The deformation fields method revisited: Stable simulation of instationary viscoelastic fluid flow using integral models

Martien A. Hulsen*, Patrick D. Anderson

Department of Mechanical Engineering, Eindhoven University of Technology, PO Box 513, Eindhoven, MB 5600, The Netherlands

**Abstract**

The implementation of the deformation fields method for integral models within a finite element context [1,2] has been updated with various techniques to have a numerical stability that is comparable to state-of-the-art implementations of differential models. In particular, the time-dependent stability in shear flow, decoupled schemes for zero or small solvent viscosities and the log-conformation representation now have counterparts in the numerical implementation of integral models leading to similar numerical stability. The new techniques have been tested in transient shear flow and the flow around a cylinder confined between two plates for the integral version of upper-convected Maxwell model and for integral models having a non-constant damping function.

1. Introduction

Integral models are frequently used for modelling the constitutive behaviour of polymer liquids. In particular the factorable K-BKZ equation, where the linear rheology is specified by the memory function and the non-linear rheology is given by the damping function, is very popular. For an extensive review of the various damping functions that have been used see [3]. Also, new constitutive models based on molecular considerations are often formulated in terms of memory integrals, see for example [4]. Furthermore, recent developments in modelling complex fluids with a broad power-law relaxation spectrum are also of the integral type [5].

Various techniques have been proposed to simulate flows of viscoelastic fluids described by integral models. For a review of the various finite element method implementations for integral models see [6]. As has been concluded in that review, currently only two methods are capable of performing transient simulations: the Lagrangian Integral Method (LIM) [7,8] and the Deformation Fields Method (DFM) [1,2]. LIM is suitable in a Lagrangian formulation, whereas DFM can be used in any formulation, but is used mainly in an Eulerian (fixed domain) or Arbitrary Lagrange Euler (ALE) formulation.

Despite the many good constitutive models available, numerical simulation of viscoelastic fluid flows using integral models is less popular than simulations with differential models. The reason is that implementation of integral models is more involved because of the complete deformation history that needs to be determined. As a result, new stabilization techniques have been specifically developed for differential models and the numerical techniques for integral models are lagging behind in this regard. In particular, the time-dependent stability in shear flow of the velocity-gradient projection technique [9,10], decoupled schemes for zero or small solvent viscosities [11] and the log-conformation representation (LCR) [12] do not have a similar counterpart in the numerical implementation of integral models.

In this paper, we propose and implement various techniques within a finite element context to give the DFM implementation of integral models stability properties comparable with state-of-the-art finite element techniques for differential models. To show stability and accuracy, we apply the techniques to a perturbed simple shear flow and the standard benchmark geometry of the flow around a cylinder confined between plates.

The paper is structured as follows. In Section 2, we present the balance equations and the constitutive models used. Also the DFM is described and rewritten in terms of the deformation gradient instead of the Finger deformation tensor. In Section 3, the numerical discretization based on the finite element method is described. In Section 4, simulation results are presented. Finally, in Section 5 we end with a discussion and final conclusions.

2. Governing equations

2.1. Balance equations and constitutive models

Consider a flow in a fixed domain Ω (see Fig. 1). We assume that inertia can be neglected and that the fluid is incompressible. Therefore the balance of momentum and mass become

---

*Corresponding author.

E-mail addresses: m.a.hulsen@tue.nl (M.A. Hulsen), p.d.anderson@tue.nl (P.D. Anderson).

https://doi.org/10.1016/j.jnnfm.2018.03.001

Received 30 October 2017; Received in revised form 20 January 2018; Accepted 3 March 2018

Available online 07 March 2018

0377-0257/ © 2018 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/BY/4.0/).
− \nabla \cdot (2\eta_0 \mathbf{D}) + \nabla p - \nabla \cdot \mathbf{\tau} = 0, \quad (1)

