: by an obvious extension, it applies also to fully implicit time dependent simulations.

The discrete problem can be put in a compact form:
find $(\boldsymbol{\chi}_h, \mathbf{u}_h, p_h) \in T_h \times V_h(\mathbf{u}_\Gamma) \times Q_h$ *such that*

$$F(\lambda; (\boldsymbol{\chi}_h, \mathbf{u}_h, p_h)) = 0$$

where F is defined in variational form for all $(\xi_h, \mathbf{v}_h, q_h)$ by

$$\begin{aligned}\langle F(\lambda; (\boldsymbol{\chi}_h, \mathbf{u}_h, p_h)), (\xi, \mathbf{v}, q)\rangle = {} & t_h(\mathbf{u}_h; \boldsymbol{\chi}_h, \xi_h) + a_0(\boldsymbol{\chi}_h, \xi_h) + b_1(\xi_h, \mathbf{u}_h) \\ & - l(\xi_h) + b_2(\boldsymbol{\chi}_h, \mathbf{v}_h) - c(\mathbf{u}_h, \mathbf{v}_h) \\ & - d(\mathbf{v}_h, p_h) - d(\mathbf{u}_h, q_h)\end{aligned}$$

and where $\langle\cdot,\cdot\rangle$ stands for the duality product induced by the L^2 pivot space, i.e. $\langle\varphi,\phi\rangle = \int_\Omega \varphi\phi\,\mathrm{d}x$ for all φ, ϕ defined in Ω. The function F has two variables $\lambda \in \mathbb{R}^+$ and $U = (\boldsymbol{\chi}_h, \mathbf{u}_h, p_h) \in T_h \times X_h \times Q_h$. The λ variable will be used as a continuation parameter in the next section. The Newton method defines the sequence $\left(\boldsymbol{\chi}_h^{(k)}, \mathbf{u}_h^{(k)}, p^{(k)}\right)_{k\geqslant 0}$ by recurrence as:

- $k = 0$: let $\left(\boldsymbol{\chi}_h^{(0)}, \mathbf{u}_h^{(0)}, p^{(0)}\right) \in T_h \times V_h(\mathbf{u}_\Gamma) \times Q_h$ being given.
- $k \geqslant 0$: let $\left(\boldsymbol{\chi}_h^{(k-1)}, \mathbf{u}_h^{(k-1)}, p^{(k-1)}\right) \in T_h \times V_h(\mathbf{u}_\Gamma) \times Q_h$ being known. Find $(\delta\boldsymbol{\chi}_h, \delta\mathbf{u}_h, \delta p_h) \in T_h \times V_h(0) \times Q_h$ such that

$$\begin{aligned}&\frac{\partial F}{\partial U}\left(\lambda; \left(\boldsymbol{\chi}_h^{(k-1)}, \mathbf{u}_h^{(k-1)}, p_h^{(k-1)}\right)\right) \cdot (\delta\boldsymbol{\chi}_h, \delta\mathbf{u}_h, \delta p_h) \\ &\quad = -F\left(\lambda, \left(\boldsymbol{\chi}_h^{(k-1)}, \mathbf{u}_h^{(k-1)}, p_h^{(k-1)}\right)\right)\end{aligned}$$

and then defines

$$\begin{aligned}\boldsymbol{\chi}_h^{(k)} = \boldsymbol{\chi}_h^{(k-1)} + \delta\boldsymbol{\chi}_h, \quad \mathbf{u}_h^{(k)} = \mathbf{u}_h^{(k-1)} + \delta\mathbf{u}_h \quad \text{and} \quad p_h^{(k)} \\ = p_h^{(k-1)} + \delta p_h\end{aligned}$$

At each step $k \geqslant 0$, this algorithm solves a linear subproblem involving the Jacobian $\frac{\partial F}{\partial U}$. The Newton method has only local convergence properties, i.e. the initial value should be close enough to the solution. In order to circumvent this limitation, a globalized Newton variant is used here. It is based on a damping strategy, as described and implemented in the `Rheolef` free software FEM library [61]. Note the absolute value function that appears in the upwind term in the definition (14) of t_h:

$$s_h(\mathbf{u}_h; \boldsymbol{\chi}_h, \xi_h) = \frac{1}{2}\sum_{S\in\mathscr{S}_h^{(i)}} \int_S [[\boldsymbol{\chi}_h]] : [[\xi_h]] |\mathbf{u}_h \cdot \mathbf{n}| \mathrm{d}s$$

This term is not differentiable with respect to $\mathbf{u}_h$ and neither is t_h nor F. Nevertheless, the absolute value is convex and some tools from subdifferential calculus can be used to overcome this new difficulty. Let us introduce the multi-valued sign function:

$$\mathrm{sgn}(x) = \begin{cases} \{1\} & \text{when } x > 0 \\ [-1,1] & \text{when } x = 0 \\ \{-1\} & \text{when } x < 0 \end{cases}$$

Then, the subdifferential of the absolute value function is $\mathrm{sgn}(x)$ and for all $\delta\mathbf{w}_h, \mathbf{w}_h, \mathbf{u}_h, \mathbf{v}_h \in \mathbf{X}_h$, we introduce a generalization of the partial derivative:

$$\frac{\partial s_h}{\partial \mathbf{u}_h}(\mathbf{u}_h; \boldsymbol{\chi}_h, \xi_h) \cdot (\delta\mathbf{u}_h) = \frac{1}{2}\sum_{S\in\mathscr{S}_h^{(i)}} \int_S [[\boldsymbol{\chi}_h]] : [[\xi_h]] \mathrm{sgn}(\mathbf{u}_h \cdot \mathbf{n}) \delta\mathbf{u}_h \cdot \mathbf{n}\,\mathrm{d}s$$

The Jacobian $\frac{\partial F}{\partial U}$ can thus be defined as a multi-valued subgradient set. In 1993, Qi and Sun [54] showed how the Newton method extends to this case: any element that belongs to this set represents a valid direction for the Newton correction step. Such a *non-smooth* Newton method was successfully implemented for the steady Navier–Stokes equations with the discontinuous Galerkin method and upwind scheme [62].

The multi-valued Jacobian $\frac{\partial F}{\partial U}$ is defined for all $(\boldsymbol{\chi}_h, \mathbf{u}_h, p_h) \in T_h \times V(\mathbf{u}_\Gamma) \times Q_h$, and $(\delta\boldsymbol{\chi}_h, \delta\mathbf{u}_h, \delta p_h) \in T_h \times V(0) \times Q_h$ $(\xi_h, \mathbf{v}_h, q_h)$ by

$$\begin{aligned}&\left\langle \frac{\partial F}{\partial U}(\lambda, (\boldsymbol{\chi}_h, \mathbf{u}_h, p_h)) \cdot (\delta\boldsymbol{\chi}_h, \delta\mathbf{u}_h, \delta p_h), (\xi_h, \mathbf{v}_h, q_h) \right\rangle \\ &\quad = a_1(\boldsymbol{\chi}_h, \mathbf{u}_h; \delta\boldsymbol{\chi}_h, \xi_h) + b_{11}(\boldsymbol{\chi}_h, \mathbf{u}_h; \xi_h, \delta\mathbf{u}_h) + b_{12}(\boldsymbol{\chi}_h; \delta\boldsymbol{\chi}_h, \mathbf{v}_h) \\ &\qquad - c(\delta\mathbf{u}_h, \mathbf{v}_h) - d(\mathbf{v}_h, \delta p_h) - d(\delta\mathbf{u}, q)\end{aligned}$$

where

$$\begin{aligned}a_1(\boldsymbol{\chi}, \mathbf{u}; \delta\boldsymbol{\chi}, \xi) = {} & \frac{\partial t_h}{\partial \boldsymbol{\chi}}(\mathbf{u}; \boldsymbol{\chi}, \xi) \cdot (\delta\boldsymbol{\chi}) + \frac{\partial a}{\partial \boldsymbol{\chi}}(\boldsymbol{\chi}, \xi) \cdot (\delta\boldsymbol{\chi}) \\ = {} & \lambda t_{0,h}(\mathbf{u}; \delta\boldsymbol{\chi}, \xi) + s_h(\mathbf{u}; \delta\boldsymbol{\chi}, \xi) + \int_{\partial\Omega} \delta\boldsymbol{\chi} : \xi \max(0, -\mathbf{u}_\Gamma \cdot \mathbf{n}) \mathrm{d}s \\ & + \frac{a\lambda}{\eta_p}\int_\Omega \left(\frac{\partial \kappa}{\partial \boldsymbol{\beta}}\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}, 2D(\mathbf{u})\right) : \delta\boldsymbol{\chi}\right) : \xi \mathrm{d}x \\ & + \int_\Omega \delta\boldsymbol{\chi} : \xi \mathrm{d}x + \int_\Omega \left(\exp'\left(-\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right) : \delta\boldsymbol{\chi}\right) : \xi \mathrm{d}x\end{aligned}$$