\nabla \cdot \mathbf{u} = 0, \quad (2)

where \( \mathbf{u} \) is the velocity vector field, \( p \) is the pressure field, \( \mathbf{\tau} \) is the extra-stress tensor field, \( \eta_0 \) is the initial viscosity of the fluid and \( \mathbf{D} = (\mathbf{L} + \mathbf{L}^T)/2 \) is the symmetric part of the velocity gradient tensor \( \mathbf{L} = (\nabla \mathbf{u})^T \). All fields are dependent on time \( t \) and position in space \( x \), i.e. \( \mathbf{u}(t, x) \), but function arguments will only be added when needed. The extra-stress tensor field \( \mathbf{\tau} \) is given by a memory integral of the K-BKZ type:

\[ \mathbf{\tau}(t, x) = \int_{t'}^t m(t - t')h(l(t))\mathbf{B} \, dt', \quad (3) \]

where \( \mathbf{B} = \mathbf{B}(t, t', x) \) is the Finger deformation tensor of a material particle at the current position \( x \) and time \( t \), with respect to the reference configuration at time \( t' \), \( m(t) \) (with \( t' \leq t \geq 0 \)) is the memory function and \( h(l(t)) \) is the damping function depending on the first invariant \( l_1 = (\mathbf{B} \cdot \mathbf{B}) \).

For the purpose of this paper, but without loss of generality, we restrict \( m(t) \) to be a single exponential

\[ m(t) = \frac{G}{\lambda} \exp(-\tau/\lambda), \quad (4) \]

where \( G \) is the modulus and \( \lambda \) is the relaxation time. For the damping function we will use

\[ h(l_1) = \frac{1}{1 + a(l_1 - 3)}, \quad (5) \]

as introduced in [13] and also used in [5] for the fractional K-BKZ integral model. Note, that for \( a = 0, \eta_0 = 0 \) the constitutive model reduces to the integral form of the upper-convected Maxwell (UCM) model.

For verification of the results, we use the UCM model in differential form, given by

\[ \tau = Gc, \quad \frac{\partial \mathbf{c}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{c} = -\mathbf{c} \cdot \mathbf{L}^T \mathbf{c} - c \mathbf{L}^T + (c - I)/\lambda = 0, \quad (6) \]

where \( c \) is the conformation tensor. For the UCM model \( \eta_0 = 0 \). Note, that in equilibrium \( c = I \) and \( \tau = GI \). For integral models it is quite common to drop isotropic terms from the extra-stress expression (see, for example [14]). To easily compare the integral and differential formulation of UCM, we did the same for the latter here also.

### 2.2. Initial and boundary conditions

It is assumed that before the start time \( t = 0 \) the flow is at rest, which means that \( \mathbf{B}(t, t', x) = \mathbf{I} \) for \( t' \leq t < 0 \) and all \( x \in \Omega \). The stress tensor in this equilibrium state is \( \tau = GI \). The boundary conditions on \( \Gamma \) are similar to the (Navier-) Stokes equations, for example specification of the velocity vector on one part of \( \Gamma \) (Dirichlet) and the traction vector on the remaining part of \( \Gamma \) (Neumann). If the velocity vector on the boundary is such that there is an inflow boundary \( \Gamma_{\text{in}} \) (see Fig. 1), the fluid deformation history at that boundary needs to be specified. However, if the solvent viscosity is zero (\( \eta_0 = 0 \)) it is not allowed to specify the fluid history completely, similar to the restrictions on imposing stress components at the inflow boundary for differential models [15,16]. In order to avoid complications related to inflow boundaries, we assume periodical boundary conditions for the actual problems considered in this paper. In case of periodical boundary conditions the inflow boundary is coupled to outflow boundary, therefore the flow in the domain \( \Omega \) provides its own deformation history for the fluid particles entering the domain.

### 2.3. Deformation fields

For the evaluation of the memory integral in Eq. (3), we need the deformation history \( \mathbf{B}(t, t', x) \), \( -\infty < t' \leq t \) for the particle that is currently at the position \( x \). In a Lagrangian description, having the positions of the material particles as a primary unknown, it is relatively easy to obtain the deformation gradient \( \mathbf{F}(t, t', x) = \partial \mathbf{x}/\partial \mathbf{x}' \), \( -\infty < t' \leq t \) and then compute the Finger tensor via \( \mathbf{B} = \mathbf{F} \cdot \mathbf{F}^T \). This is the approach taken in [7,8] and works well for problems where a fixed amount of fluid is being studied, such as stretching a filament. However, for problems in a fixed domain the Eulerian description is preferred. For that purpose, the method of deformation fields [1,2] has been developed.

Assume that at \( t = 0 \) the flow starts and there is no deformation of the material before that. One way of obtaining \( \mathbf{F}(t, t', x) \) for \( t' \leq t \) is defining (a large) number of discrete reference times \( t_i' \) (both for \( t < 0 \) and \( t > 0 \)) and solve the following equation

\[ \mathbf{F} = \mathbf{L} \cdot \mathbf{F}, \quad \mathbf{F}(t = t_i') = \mathbf{I}, \quad (7) \]

forward in time for all discrete reference times \( t_i' \leq t \), i.e. a discrete time point ‘kicks in’ as soon as time \( t \) passes the time point. In an Eulerian frame, this equation is written as follows,

\[ \frac{\partial \mathbf{F}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{F} = \mathbf{L} \cdot \mathbf{F}, \quad \mathbf{F}(t = t_i') = \mathbf{I}. \quad (8) \]

The solutions for \( \mathbf{F}(t, t', x) \) are called deformation fields. There is a problem with this approach: the number of ‘active fields’, i.e. the fields with \( t_i' \leq t \) keeps increasing as time increases. This can be overcome by removing the ‘oldest field’ as a new field becomes active, hence keeping the number of active fields constant. This is basically the first generation of the deformation fields method as introduced in [1], except that in [1] \( \mathbf{B} \) is computed directly (see below) and not with \( \mathbf{F} \) as an intermediate field. A drawback of this approach, in contrast to differential models, is that the fields are unsteady for steady flows, which leads to additional time discretization errors. Also the labelling of the fields by the reference time leads to difficult bookkeeping, including the removal and creation of fields at each time step. As a result, also the distribution of fields over the history is not very flexible and basically coupled with the time step \( \Delta t \).

The second-generation deformation fields [2] removes much of the drawbacks of the first generation by labeling the deformation gradient tensor not by an absolute (fixed) reference time \( t' \), but by the reference time relative to the current time, i.e. by the age \( \tau = t - t': \mathbf{F}(t, \tau, x) \). For that, Eq. (7) is rewritten in terms of the new set of independent variables as follows:

\[ \frac{\partial \mathbf{F}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{F} = \mathbf{L} \cdot \mathbf{F}, \quad (9) \]

which is a 4D hyperbolic equation (assuming 3D space) with a ‘convection speed’ of 1 in the \( \tau \)-axis direction. The boundary condition (for the \( \tau \)-axis) is

\[ \mathbf{F}(t, \tau = 0, x) = \mathbf{I}, \quad (10) \]

and the initial condition (fluid at rest for \( t < 0 \)) is

\[ \mathbf{F}(t = 0, \tau, x) = \mathbf{I}. \quad (11) \]

In principal the age axis is unlimited, i.e. \( \tau \in [0, \infty) \). However, in
practice we are dealing with ‘fading memory fluids’ and the age axis is taken to be finite i.e. \( \tau \in [0, \tau_c] \), with \( \tau_c \) the cut-off time.

Note, that in [2] the Finger tensor \( \mathbf{B}(t, \tau, x) \), \( \tau \in [0, \tau_c] \) is computed directly using

\[
\frac{\partial \mathbf{B}}{\partial t} + \frac{\partial \mathbf{B}}{\partial \tau} + \mathbf{u} \cdot \nabla \mathbf{B} = \mathbf{L} \mathbf{B} + \mathbf{B} \mathbf{L}^T, \tag{12}
\]

with boundary condition (for the \( \tau \)-axis)

\[
\mathbf{B}(t, \tau = 0, x) = \mathbf{I}, \tag{13}
\]

and initial condition (fluid at rest for \( t < 0 \))

\[
\mathbf{B}(t = 0, \tau, x) = \mathbf{I}. \tag{14}
\]

We call this the \( B \)-approach, which has the advantage that the number of components to solve is less, since \( \mathbf{B} \) is symmetric. A disadvantage is that the equation to solve contains an additional term as compared to the \( F \)-approach given by Eq. (9), possibly increasing the computational effort. More importantly however, it turns out that the \( B \)-approach is less stable than the \( F \)-approach and that the latter is the preferred approach, as we will show later.