$$\begin{aligned}b_{11}(\boldsymbol{\chi}, \mathbf{u}; \xi, \delta\mathbf{u}) = {} & \lambda t_{0,h}(\delta\mathbf{u}; \boldsymbol{\chi}, \xi) + \lambda \frac{\partial s_h}{\partial \mathbf{u}}(\mathbf{u}; \boldsymbol{\chi}, \xi) \cdot (\delta\mathbf{u}) + b_1(\xi, \delta\mathbf{u}) \\ & + \int_\Omega \kappa\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}, 2D(\mathbf{u})\right) : \xi \mathrm{d}x\end{aligned}$$

$$\begin{aligned}b_{12}(\boldsymbol{\chi}; \delta\boldsymbol{\chi}, \mathbf{v}) = {} & \frac{\partial b_2}{\partial \boldsymbol{\chi}}(\boldsymbol{\chi}, \mathbf{v}) \cdot (\delta\boldsymbol{\chi}) \\ = {} & -\int_\Omega \delta\boldsymbol{\chi} : D(\mathbf{v}) \mathrm{d}x - \int_\Omega \left(\exp'\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}\right) : \delta\boldsymbol{\chi}\right) : \xi \mathrm{d}x\end{aligned}$$

and where $t_{0,h}$ represents

$$t_{0,h}(\mathbf{u}_h;\boldsymbol{\chi}_h,\boldsymbol{\xi}_h)=\sum_{K\in\mathscr{T}_h}\int_K((\mathbf{u}_h\cdot\nabla)\boldsymbol{\chi}_h):\boldsymbol{\xi}_h\,\mathrm{d}x-\sum_{S\in\mathscr{S}_h^{(i)}}\int_S[[\boldsymbol{\chi}_h]]:\{\{\boldsymbol{\xi}_h\}\}(\mathbf{u}_h\cdot\mathbf{n})\mathrm{d}s+\int_\Omega(\boldsymbol{\chi}_h W(\mathbf{u}_h)-W(\mathbf{u}_h)\boldsymbol{\chi}_h):\boldsymbol{\xi}_h\,\mathrm{d}x$$

Here $\exp'(\boldsymbol{\chi}):\delta\boldsymbol{\chi}$ denotes the differential at $\delta\boldsymbol{\chi}$ of the exponential of a $d\times d$ matrix. Recall that when $\boldsymbol{\chi}$ and $\delta\boldsymbol{\chi}$ commute, then $\exp'(\boldsymbol{\chi}):\delta\boldsymbol{\chi}=\exp(\boldsymbol{\chi})\delta\boldsymbol{\chi}$ while the general case is more complex. Also, $\frac{\partial\kappa}{\partial\boldsymbol{\beta}}$ involves the derivatives of eigenvalues and eigenvectors of a $d\times d$ matrix with respect to the matrix coefficients. Note that since κ is linear with respect to its second variable $\boldsymbol{\gamma}$ we have $\frac{\partial\kappa}{\partial\boldsymbol{\gamma}}(\boldsymbol{\beta},\boldsymbol{\gamma}):\delta\boldsymbol{\gamma}=\kappa(\boldsymbol{\beta},\delta\boldsymbol{\gamma})$. A key point of the present implementation of the Newton method is the exact computation of both $\exp(\boldsymbol{\chi})$, $\exp'(\boldsymbol{\chi})$, $\kappa(\boldsymbol{\beta},\boldsymbol{\gamma})$ and $\frac{\partial\kappa}{\partial\boldsymbol{\beta}}(\boldsymbol{\beta},\boldsymbol{\gamma})$. Indeed, these computations can be performed explicitly: Rouvière [56, p. 297] proposes some tools for the derivation of the exponential of a matrix, while Hairer and Wanner [27, p. 102] give some formula for computing the derivative of eigenvalues and Magnus and Neudecker [43, p. 177] also present the derivative of eigenvectors with respect to the coefficient of a matrix. Kane et al. [35, p. 50] presented some explicit expressions for $\exp(\boldsymbol{\chi})$ and $\exp'(\boldsymbol{\chi})$ based on the `mapple` software: these expressions contain some errors and we provide the correct expressions in Appendix A.1 for completeness. Appendix A.2 shows that κ is continuous and differentiable with respect to the matrix coefficients and details for the first time the computation of these quantities. Finally, the integrals involving nonlinear expressions are evaluated by a Gauss quadrature formula with six interior nodes in a triangle: this formula is exact for polynomials with degree lower or equal to four. Numerical experiments with higher order of quadrature do not show significant changes in the numerical results.

3.4. Euler–Newton continuation algorithm

The aim of the Euler–Newton continuation algorithm is to start from a previously computed solution at a smaller λ (or dimensionless Weissenberg number), perform a prediction using an Euler scheme (as if λ was a pseudo-time) and then perform corrections using the Newton method. This approach allows to reach high values of the λ parameter efficiently.

Let us denote $U=(\boldsymbol{\chi}_h,\mathbf{u}_h,p_h)\in T_h\times X_h\times Q_h$, such that the approximate nonlinear problem can be written in a concise form $F(\lambda,U)=0$. In this paragraph, we introduce an Euler–Newton continuation algorithm (see e.g. [52, p. 176] or [66]) that consists of two nested loops:

Algorithm 1 (*continuation*).

- $n=0$: Let (λ_0,U_0) be given. Compute
$$\dot U_0=-\left(\frac{\partial F}{\partial U}(\lambda_0,U_0)\right)^{-1}\frac{\partial F}{\partial\lambda}(\lambda_0,U_0)$$
- $n\geqslant 0$: Assume (λ_n,U_n) and $\dot U_n$ are known.
  (1) First choose a step $\Delta\lambda_n$ and set $\lambda_{n+1}=\lambda_n+\Delta\lambda_n$.
  (2) Then, perform an Euler prediction by computing
$$w_0=U_n-\Delta\lambda_n\left(\frac{\partial F}{\partial U}(\lambda_n,U_n)\right)^{-1}\frac{\partial F}{\partial\lambda}(\lambda_n,U_n)$$
  (3) Then, perform a Newton correction step: for all $k\geqslant 0$, with W_k being known, compute
$$W_{k+1}=W_k-\left(\frac{\partial F}{\partial U}(\lambda_{n+1},W_k)\right)^{-1}F(\lambda_{n+1},W_k)$$

Once the correction loop has converged, set $U_{n+1}=W_\infty$.

(4) Finally, compute
$$\dot U_{n+1}=-\left(\frac{\partial F}{\partial U}(\lambda_{n+1},U_{n+1})\right)^{-1}\frac{\partial F}{\partial\lambda}(\lambda_{n+1},U_{n+1})$$

The step $\Delta\lambda_n$ can either be fixed or chosen by adjusting the contraction ratio of the Newton method [52,66]. The previous algorithm requires the computation of $\frac{\partial F}{\partial\lambda}$, which is given by:

$$\begin{aligned}&\left\langle\frac{\partial F}{\partial\lambda}(\lambda,(\boldsymbol{\chi}_h,\mathbf{u}_h,p_h)),(\boldsymbol{\xi}_h,\mathbf{v}_h,q_h)\right\rangle\\ =\;& t_{0,h}(\mathbf{u}_h;\boldsymbol{\chi}_h,\boldsymbol{\xi}_h)+s_h(\mathbf{u}_h;\boldsymbol{\chi}_h,\boldsymbol{\xi}_h)+\int_{\partial\Omega}\max(0,-\mathbf{u}_{\Gamma^-}\cdot\mathbf{n})\boldsymbol{\chi}_h:\boldsymbol{\xi}_h\,\mathrm{d}s\\ &+\frac{a}{\eta_p}\int_\Omega\left(\frac{\partial\kappa}{\partial\boldsymbol{\chi}}\left(\frac{a\lambda}{\eta_p}\boldsymbol{\chi}_h,2D(\mathbf{u}_h)\right):\boldsymbol{\chi}_h\right):\boldsymbol{\xi}_h\,\mathrm{d}x\\ &-\frac{a}{\eta_p}\int_\Omega\frac{\partial f}{\partial\mu}\left(\frac{a\lambda}{\eta_p},-\boldsymbol{\chi}_h\right):\boldsymbol{\xi}_h\mathrm{d}x-\frac{a}{\eta_p}\int_\Omega\frac{\partial f}{\partial\mu}\left(\frac{a\lambda}{\eta_p},\boldsymbol{\chi}_h\right):D(\mathbf{v}_h)\mathrm{d}x\end{aligned}$$

where

$$\frac{\partial f}{\partial\mu}(\mu,\boldsymbol{\chi})=\begin{cases}\frac{\boldsymbol{\chi}^2}{2} & \text{when } \mu=0\\ -\frac{1}{\mu^2}(\exp(\mu\boldsymbol{\chi})-I)+\frac{1}{\mu}\exp'(\mu\boldsymbol{\chi}):\boldsymbol{\chi} & \text{otherwise}\end{cases}$$

3.5. Automatic adaptive mesh

The anisotropic auto-adaptive mesh feature available in the `Rheolef` free software FEM library [61] has been used in this paper. This feature is based on the free software `bamg` bidimensional anisotropic mesh generator developed by Hecht [29]. Let us first summarize the principle of the adaptive mesh procedure. Let $\mathscr{T}_h$ be an initial mesh and $U=(\boldsymbol{\chi}_h,\mathbf{u}_h,p_h)\in T_h\times X_h\times Q_h$ be the solution of the discrete nonlinear problem $(FV)_h$ associated to the mesh $\mathscr{T}_h$. Let ϕ be a *governing field* to be suitably chosen from the solution U. For a piecewise linear interpolation of ϕ, the interpolation error in the unitary direction $\mathbf{d}\in IR^2$ is estimated at any element $K\in\mathscr{T}_h$ by:

$$e_{K,\mathbf{d}}=h_{K,\mathbf{d}}^2\left|\frac{\partial^2\phi}{\partial\mathbf{d}^2}\right|\quad\text{in } K$$

where $h_{K,\mathbf{d}}$ is the length of K in the $\mathbf{d}$ direction and

$$\frac{\partial^2\phi}{\partial\mathbf{d}^2}=\mathbf{d}^T\mathbf{H}(\phi)\mathbf{d}\quad\text{and}\quad\mathbf{H}(\phi)=\begin{pmatrix}\frac{\partial^2\phi}{\partial x^2} & \frac{\partial^2\phi}{\partial x\partial y}\\ \frac{\partial^2\phi}{\partial x\partial y} & \frac{\partial^2\phi}{\partial y^2}\end{pmatrix}$$

Here $\mathbf{H}(\phi)$ denotes the Hessian of ϕ. One possible way of adapting the mesh to the computation of ϕ is to equi-distribute this error, i.e. to make it constant over all triangles and in all directions. Let λ_1,λ_2 be the eigenvalues of $\mathbf{H}(\phi)$ and $\mathbf{d}_1$ and $\mathbf{d}_2$ the associated eigenvectors:

$$\frac{\partial^2\phi}{\partial\mathbf{d}_1^2}=\lambda_1\quad\text{and}\quad\frac{\partial^2\phi}{\partial\mathbf{d}_2^2}=\lambda_2$$

The error $e_{K,\mathbf{d}}$ is independent of $\mathbf{d}$ and K when there exists a constant $e_0>0$ independent of K such that $e_{K,\mathbf{d}_1}=e_{K,\mathbf{d}_2}=e_0$. This writes equivalently

$$h_{K,\mathbf{d}_1}^2|\lambda_1|=h_{K,\mathbf{d}_2}^2|\lambda_2|=e_0,\quad\forall K\in\mathscr{T}_h$$

The constant e_0 represents a global surface density factor. The adapted mesh generator tries to shrink elements in all directions with a factor $\sqrt{e_0}$, hence all the bidimensional element areas are reduced by a factor e_0. Suppose that $\mathbf{H}(\phi)$ is non-singular, i.e. $\lambda_1\lambda_2\neq 0$. The constant e_0 being known, our aim is now to build

triangles of length h_i in the $\mathbf{d}_i$ direction with $h_i = \sqrt{e_0/|\lambda_i|}$, $i = 1, 2$. Such a triangle has no privileged direction in a metric such that the two $h_i\mathbf{d}_i$ vectors, $i = 1, 2$, have the same norm. Thus, let us introduce the metric $\mathbf{M}(\phi)$ tensor that have the same eigenvectors as $\mathbf{H}(\phi)$ and $|\lambda_i|$, $i = 1, 2$ as eigenvalues. The induced norm $\|\cdot\|_M$ satisfies

$$\|h_i\mathbf{d}_i\|_M = h_K\sqrt{\mathbf{d}_i^T\mathbf{M}(\phi)\mathbf{d}_i} = \sqrt{e_0},\ i = 1, 2$$

It then suffices to build an isotropic mesh in the Riemann space associated to the metric $\mathbf{M}(\phi)$. In the Euclidean space, this mesh locally shrinks with a factor h_i in the $\mathbf{d}_i$ direction.

It remains to choose a suitable ϕ governing field. Several numerical experiments have motivated the use of a combination of a *free energy* and of a *viscous dissipation* term:

$$\phi = \lambda \mathrm{tr}\left(f\left(\frac{a\lambda}{\eta_p}, \boldsymbol{\chi}\right)\right) + \eta_v|D(\mathbf{u})|^2$$

where f is defined in (7). As both $\boldsymbol{\chi}_h$ and $D(\mathbf{u}_h)$ are linear and piecewise discontinuous, the governing field ϕ is also approximated by a piecewise linear and discontinuous function ϕ_h. Obtaining its hessian thus requires a L^2 projection and two derivations. The L^2 projection of ϕ_h gives a piecewise linear and continuous approximation of ϕ, denoted by $\tilde{\phi}_h$. The discrete Hessian $\mathbf{H}_h$ is obtained from $\tilde{\phi}_h$ by computing first its discrete gradient $\mathbf{g}_h$, continuous and piecewise linear, from the variational formula

$$\int_\Omega \mathbf{g}_h \cdot \mathbf{v}_h \mathrm{d}x = \int_\Omega \nabla\tilde{\phi}_h \cdot \mathbf{v}_h \,\mathrm{d}x$$

where $\mathbf{v}_h$ is any continuous piecewise linear vector. Then $\mathbf{H}_h$ is obtained from the variational formula

$$\int_\Omega \mathbf{H}_h : \xi_h \mathrm{d}x = \int_\Omega D(\mathbf{g}_h) : \xi_h \,\mathrm{d}x$$

where ξ_h is any continuous piecewise linear tensor. Solving a problem using such an automatic adaptive mesh is an iterative process, which involves three main steps :

1. Starting from an initial mesh $\mathscr{T}_h$, solve the problem using the Newton method. Let U be the corresponding solution associated to the mesh $\mathscr{T}_h$.
2. From U, compute the governing field ϕ_h.
3. From the governing field ϕ_h, defined on the mesh $\mathscr{T}_h$, generate a totally new mesh, denoted by $\mathscr{T}_h^{(1)}$.

Then, $\mathscr{T}_h^{(1)}$ is used to solve the problem, and so on, until convergence of both the mesh and its associated solution. Hence, the final adapted mesh minimizes the interpolation error for the governing field. This choice of the governing field has been found able to catch accurately the boundary layers and secondary vortices, as showed in the next section.

4. Tests on the smoothed driven cavity flow

For the purpose of comparison with previous studies [18,28], we consider the steady bi-dimensional smoothed driven cavity benchmark problem with $\Omega =]0, L[^2$ with $L > 0$. The boundary velocity $\mathbf{u}_\Gamma$ is zero except on the top boundary $\{y = L\}$ where $\mathbf{u}_\Gamma(x, L) = (16Ux^2(L-x)^2/L^2, 0)$ with $U > 0$. The fluid parameters are $a = 1$ and $\eta_p = \eta_v$ and the dimensionless Weissenberg number is $We = \lambda U/L$. Despite its simple geometry, this is a very difficult problem: to our knowledge, most numerical methods based on the initial formulation failed for $We \leqslant 0.1$. Due to singularities near corners between the lid and the sidewalls, the lid-driven cavity may encounter the high We number problem or present many challenges in the form of singularity points in the flow. Therefore, it is known as a highly stringent test problem for numerical methods.