3. Numerical discretization

The unknown field variables in the \( F \)-approach are the velocity \( \mathbf{u}(t, x) \), the pressure \( p(t, x) \) and the deformation gradient \( \mathbf{F}(t, \tau, x) \). The governing equations are Eqs. (1)-(3) and Eq. (9). We will use the finite element method to discretize in real space (\( x \)) and in ‘age space’ (\( \tau \)). Time discretization will be based on second-order backward differentiating (BDF2). We start with the discretization of the deformation gradient.

3.1. Weak formulation and discretization of the deformation gradient

We will apply the finite element method to discretize Eq. (9) in both the \( \tau \)-direction and in real space. For that, we write the weak formulation as follows: find \( \mathbf{F}(t, \tau, x) \) such that

\[
\int_0^{\tau} \left( \mathbf{H}(t, \tau, x) \cdot \frac{\partial \mathbf{F}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{F} - \mathbf{L} \mathbf{F} \right) \, d\tau = 0, \tag{15}
\]

for all test functions \( \mathbf{H}(t, \tau, x) \). Here, \( (\cdot) \) is the usual inner product for tensors in real space. Note, that we have limited integration in the \( \tau \)-direction to the finite interval with a cut-off time, i.e. \( [0, \tau_c] \).

Although quite general four-dimensional elements could be used for discretization, we will not do that. First, we divide the integration domain \( [0, \tau_c] \) into \( N \) intervals \( \Delta \tau = \tau_c/N \), \( k = 0, ..., N-1 \), where \( \tau_c = \tau_c \) and \( \tau_N = \tau_c \). This division is global and independent of \( x \), thus creating ‘slab’ elements. We will employ ‘stretched’ grids as in [2], where \( \Delta \tau = f \Delta \tau_k \) (\( k = 0 \)) with constant factor \( f \). The factor \( f \) is uniquely determined by \( \Delta \tau_k \) and \( \tau_k \) and can be found by iteration. In this work, we have used \( \Delta \tau = 3\lambda /N \), which is close to the optimal value for a single exponential memory function, as determined in [2]. When appropriate, the value of \( f \) in the simulations is included in the caption of the figures.

We apply the discontinuous Galerkin method in \( \tau \)-space, using piecewise continuous base functions, that can jump across the slab boundaries. In real coordinate space we employ a standard mesh with continuous base functions. Since we solve a hyperbolic equation, we apply SUPG stabilization. We can now write the discretized weak form on a single slab:

\[
\int_{\tau_k}^{\tau_{k+1}} \left( \mathbf{H}(t, \tau, x) \cdot \frac{\partial \mathbf{F}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{F} - \mathbf{L} \mathbf{F} \right) \, d\tau + \mathbf{H}(t, \tau, x) \cdot [\mathbf{F}(t, \tau, x)]_{\tau_k} = 0, \tag{16}
\]

where \([\mathbf{F}](t, \tau, x)\) is the jump of \( \mathbf{F} \) at the slab boundary \( \tau_k \) and \( \mathbf{F} \) is the SUPG time scale parameter. Here, the superscripts + and − denote the value at \( \tau_k \) in interval \( k \) and \( k-1 \), respectively. Note, that the use of the DG method together with the division into intervals (slabs) makes it possible to solve \( \mathbf{F} \) in the \( \tau \)-direction ‘slab by slab’ like a time step scheme. First the entry condition \( \mathbf{F}^-(\tau_c) = \mathbf{I} \) is imposed, then slab 0 is solved, which gives \( \mathbf{F}^+(\tau_c) \) as input for slab 1, etc.

Now we introduce the base functions for \( \mathbf{F}(t, \tau, x) \) in \( \tau \)-direction by writing the test and trial functions as follows

\[
\mathbf{H}(t, \tau, x) = \sum_i \phi_i(\tau) \mathbf{H}(t, x), \quad \mathbf{F}(t, \tau, x) = \sum_i \phi_i(\tau) \mathbf{F}(t, x), \tag{17}
\]

where \( \phi_i(\tau) \) are piecewise continuous base functions. The functions \( \mathbf{F}(t, x) \) are called the discrete deformation fields, where the \( j \)-index denotes an index in the \( \tau \)-direction. The number of discrete fields depends on the polynomial order of \( \phi(\tau) \). For order \( m \) base function the number of ‘nodes’ within an interval is \( n_i = m + 1 \), and therefore the number of fields is \( n_iN = (m + 1)N \) and the number of components of the fields is \( n_iN = n_iN^d = (m + 1)Nd^2 \), where \( d \) is the space dimension (2 or 3). The discrete deformation fields can be discretized in real space using standard finite element base functions, for which we use linear base functions \( (P_1) \) on triangles/tetrahedra and bi- or trilinear base functions \( (Q_1) \) on quadrilaterals/hexahedra.

Substituting Eq. (17) into Eq. (16), we get ‘slab for slab’:

\[
\int_{\tau_k}^{\tau_{k+1}} \left( \mathbf{H}(t, \tau, x) \cdot \frac{\partial \mathbf{F}}{\partial \tau} + \frac{\partial \mathbf{F}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{F} - \mathbf{L} \mathbf{F} \right) \, d\tau + \mathbf{H}(t, \tau, x) \cdot \mathbf{F}(t, \tau, x)_{\tau_k} = 0, \tag{18}
\]

where

\[
m_i^k = \int_{\tau_k}^{\tau_{k+1}} \phi_i(\tau) \, d\tau, \tag{19}
\]

\[
n_i^k = \int_{\tau_k}^{\tau_{k+1}} \phi_i(\tau) \frac{d\phi_i(\tau)}{d\tau} \, d\tau + \phi_i(\tau_k)\phi_i(\tau_{k+1}), \tag{20}
\]

and the indices \( i \) and \( j \) are limited to fields in slab \( k \).

The system Eq. (18) can be solved ‘slab for slab’ following the sequence \( k = 0, ..., N-1 \), where the solution values at the end of the slab, i.e. \( \mathbf{F}^+(t, \tau_{k+1}, x) \), are input for the next slab. This is similar to a time-stepping scheme. In fact, the DG-method using order \( q \) polynomials leads to a \( 2q+1 \) order accurate implicit Runge–Kutta method [17].

Time discretization is based on backward differentiating, where we combine implicit BDF2 for all terms, except \( \mathbf{L} \mathbf{F} \), which is treated with extrapolated BDF [18] of the same order. In this way, the components of the fields are decoupled and the system matrix for each component is the same. The linear system is non-symmetric and a direct solver from the HSL-library (MA41) [19] is used in this work to obtain the solution for \( F \).

After computing the deformation fields, memory integrals, like in Eq. (3), are numerically evaluated by applying a standard Gauss rule on each slab in \( \tau \)-direction and adding the results of all \( N \) slabs.

The \( B \)-approach is similar to the \( F \)-approach, applying the same discretization methods and base functions to Eq. (12) for the tensor field \( \mathbf{B} \), instead of Eq. (9) for the tensor field \( \mathbf{F} \). Note, that in the \( F \)-approach the \( B \)-tensor field is computed by \( \mathbf{F}^T \mathbf{F} \) and is therefore guaranteed to be positive semidefinite everywhere, whereas in the \( B \)-approach positive semidefiniteness of \( B \) might be locally violated due to poor resolution of the mesh. The latter is also true for the stress tensor \( \tau \) in the \( B \)-approach, although to a lesser degree, since it is a weighted sum of many \( B \)-fields.

3.2. Discretization of the momentum and mass balance

Our aim is to formulate a discretization of the momentum and mass balance that is able to simulate time-dependent flows having zero or low-solvent viscosity contribution. For that, we proceed similarly as in [11], where the differential constitutive model is integrated over the
time step $\Delta t$ to obtain the stress tensor at the end of the time step ($t_{n+1}$) up to an error of $O(\Delta t^2)$. An ‘Euler-like’ scheme is applied, where the velocity and pressure field is taken implicit and the stresses (conformation) taken explicit. The resulting equation for $\tau_{n+1}$ is substituted in the momentum balance, evaluated at time $t_{n+1}$ to obtain a Stokes-like set of equations. This set of equations introduces a time error of $O(\Delta t^2)$ for the momentum balance, consistent with second-order integration of the constitutive model.