Fig. 3a plots the residual term $\|F(\lambda, U)\|_{L^2}$ versus the iteration number n_{iter} of the damped Newton method. The loop is initialized from the Newtonian solution, associated to $We = 0$ and the solution is then computed *directly* for a specific We. Observe the quadratic convergence in log scale, up to $We = 0.5$. The algorithm stops when the machine precision is reached. For the last test case, $We = 0.6$ and the damped strategy still allows the convergence of the algorithm at the price of a larger number of iteration and the loss of the quadratic convergence. In that case, the Euler–Newton continuation algorithm, which restarts from a previous computed solution at a smaller We, is more efficient. We used a Weissenberg number increment of $\Delta We = 0.1$ in the present computations. Moreover, the Euler–Newton continuation algorithm is more robust and allows reaching solutions at much higher Weissenberg numbers. With this algorithm, each step uses about five resolutions of the Jacobian linear system and the solution at $We = 1$ can be reached from the solution at $We = 0$ with about fifty resolutions of linear systems. This convergence property is *mesh-invariant*, as showed by Fig. 3b (for the mesh-invariance property of nonlinear algorithms, see [61, chap. 8]). This algorithm provides also all the intermediate solutions at $We = 0.1, \ldots, 0.9$. Such very efficient viscoelastic computations for large Weissenberg numbers can be compared with the thousands of steps involving linear systems and required by all the current time-dependent approaches for reaching the steady solution [18,28,35,15]. Moreover, the number of time steps required by these methods increases with the Weissenberg number, and also with the mesh refinement, when non-fully implicit time schemes are used.

The current implementation uses the `Rheolef` free software FEM library [61,62] that is available as a standard package under

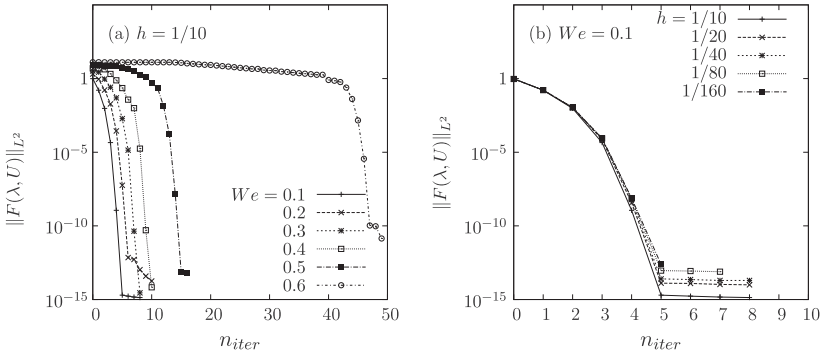

Fig. 3. Convergence of the damped Newton algorithm, starting from the solution at $We = 0$ for the Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$: (a) versus We for an uniform mesh with $h = 1/10$ and (b) versus mesh refinement h for $We = 0.1$.

Table 1
The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Maximal and critical Weissenberg number versus mesh refinement: (left) uniform mesh; (right) adapted mesh.

h	$We_{\max}$
1/20	3.74
1/40	2.19
1/80	2.21
1/160	2.08

$h_{\min}$	$We_{\max}$	We_c
10^{-2}	3.13	2.33
5×10^{-3}	2.78	2.03

the Debian and Ubuntu GNU/Linux systems. The Jacobian matrix $\frac{\partial F}{\partial U}$ is large and sparse: the linear system is solved with a direct method with the help of the parallel and massively distributed memory linear solver `mumps` [2] together with the `scotch` [53] ordering algorithm for minimizing the fill-in of the sparse matrix. For the largest meshes ($h = 1/80, 1/160$ and the adaptive meshes), the computations are run on 32 processors on a BullX DLC supercomputer (Bull Newsca) composed of nodes having two Intel Sandy-Bridge processors and connected to a FDR infiniband non-blocking low latency network. The computation of a whole branch of solutions is performed in less than one hour of real time.

Table 1 (left) groups the maximal Weissenberg number, denoted as $We_{\max}$, reached by the Euler–Newton continuation algorithm: for higher We, the algorithm stops, due to a singular solution where the Jacobian matrix $\frac{\partial F}{\partial U}$ is non-invertible. When such a singularity occurs in the continuation algorithm, the Weissenberg number increment is divided by two and the iteration restarted. The Weissenberg number increment is limited to 10^{-7}. When using an adaptive mesh, we observe that the Euler–Newton continuation algorithm is able to jump behind the singular point and to continue its progression on a branch of solutions until a second singular point is reached near $We = 3$. In that case, the determinant of the Jacobian matrix exhibits a change of sign after the first singular point (see e.g. [66]). The Weissenberg number where this change of sign occurs is denoted as We_c in Table 1 (right). Note that for the finest uniform mesh ($h = 1/160$), the singular point is $We_c = 2.08$ and for the finest adapted one, $We_c = 2.03$. Then We_c seems to become mesh insensitive with mesh refinement. A deeper analysis of singular points is planed as future work. It requires more advanced tools than the simple continuation Euler–Newton algorithm. The first and second singular points could be e.g. bifurcation points associated to a loss of stationarity of the solution: for larger We, the solution becomes non-stationary, as suggested by time-dependent simulations (see e.g. [18], Fig. 8). Pakdel, Spiegelberg and McKinley also showed by experimental observations [50] for the motion of viscoelastic fluids in the lid-driven cavity geometry that, at large Weissenberg numbers, the fluid motion becomes unstable and a three-dimensional flow develops.

Fig. 4 plots the L^2 norm of the velocity and the log-conformation tensor: observe the good convergence of these quantities with mesh refinement. There are five uniform meshes from $h = 1/20$ to $h = 1/160$ and two adapted meshes with $h_{\min} = 10^{-2}$ and 5×10^{-3}.

Figs. 5 and 6 show the adapted meshes and stream isovalues of the function for $We = 1, 2$ and 3. The stream function φ is defined as the unique function that satisfies $-\Delta\varphi = \partial_y u_x - \partial_x u_y$ in Ω with $\varphi = 0$ on $\partial\Omega$. Ten negative and fifteen positive equi-spaced isolines are represented on each plot. Observe that viscoelastic effects break the symmetry observed for the velocity field of cavity flows of viscous Newtonian fluids at zero Reynolds number. At low Weissenberg number, the flow remains two-dimensional but the center of the primary recirculating vortex in the cavity shifts progressively upstream (left). These results are in qualitative agreement with experimental results [50]. Note also that inertia in viscous Newtonian fluids (i.e. $We = 0$ and $Re > 0$) has opposite effects (see e.g. [62, part 2]): the center of the primary recirculating vortex in the cavity shifts downstream (right).

Furthermore, our results are compared quantitatively to previous studies: the locations of the primary vortex center for $We = 1$ are listed in Table 2. Our results are compared with the results of time-dependent flow using both a finite element method [51] and a lattice Boltzmann method [67]. It is found that the results of our steady computations are consistent with those of these authors.

Figs. 7 and 8 show zooms on the left and right secondary vortices. While the main vortex moves from left to right when We increases, the left vortex grows and the right one decreases in activity. Note that this is also in opposition with inertial effects for viscous Newtonian fluids.

Table 3 groups the main and secondary vortices center position and activity for $We = 1, 2$ and 3. These values are provided for future cross validation purpose. One can observe that the main vortex goes left and decreases in activity while the center of the left secondary vortex goes up and right and its activity increases. Also, the center of the right secondary vortex roughly stays in place while its activity remains constant. Finally, Fig. 9 plots the cut of the first component u_x of the velocity and the χ_{xx} component along the $x = 1/2$ vertical axis, as computed with the adaptive mesh. The computations with the finest uniform mesh ($h = 1/160$) give similar results (not shown as the difference is not graphically perceptible). Observe the excellent quantitative agreement with

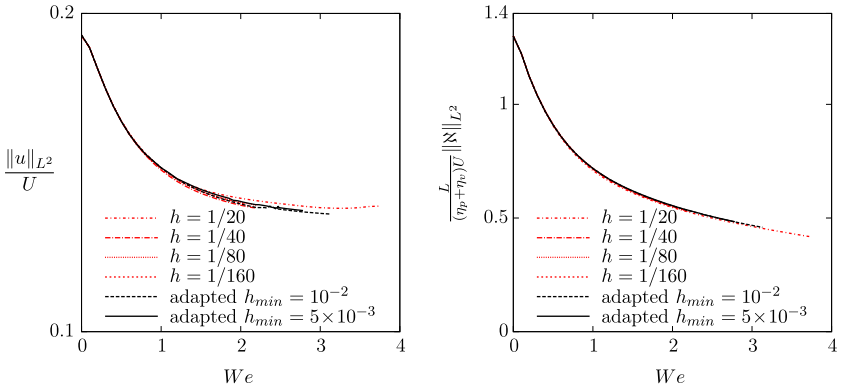

Fig. 4. The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Norm of the solution versus We.