In order to apply a similar procedure for integral models, we need to obtain a ‘differential form’ of the integral model Eq. (3). For that, we differentiate the integral model with respect to time. This is performed in Appendix A and the result is

$$
\frac{\partial \tau}{\partial t} + \mathbf{u} \cdot \nabla \tau = m(0) I + A + K : 2D + L \cdot \tau + \tau L^T.
$$

(21)

where

$$
\tau = \int_0^\infty m(\tau) h(\tau) B d\tau,
$$

(22)

$$
A = \int_0^\infty \frac{dn(\tau)}{d\tau} h(\tau) B d\tau,
$$

(23)

$$
K = \int_0^\infty m(\tau) \frac{dn}{d\tau} BB d\tau.
$$

(24)

Note, that the second-order tensors $\tau$, $A$, and the fourth-order tensor $K$ can be computed once the deformation history $B(t)$ is known.

The discretization of the momentum and mass balance is now performed as follows. At the next discrete time ($t_{n+1}$) the momentum and mass balance are evaluated:

$$
\mathbf{V} \cdot (2\eta \mathbf{D}_{n+1} + \mathbf{P}_{n+1} - \mathbf{V} \cdot \mathbf{\tau}_{n+1}) = \mathbf{0},
$$

(25)

$$
\mathbf{V} \cdot \mathbf{u}_{n+1} = \mathbf{0}.
$$

(26)

An expression for $\mathbf{\tau}_{n+1}$ is found by integrating the differential expression Eq. (21) from $t_n$ to $t_{n+1} = t_n + \Delta t$ using a combined explicit-implicit Euler scheme, where $\tau$, $A$, and $K$ are evaluated at $t_n$ and $u$ at $t_{n+1}:

$$
\frac{\mathbf{\tau}_{n+1} - \mathbf{\tau}_n}{\Delta t} + \mathbf{V} \cdot \mathbf{\tau}_n = m(0) I + A_n + K_n : 2\mathbf{D}_{n+1} + (\mathbf{V} \mathbf{u}_{n+1})^T \mathbf{\tau}_n
$$

$$
+ \mathbf{\tau}_n \mathbf{V} \mathbf{u}_{n+1}.
$$

(27)

After evaluation of $\mathbf{\tau}_{n+1}$ and substitution into Eq. (25) we get the following ‘Stokes’ like system for $\mathbf{u}_{n+1}$ and $\mathbf{P}_{n+1}:

$$
\mathbf{V} \cdot (2\eta \mathbf{D}_{n+1} + \mathbf{P}_{n+1} - \Delta t \mathbf{V} \cdot (-\mathbf{u}_{n+1} \cdot \mathbf{V} \mathbf{\tau}_n
$$

$$
+ K_n : [(\mathbf{V} \mathbf{u}_{n+1})^T + \mathbf{V} \mathbf{u}_{n+1}] + (\mathbf{V} \mathbf{u}_{n+1})^T \mathbf{\tau}_n
$$

$$
+ \mathbf{\tau}_n \cdot \mathbf{V} \mathbf{u}_{n+1}) - \mathbf{V} \cdot \mathbf{\tau}_n + \Delta t [m(0) I + A_n])
$$

$$
\mathbf{V} \cdot \mathbf{u}_{n+1} = \mathbf{0}.
$$

(28)

(29)

Note, that the error in $\mathbf{\tau}_{n+1}$ is $O(\Delta t^2)$. Therefore, we can expect an error of also $O(\Delta t^2)$ in the time discretization of the momentum balance. Since the deformation at time $t_n$, $\mathbf{B}_n$ is known and a continuous field, the system is well-defined and can be discretized in space using standard velocity-pressure finite-elements. For that, we use Taylor-Hood elements ($P_2/P_1$ on triangles/tetrahedra, $Q_2/Q_1$ on quadrilaterals/hexahedra).

One practical problem in the finite element discretization is the evaluation of the stress convection term $\mathbf{u}_{n+1} \cdot \mathbf{V} \mathbf{\tau}_n$. We avoid taking spatial derivatives of $\mathbf{\tau}_n$, which is not in a $P_1$ (triangles/tetrahedra) or $Q_1$ (quadrilaterals/hexahedra) space, by first projecting $\mathbf{\tau}_n$ on a $P_1$ or $Q_1$ space and using the projected stress field in evaluating the stress gradients.