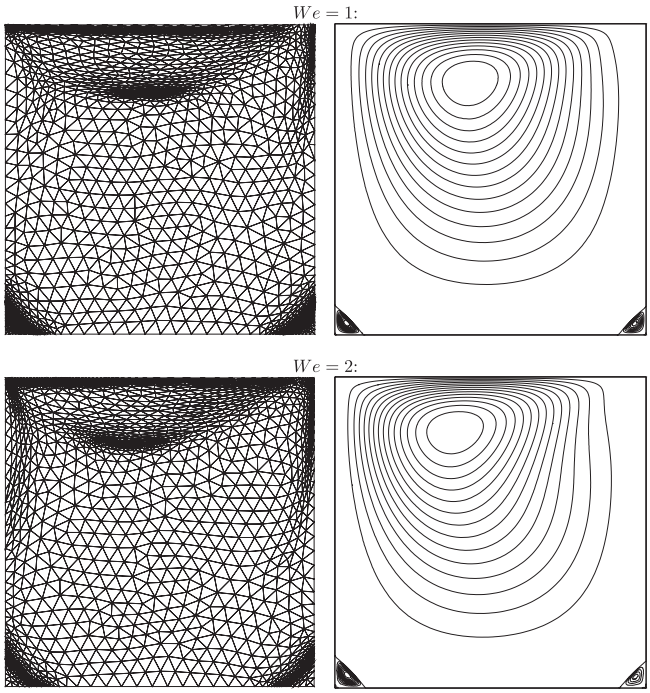

Fig. 5. The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Adapted mesh and stream function isovalues for $We = 1$ and 2.

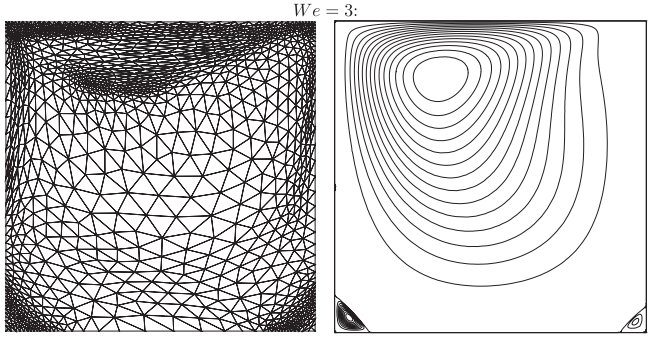

Fig. 6. The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Adapted mesh and stream function isovalues for $We = 3$.

Table 2
The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Comparison of the dimensionless main vortex center position for $We = 1$ with others authors.

x_m	y_m	Reference
0.439	0.816	Pan et al. [51]
0.433	0.803	Su et al. [67]
0.429	0.818	Present

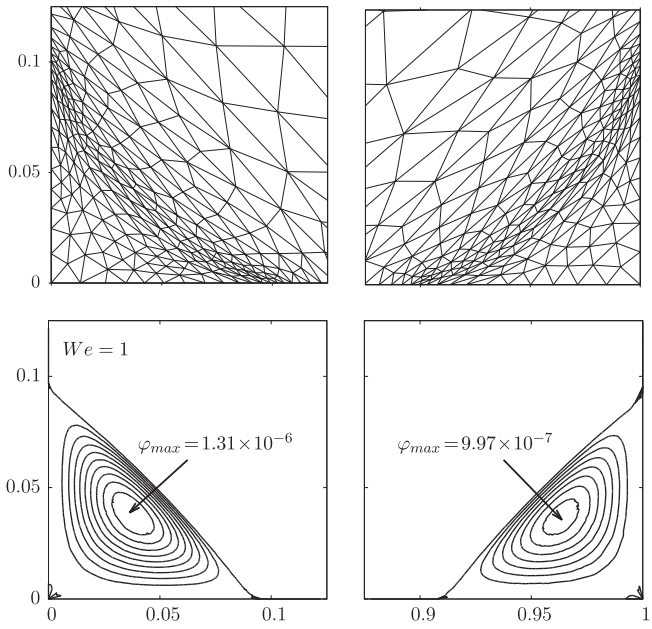

Fig. 7. The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Zoom around vortex for $We = 1$.

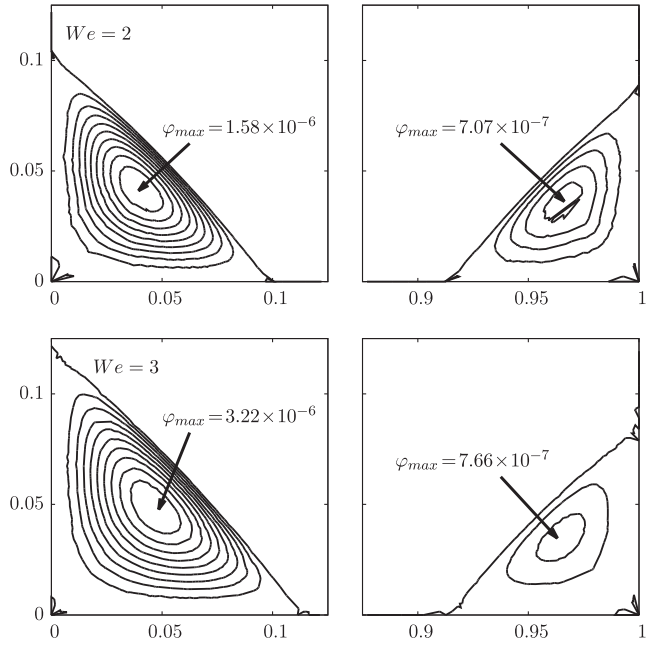

Fig. 8. The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Zoom around vortex for $We = 2$ and 3.

Table 3
The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Dimensionless main and secondary vortices center position and activity for $We = 1$, 2 and 3.

We	Main			Left			Right		
	x_m	y_m	$\varphi_{\min}$	x_l	y_l	$\varphi_{\max}$	x_r	y_r	$\varphi_{\max}$
1	0.429	0.818	−0.0619	0.0364	0.0388	1.31×10^{-6}	0.9637	0.0355	9.97×10^{-7}
2	0.386	0.828	−0.0555	0.0394	0.0411	1.58×10^{-6}	0.9663	0.0363	7.07×10^{-7}
3	0.335	0.824	−0.0531	0.0485	0.0477	3.22×10^{-6}	0.9631	0.0347	7.66×10^{-7}

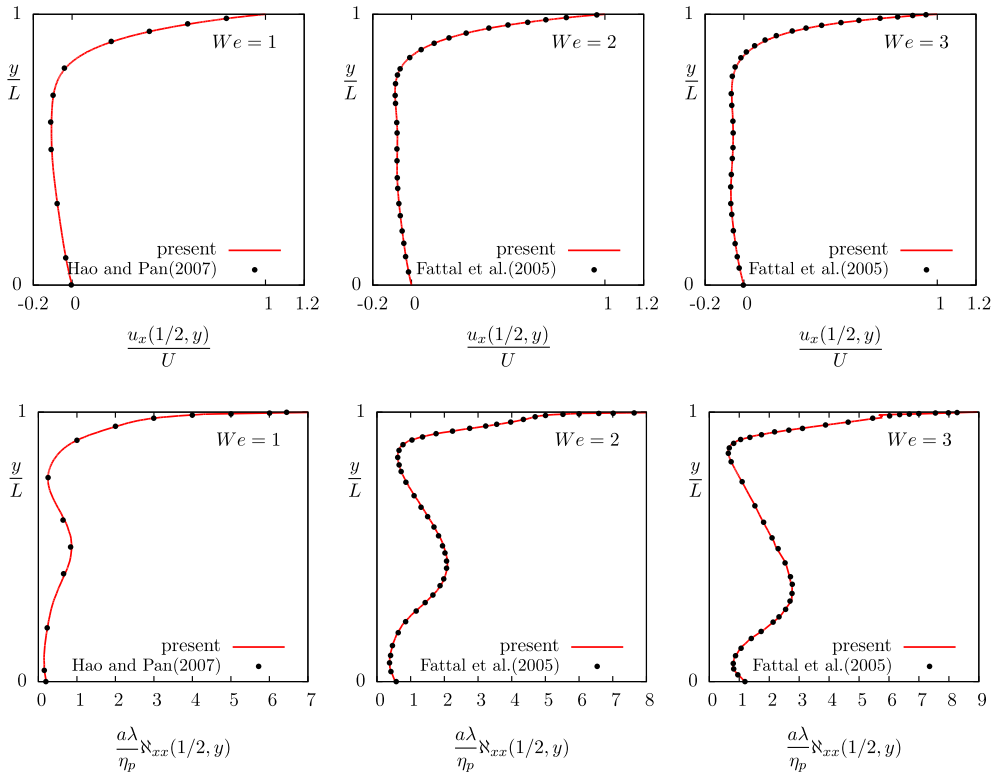

Fig. 9. The Oldroyd-B problem on the driven cavity with $\eta_p = \eta_v$. Cuts along the $x = 1/2$ vertical line: (top) $u_x(1/2,y)$; (bottom) $\chi_{xx}(1/2,y)$. Comparisons with results obtained by Hao and Pan [28], Fig. 2 for $We = 1$ and by Fattal and Kupferman [18], Figs. 7 and 10 for $We = 2$ and 3, respectively.

both the computations of Hao and Pan [28] and Fattal and Kupferman [18].

5. Conclusion

The new log-conformation formulation of viscoelastic fluid flow presented in this paper allows a direct steady numerical resolution by a Newton method. Moreover, the use an exact divergence free finite element method velocity–pressure approximation and a discontinuous Galerkin upwind treatment for stresses leads to a robust discretization. A demonstration is provided by the computation of steady solutions at high Weissenberg numbers for the difficult benchmark of the lid driven cavity flow. Numerical results are in good agreement, qualitatively with experiment measurements on real viscoelastic flows, and quantitatively with computations performed by others authors. The numerical algorithm is thus robust. It is also very efficient, as it requires few and mesh-invariant number of linear system resolutions to reach solutions at high Weissenberg number. An adaptive mesh procedure is also proposed, in order to catch accurately both boundary layers and main and secondary vortices.

We provide new data for future cross validation purpose and point out the existence of a singular point near $We = 2$, where the determinant of the Jacobian vanishes and exhibits a change of sign. This singular point has been found quite robust thought mesh refinement. As suggested by both experimental measurements and time-dependent simulations, it could be a bifurcation points associated to a loss of stationarity of the solution. The analysis of this situation by a steady approach is also possible with the tools of the bifurcation theory. In 2009, a pioneering work was performed in this direction by Howell [30] and it should be carried on in the context of the log-conformation formulation.

There are many geometries that could be explored with this approach: contractions, as in the previous reference, and flow around obstacles are interesting benchmarks, while experimental measurements are also available. As new mathematical tools are now available for solving viscoelastic flows problems, the situation becomes mature for considering efficient tridimensional flow simulations. The present log-conformation formulation extends naturally to more complex viscoelastic fluid models, such as Phan-Thien and Tanner, Giesekus or elastoviscoplastic one [59,60]. The integration of the Gordon–Schowalter derivative parameter in the present work also open the door of the numerical modeling of liquid crystals, fiber suspensions or active fluids.

Appendix A

This appendix contains explicit computations of some complex expressions, namely the exponential of a matrix, the κ function and their derivatives with respect to the matrix coefficients.

A.1. The matrix exponential and its derivatives

The present work uses the free software formal calculus system `maxima` [46]. With this system, the computation of the exponential of a 2×2 symmetric matrix writes:

```
load(linearalgebra);
domxexpt : false;
chi : matrix([a,b],[b,c]);
exp_chi : factor(matrixexp(chi));
```

The derivative $\exp'(\boldsymbol{\chi})$ of this expression with respect to the coefficients of the matrix $\boldsymbol{\chi}$ are then treated without any difficulty by this formal calculus system. Notice that Kane et al. [35, p. 50] proposed some explicit expressions for $\exp(\boldsymbol{\chi})$ and $\exp'(\boldsymbol{\chi})$, based on the `mapple` software: these expressions contained some errors and, for completeness, the correct expression are provided here.

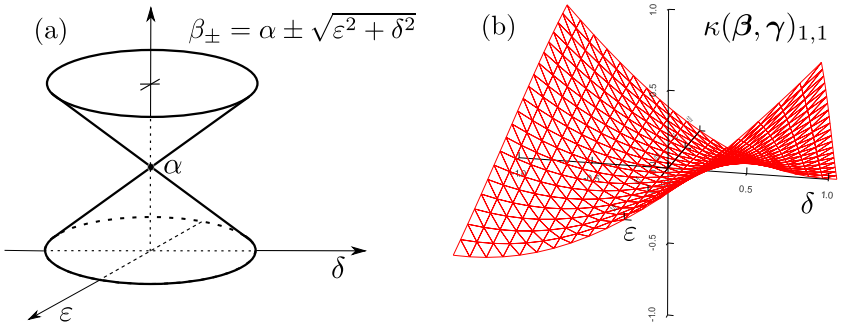

Fig. 10. (a) The eigenvalue function $\beta_\pm = \alpha \pm \sqrt{\varepsilon^2+\delta^2}$ and (b) the $\kappa(\boldsymbol{\beta},\boldsymbol{\gamma})_{1,1}$ function, with $\boldsymbol{\gamma} = ((1,2),(2,3))$.

$$\boldsymbol{\chi} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}, \quad \exp(\boldsymbol{\chi}) = \alpha_2 \begin{pmatrix} \alpha_4 + \frac{(a-c)\alpha_3}{\alpha_1} & \frac{2b\alpha_3}{\alpha_1} \\ \frac{2b\alpha_3}{\alpha_1} & \alpha_4 - \frac{(a-c)\alpha_3}{\alpha_1} \end{pmatrix}$$

where $\alpha_1 = \sqrt{(a-c)^2+4b^2}$, $\alpha_2 = \exp((a+c)/2)$, $\alpha_3 = \sinh(\alpha_1/2)$, and $\alpha_4 = \cosh(\alpha_1/2)$. Notice that the expression in [35] for $\exp'(\boldsymbol{\chi})$ should also be fixed in a similar way.

Observe that the obtained expression degenerates when both $a = c$ and $b = 0$, i.e. when the matrix is proportional to the identity. In that case $\boldsymbol{\chi} = aI$ and $\exp(\boldsymbol{\chi}) = \exp(a)I$. This possible degeneracy is decided when both $|a-c| < \varepsilon_m$ and $|b| < \varepsilon_m$ where ε_m stands for the machine epsilon, i.e. the higher positive floating number such that $1+\varepsilon_m = 1$. Finally, the expressions produced by `maxima` are simplified and optimized in an automatic way: they are then formatted in `fortran`, `C` or `C++` languages for a direct use by the Euler–Newton continuation algorithm. The formal calculus approach applies also for a 3×3 symmetric matrix: such expression is suitable for tridimensional flow geometries. The explicit expression of the 3×3 matrix exponential is not presented in this paper, as it expands on many pages. Nevertheless, the formal calculus approach is able to automatically generate the corresponding `fortran` or `C++` code in the tridimensional case.

A.2. The κ function and its derivatives

Magnus and Neudecker [43, p. 177] showed that the eigenvalues and eigenvectors are differentiable provided that eigenvalues are simple. Otherwise, when there are multiple eigenvalues, there is a serious problem. Let us study this difficulty by considering the 2×2 matrix function:

$$\boldsymbol{\beta}(\varepsilon,\delta) = \begin{pmatrix} \alpha+\varepsilon & \delta \\ \delta & \alpha-\varepsilon \end{pmatrix}$$

where α is some fixed constant and ε and δ are two variables at the vicinity of zero. The two eigenvalues and the two associated unnormalized eigenvectors are

$$\beta_\pm = \alpha \pm \sqrt{\varepsilon^2+\delta^2} \quad \text{and} \quad \mathbf{v}_\pm = \begin{pmatrix} 1 \\ -\left(\frac{\varepsilon}{\delta}\right) \pm \sqrt{1+\left(\frac{\varepsilon}{\delta}\right)^2} \end{pmatrix}$$

Both eigenvalues are continuous in ε and δ but not differentiable: the conical surface formed by the eigenvalues has a singularity at $(\varepsilon,\delta) = (0,0)$ (see Fig. 10a). For instance

$$\frac{\partial \beta_\pm}{\partial \varepsilon} = \pm\left(1+\left(\frac{\delta}{\varepsilon}\right)^2\right)^{-\frac{1}{2}}$$

For a fixed ratio $c = \delta/\varepsilon$ however, we can pass through $(0,0)$ without noticing the singularity, but the derivative depends upon c. Thus, the derivative $\frac{\partial \beta_\pm}{\partial \varepsilon}$ are multi-valued in $(0,0)$ and the two eigenvalues are not derivable at the origin. Notice that the two eigenvectors only depends upon c.