We note that in this work we use the DEVSS-G method [10,20] for additional stabilization of the momentum balance. This involves adding an additional term $-\mathbf{V} \cdot (\mathbf{V} \mathbf{u}_{n+1} - G^T_{n+1})$ to the left-hand side of Eq. (28) and solving an additional equation

$$
- \mathbf{V} \mathbf{u}_{n+1} + G^T_{n+1} = \mathbf{0}.
$$

(30)

The auxiliary variable $G$ is interpolated using continuous basis functions ($P_1$ on triangles/tetrahedra, $Q_1$ on quadrilaterals/hexahedra), which means that $G$ is the least-squares projection of the velocity gradient $(\mathbf{V} \mathbf{u})^T$ on the $P_1$ or $Q_1$ continuous finite element space. For the coefficient $\alpha$ we take the zero-shear-rate polymer viscosity, which is $\alpha \lambda_0$ for the memory function given in Eq. (4).

The linear system is non-symmetric and a direct solver from the HSL-library (MA41) [19] is used in this work to obtain the solution for $\mathbf{u}_{n+1}$, $\mathbf{P}_{n+1}$ and $G_{n+1}$.

The solution for $\mathbf{u}_{n+1}$ is used to obtain the deformation fields at the new time $t_{n+1}$, i.e. $\mathbf{F}(t_{n+1}, x)$, using the discretization discussed in Section 3.1 with $L = (\mathbf{V} \mathbf{u}_{n+1})^T$. Similar to differential models [9,21], it is also possible to use $L = \mathbf{G}_{n+1}$ instead. It will be shown below that the latter option leads to much better temporal stability in shear flow.

4. Results

4.1. Planar Couette flow of an upper-convected Maxwell fluid

We consider an upper-convected Maxwell fluid (UCM) in integral form for a planar Couette problem on a $H \times H$ square (see Fig. 2). On the upper boundary a horizontal velocity of $U$ and a vertical velocity of zero is imposed. On the lower wall the velocity components are zero. In the horizontal direction periodical boundary conditions are assumed. In the lower left corner the pressure value is set to zero (Dirichlet). The Weissenberg number of this flow is defined as $\text{Wi} = \lambda \gamma$, with $\gamma = U/H$ the (average) shear rate. In the actual computations we set $U = 1$, $H = 1$ and thus $\gamma = 1$, $\text{Wi} = \lambda$. At the start of the flow the deformation fields will be set to the steady state values plus a small random perturbation of $\mathcal{C}(10^{-3})$, i.e.

$$
F(t = 0, \tau, x) = I + \gamma \mathbf{r} e_r e_\tau + 10^{-3} \text{rand}.
$$

(31)

where rand is $2 \times 2$ tensor where each of the components in the $(x, y)$ Cartesian system is a random number on the interval $[0, 1]$, set independently for all nodes and all fields. Note, that $\gamma$ is the shear strain, which is equal to $\text{Wi}$ over a time span equal to $\lambda$. Therefore, the perturbation is assumed to be small enough for the difference from the steady state to be considered a linear perturbation for high values of $\text{Wi}$. Results for the integral UCM at $\text{Wi} = 10$ for a $20 \times 20$ mesh (with quadrilateral elements) are shown in Fig. 3. The numerical technique applied is described in [11] (without LCR). The numerical parameters used are given in the figure caption.

For comparison, we included the results obtained using a differential UCM model (see Eq. (6)) in Fig. 4. We see that our implementation of the deformation fields behaves quite similar to the differential model implementation. We get the typical stable curve
\((9,21)\) when \(L = G\) is used in the evolution of \(F\). After an initial increase, the perturbation will die out and eventually becomes a straight line in the log plot. The form of the transient for the continuous spectrum, as predicted by Kupferman [22], is also shown. Eventually, there is an even slower decaying eigenvalue, which is probably related to the two discrete Gorodtsov & Leonov (GL) modes [23]. The discrete GL-modes decay as \(\exp(-t/2\lambda)\) for \(Wi \gg 1\) and are located near the walls. Whether we resolve the GL-modes with this rather coarse mesh, is beyond the scope of the current paper.