Nevertheless, the situation is not hopeless for the κ function: Fig. 10b plots the $\kappa(\boldsymbol{\beta},\boldsymbol{\gamma})_{1,1}$ component as a function of (ε,δ) for a specific $\boldsymbol{\gamma}$ value. Other components or $\boldsymbol{\gamma}$ values show a similar behavior. This observation suggests that κ is differentiable at $(\varepsilon,\delta) = (0,0)$ and that its derivative is zero. Let us prove this conjecture. But observing closely the definition (6a)–(6c) of the κ function shows that $\tilde{\kappa}$ depends only upon $(\beta_+ - \beta_-)/2 = \sqrt{\varepsilon^2+\delta^2}$ and is independent of α. More precisely, from (6b) we have $\tilde{\kappa}_{1,2} = \hat{\kappa}\left(\sqrt{\varepsilon^2+\delta^2}\right)\tilde{\gamma}_{1,2}$ where with $\hat{\kappa}(x) = 1 - x/\tanh(x)$. Then

$$\frac{\partial \tilde{\kappa}_{1,2}}{\partial \varepsilon} = \varepsilon g\left(\sqrt{\varepsilon^2+\delta^2}\right)\tilde{\gamma}_{1,2} + \hat{\kappa}\left(\sqrt{\varepsilon^2+\delta^2}\right)\frac{\partial \tilde{\gamma}_{1,2}}{\partial \varepsilon} \tag{17}$$

with $g(x) = \hat{\kappa}'(x)/x$. Observe that $\lim_{x\to 0} g(x) = -2/3$. Let us fix the direction $c = \delta/\varepsilon$ and do $\varepsilon \to 0$. By this way $\mathbf{v}_\pm$ depends only on c and so does the unitary matrix $\mathbf{q}$ whose columns are the normalized eigenvectors $\mathbf{v}_\pm/|\mathbf{v}_\pm|$. Thus $\tilde{\boldsymbol{\gamma}} = \mathbf{q}^T\boldsymbol{\gamma}\mathbf{q}$ does not dependent on ε. Then, the first term of the right-hand side of (17) behaves has $\mathcal{O}(\varepsilon)$. We now turn to the second term of the right-hand side of (17): On one hand, we have $\hat{\kappa}(x) = -x^2/3 + \mathcal{O}(x^3)$ and then $\hat{\kappa}\left(\sqrt{\varepsilon^2+\delta^2}\right) = \mathcal{O}(\varepsilon^2)$. On the other hand, $\frac{\partial \mathbf{v}_\pm}{\partial \varepsilon} = \mathcal{O}(\delta^{-1}) = \mathcal{O}(\varepsilon^{-1})$ and so are $\frac{\partial \mathbf{q}}{\partial \varepsilon}$ and $\frac{\partial \tilde{\gamma}_{1,2}}{\partial \varepsilon}$. Then, the second term of the right-hand side of (17) also behaves has $\mathcal{O}(\varepsilon)$. Finally, $\tilde{\kappa}_{1,2}$ is differentiable in $(0,0)$ and its derivative is zero. Now, let us turn to $\kappa = \mathbf{q}\tilde{\kappa}\mathbf{q}^T$. We have:

$$\frac{\partial \kappa}{\partial \varepsilon} = \frac{\partial \mathbf{q}}{\partial \varepsilon}\tilde{\kappa}\mathbf{q}^T + \mathbf{q}\frac{\partial \tilde{\kappa}}{\partial \varepsilon}\mathbf{q}^T + \mathbf{q}\tilde{\kappa}\frac{\partial \mathbf{q}^T}{\partial \varepsilon}$$

Recall that $\mathbf{q}$ depends only upon the constant direction c. Each term behaves as $\mathcal{O}(\varepsilon)$ and finally $\frac{\partial \kappa}{\partial \varepsilon} = 0$ at $(0,0)$. As ε and δ are interchangeable in the expression of κ, a similar deduction leads to $\frac{\partial \kappa}{\partial \delta} = 0$. Remark that, since the derivative is zero, it does not depend on the direction c. To conclude, κ is fully differentiable with respect to $\boldsymbol{\beta}$, even when $\boldsymbol{\beta}$ admits multiple eigenvalues.

The `maxima` code for computing $\kappa(\boldsymbol{\beta},\boldsymbol{\gamma})$ writes:

```
beta : matrix([a,b],[b,c]);
gamma : matrix([g00,g01],[g10,g11]);
eig : eigenvectors(beta);
beta1 : eig[1][1][1];
beta2 : eig[1][1][2];
do_unitary(v) := v/sqrt(v[1]**2 + v[2]**2);
v1 : do_unitary(eig[2][1][1]);
v2 : do_unitary(eig[2][2][1]);
q : matrix([v1[1],v2[1]], [v1[2],v2[2]]);
tilde_gamma : transpose(q).gamma.q;
hat_kappa(x) := 1 - x/tanh(x);
k12 : hat_kappa((beta1-beta2)/
  2)*tilde_gamma[1][2];
tilde_kappa : matrix([0, k12],[k12,0]);
kappa : q.tilde_kappa.transpose(q);
```

The derivatives of this expression with respect to the matrix coefficients are then treated without any difficulty. As shown in this section, there are two different degenerative cases, when $a = c$ or $b = 0$. These special cases are decided up to the machine precision and are treated separately. Finally, the expressions produced by `maxima` are simplified, optimized and formatted in `fortran`, `C` or `C++` for a direct use by the Euler–Newton continuation algorithm.

References

[1] A.M. Afonso, M. Oliveira, P.J. Oliveira, M.A. Alves, F.T. Pinho, The Finite-volume Method in Computational Rheology, In-tech publisher, 2012. pp. 141–170 (Chapter 7).
[2] P. Amestoy, A. Buttari, A. Guermouche, J.-Y. L'Excellent, B. Ucar, Multifrontal Massively Parallel Solver (MUMPS). User's Guide. Version 4.10.0. CERFACS, CNRS, INPT(ENSEEIHT)-IRIT and INRIA, France, 2011.
[3] D.N. Arnold, J. Qin, Quadratic velocity/linear pressure Stokes elements, Adv. Comput. Meth. Partial Diff. Eqn. 7 (1992) 28–34.
[4] J. Baranger, D. Sandri, Finite element approximation of viscoelastic fluid flow: existence of approximate solutions and error bounds. Part I. Discontinuous constraints, Numer. Math. 63 (1992) 13–27.
[5] R. Bird, R.C. Armstrong, O. Hassager, Dynamics of polymeric liquids, Kinetic Theory, vol. 2, Wiley, New-York, 1987.
[6] S. Boyaval, Modélisation mathématique et simulation numérique en science des matériaux, PhD Thesis, Université Paris Est, 2009.
[7] S. Boyaval, T. Lelièvre, C. Mangoubi, Free-energy-dissipative schemes for the Oldroyd-B model, ESAIM Math. Model. Numer. Anal. 43 (03) (2009) 523–561.
[8] F. Brezzi, On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers, RAIRO – Anal. Numer. 8 (2) (1974) 129–151.
[9] F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Methods, Springer, 1991.
[10] P.G. Ciarlet (Ed.), Handbook of Numerical Analysis, Numerical Methods for Non-Newtonian Fluids, vol. 16, Elsevier, 2011.
[11] O.M. Coronado, D. Arora, M. Behr, M. Pasquali, A simple method for simulating general viscoelastic fluid flows with an alternate log-conformation formulation, J. Non-Newton. Fluid Mech. 147 (3) (2007) 189–199.
[12] M.J. Crochet, M. Bezy, Elastic Effects in Die Entry Flows, in: Rheology, Springer, 1980, pp. 53–58.
[13] M.J. Crochet, R. Keunings, On numerical die swell calculation, J. Non-Newton. Fluid Mech. 10 (1) (1982) 85–94.
[14] H. Damanik, FEM Simulation of Non-isothermal Viscoelastic Fluids, PhD Thesis, Univ. Dortmund, Germany, 2011.
[15] H. Damanik, J. Hron, A. Ouazzi, S. Turek, A monolithic FEM approach for the log-conformation reformulation (LCR) of viscoelastic flow problems, J. Non-Newton. Fluid Mech. 165 (19) (2010) 1105–1113.
[16] D.A. di Pietro, A. Ern, Mathematical Aspects of Discontinuous Galerkin Methods, Springer, 2012.
[17] R. Fattal, R. Kupferman, Constitutive laws for the matrix-logarithm of the conformation tensor, J. Non-Newton. Fluid Mech. 123 (2) (2004) 281–285.
[18] R. Fattal, R. Kupferman, Time-dependent simulation of viscoelastic flows at high Weissenberg number using the log-conformation representation, J. Non-Newton. Fluid Mech. 126 (1) (2005) 23–37.
[19] E. Fernández-Cara, F. Guillén, R.R. Ortega, Some theoretical results concerning non-Newtonian fluids of the Oldroyd kind, Ann. Scuola Norm. Sup. Pisa Cl. Sci. 26 (1) (1998) 1–29.
[20] S.M. Fielding, D. Marenduzzo, M.E. Cates, Nonlinear dynamics and rheology of active fluids: simulations in two dimensions, Phys. Rev. E 83 (2011) 041910.
[21] A. Fortin, A. Zine, An improved GMRES method for solving viscoelastic fluid flow problems, J. Non-Newton. Fluid Mech. 42 (1–2) (1992) 1–18.
[22] M. Fortin, A. Fortin, A new approach for the FEM simulation of viscoelastic flows, J. Non-Newton. Fluid Mech. 32 (3) (1989) 295–310.
[23] M. Fortin, R. Pierre, On the convergence of the mixed method of Crochet and Marchal for viscoelastic flows, Comput. Meth. Appl. Mech. Eng. 73 (3) (1989) 341–350.
[24] V. Girault, P.A. Raviart, Finite Element Methods for the Navier–Stokes Equations. Theory and Algorithms, Springer, 1986.
[25] R.J. Gordon, W.R. Schowalter, Anisotropic fluid theory: a different approach to the dumbbell theory of dilute polymer solutions, J. Rheol. 16 (1972) 79–97.
[26] C. Guillopé, J.C. Saut, Résultats d'existence pour les fluides viscoélastiques à loi de comportement de type différentiel, C.R. Acad. Sci. Paris, Sér. 1 305 (11) (1987) 489–492.
[27] E. Hairer, G. Wanner, Polycopié genevois d'analyse numérique, Technical Report, Université de Genève, Suisse, 2006.
[28] J. Hao, T.-W. Pan, Simulation for high Weissenberg number viscoelastic flow by a finite element method, Appl. Math. Lett. 20 (9) (2007) 988–993.
[29] F. Hecht, BAMG: Bidimensional Anisotropic Mesh Generator, 2006. <http://www.ann.jussieu.fr/~hecht/ftp/bamg>.
[30] J.S. Howell, Dual-mixed finite element approximation of Stokes and nonlinear Stokes problems using trace-free velocity gradients, J. Comput. Appl. Math. 231 (2009) 780–792.
[31] D. Hu, T. Lelièvre, New entropy estimates for the Oldroyd-B model and related models, Commun. Math. Sci. 5 (4) (2007) 909–916.
[32] M.A. Hulsen, A sufficient condition for a positive definite configuration tensor in differential models, J. Non-Newton. Fluid Mech. 38 (1) (1990) 93–100.
[33] M.A. Hulsen, R. Fattala, R. Kupferman, Flow of viscoelastic fluids past a cylinder at high Weissenberg number: stabilized simulations using matrix logarithms, J. of Non-Newton. Fluid Mech. 127 (2005) 27–39.
[34] M.W. Johnson, D. Segalman, A model for viscoelastic fluid behavior which allows non-affine deformation, J. Non-Newton. Fluid Mech. 2 (1977) 255–270.
[35] A. Kane, R. Guénette, A. Fortin, A comparison of four implementations of the log-conformation formulation for viscoelastic fluid flows, J. Non-Newton. Fluid Mech. 164 (1) (2009) 45–50.
[36] R. Keunings, On the high Weissenberg number problem, J. Non-Newton. Fluid Mech. 20 (1986) 209–226.
[37] R. Kupferman, C. Mangoubi, E.S. Titi, A Beale–Kato–Majda breakdown criterion for an Oldroyd-B fluid in the creeping flow regime, 2014 (submitted for publication).
[38] Y. Kwon, A.I. Leonov, Stability constraints in the formulation of viscoelastic constitutive equations, J. Non-Newton. Fluid Mech. 58 (1) (1995) 25–46.
[39] Y.-L. Lee, J. Xu, C.-S. Zhang, Stable finite element discretizations for viscoelastic flow models, in: P.G. Ciarlet, J.-L. Lions (Eds.), Handbook of Numerical Analysis, Numerical Methods for Non-Newtonian Fluids, vol. 16, Elsevier, 2011, pp. 371–432 (Chapter 4).
[40] F.-H. Lin, C. Liu, P. Zhang, On hydrodynamics of viscoelastic fluids, Commun. Pure Appl. Math. 58 (11) (2005) 1437–1471.
[41] P.L. Lions, N. Masmoudi, Global solutions for some Oldroyd models of non-Newtonian flows, Chin. Ann. Math., Ser. B 21 (2) (2000) 131–146.
[42] G.G. Lipscomb, M.M. Denn, D.U. Hur, D.V. Boger, The flow of fiber suspensions in complex geometries, J. Non-Newton. Fluid Mech. 26 (3) (1988) 297–325.
[43] J.R. Magnus, H. Neudecker, Matrix Differential Calculus with Applications in Statistics and Econometrics, third ed., Wiley, New-York, 2007.
[44] C. Mangoubi, Analytical and Numerical Problems in the Computation of the Flow of Viscoelastic Fluids, PhD Thesis, Hebrew University, Kisley, Israel, 2008.
[45] J.M. Marchal, M.J. Crochet, A new mixed finite element for calculating viscoelastic flow, J. Non-Newton. Fluid Mech. 26 (1987) 77–144.
[46] Maxima, Maxima, A Computer Algebra System, 2013. <http://maxima.sourceforge.net>.
[47] J.G. Oldroyd, On the formulation of rheological equations of states, Proc. Roy. Soc. London A 200 (1950) 523–541.
[48] H.C. Öttinger, Beyond Equilibrium Thermodynamics, Wiley, 2005.
[49] R. Owens, T. Phillips, Computational Rheology, Imperial College Press, London, UK, 2002.
[50] P. Pakdel, S.H. Spiegelberg, G.H. McKinley, Cavity flows of elastic liquids: two-dimensional flows, Phys. Fluids 9 (11) (1997) 3123.
[51] T.-W. Pan, J. Hao, R. Glowinski, On the simulation of a time-dependent cavity flow of an Oldroyd-B fluid, Int. J. Numer. Meth. Fluids 60 (7) (2009) 791–808.
[52] J.-C. Paumier, Bifurcation et méthodes numériques. Applications aux problèmes elliptiques semi-linéaires, Masson, Paris, 1997.
[53] F. Pellegrini, PT-Scotch and libscotch 5.1 User's Guide, Université de Bordeaux and INRIA, France, 2010.
[54] L. Qi, J. Sun, A nonsmooth version of Newton's method, Math. Prog. 58 (1–3) (1993) 353–367.
[55] M. Renardy, Existence of slow steady flows of viscoelastic fluids with differential constitutive equations, Z. Angew. Math. Mech. 65 (9) (1985) 449–451.
[56] F. Rouvière, Petit guide de calcul différentiel à l'usage de la licence et de l'agrégation, second ed., Casini, Paris, 2003.
[57] P. Saramito, Numerical simulation of viscoelastic fluid flows using incompressible finite element method and a θ-method, M2AN 28 (1) (1994) 1–35.
[58] P. Saramito, Efficient simulation of nonlinear viscoelastic fluid flows, J. Non Newton. Fluid Mech. 60 (1995) 199–223.
[59] P. Saramito, A new constitutive equation for elastoviscoplastic fluid flows, J. Non Newton. Fluid Mech. 145 (1) (2007) 1–14.
[60] P. Saramito, A new elastoviscoplastic model based on the Herschel–Bulkley viscoplasticity, J. Non Newton. Fluid Mech. 158 (1–3) (2009) 154–161.
[61] P. Saramito, Efficient C++ Finite Element Computing with Rheolef, CNRS and LJK, 2013. <http://cel.archives-ouvertes.fr/cel-00573970>.
[62] P. Saramito, Efficient C++ Finite Element Computing with Rheolef, Discontinuous Galerkin Methods, vol. 2, CNRS and LJK, 2013. <http://cel.archives-ouvertes.fr/cel-00863021>.
[63] P. Saramito, Méthodes numériques en fluides complexes : théorie et algorithmes, CNRS-CCSD, 2013. <http://cel.archives-ouvertes.fr/cel-00673816>.
[64] P. Saramito, J.-M. Piau, Flow characteristics of viscoelastic fluids in an abrupt contraction by using numerical modeling, J. Non Newton. Fluid Mech. 52 (1994) 263–288.
[65] L.R. Scott, M. Vogelius, Norm estimates for a maximal right inverse of the divergence operator in spaces of piecewise polynomials, M2AN 19 (1) (1985) 111–143.
[66] R. Seydel, Practical Bifurcation and Stability Analysis, third ed., Springer, 2010.
[67] J. Su, J. Ouyang, X. Wang, B. Yang, W. Zhou, Lattice Boltzmann method for the simulation of viscoelastic fluid flows over a large range of Weissenberg numbers, J. Non-Newton. Fluid Mech. 194 (2013) 42–59.
[68] C. Taylor, P. Hood, A numerical solution of the Navier–Stokes equations using the finite element technique, Comput. Fluids 1 (1) (1973) 73–100.
[69] P. Wapperom, M.A. Hulsen, Thermodynamics of viscoelastic fluids: the temperature equation, J. Rheol. 42 (1998) 999.
[70] S. Zhang, A new family of stable mixed finite elements for the 3d Stokes equations, Math. Comput. 74 (250) (2005) 543–554.