Finally, we note that using \(L = (\nabla u)^T\), i.e. obtaining the velocity gradient directly from differentiating the finite element velocity field, leads to a numerically unstable flow for the integral UCM model using a deformation fields method, as shown in Fig. 3. This instability, is similar to the well-known behaviour when using the differential UCM with \(L = (\nabla u)^T\) \((9,21)\), as shown in Fig. 4.

For the remaining part of this paper, we will use \(L = G\) throughout.

### 4.2. Flow around a confined cylinder

#### 4.2.1. Problem description

We consider the planar flow past a cylinder of radius \(R\), centrally positioned between two flat plates (see Fig. 5). The distance between the two plates is \(2H\), with \(H = 2R\) and the length of the flow domain is \(30R\). In the following we will use an \((x, y)\) coordinate system with the origin at the center of the cylinder. We assume the flow domain to be periodically extended in \(x\)-direction with period \(30R\). As explained in Section 2, this avoids having to specify inflow boundary conditions for the deformation fields. The flow is generated by imposing a constant flow rate \(Q\). We assume no-slip boundary conditions on the cylinder and walls of the channel. We only consider half of the domain and use symmetry conditions on the center line, i.e. zero tangential traction. The dimensionless parameters governing the problem are the Weissenberg number \(Wi = \lambda U/R\), the viscosity ratio \(\beta = \eta_i/\eta_0\), and the parameter \(\alpha\) in the damping function (Eq. (5)). Here, \(U = Q/2H\) is the average velocity and \(\eta_0 = \eta_i + \dot{\gamma}\) is the zero-shear-rate viscosity of the fluid.

#### 4.2.2. Mesh

We use two series of uniformly refined meshes, created using Gmsh [24]. The element sizes are imposed by specifying the size in geometrical points on the boundary of the model in Gmsh. The characteristics of the first series of meshes used (M0–M4) are given in Table 1. In Fig. 6, the starting mesh M0 is shown. The other meshes are refinements (starting from M0) using a factor of two in each coordinate direction, as compared to the previous refinement. The characteristics of the second series of meshes (M0r–M4r) are given in Table 2 and are similar to the corresponding meshes in Table 1, except that the level of refinement near the cylinder is extended to the complete center line. In Fig. 7, the starting mesh M0r is shown.

#### 4.2.3. Comparison of differential and integral UCM model at \(Wi = 0.5\)

First, we solve the problem for a differential UCM model at \(Wi = 0.5\). The numerical technique used is described in [11] (without LCR). After starting the flow we simulate up to \(t = 20a\) with a time step \(\Delta t = 0.1a\). Over the interval \(19a \leq t \leq 20a\), the relative change in the stress in the domain is \(\epsilon(10^{-3})\), which we assume is sufficient to have reached the steady state solution. In Fig. 8, we show the xx-component of the conformation tensor \(\epsilon\) on (part of) the center line and on the cylinder \((-1 < x/R < 1)\) for all the meshes M0–M4. The maximum difference between meshes M3 and M4 is 0.7 (around 1\%). We see, that convergence on the cylinder boundary is quite slow and very refined meshes are needed to have (visual) convergence, even for this relatively small Weissenberg number. This is related to the formation of a thin stress boundary layer near the cylinder as shown in Fig. 9. Convergence for the field variables \(p\) and the other conformation

<table>
<thead>
<tr>
<th>Mesh</th>
<th>(h_{cyl})</th>
<th>(h_{wall})</th>
<th>(h_{P=15R,wall})</th>
<th>(h_{P=15R,com})</th>
<th>(N_{elem})</th>
<th>(N_{elem,cyl})</th>
</tr>
</thead>
<tbody>
<tr>
<td>M0</td>
<td>0.08</td>
<td>0.4</td>
<td>1.2</td>
<td>1.2</td>
<td>953</td>
<td>40</td>
</tr>
<tr>
<td>M1</td>
<td>0.04</td>
<td>0.2</td>
<td>0.6</td>
<td>0.6</td>
<td>3467</td>
<td>79</td>
</tr>
<tr>
<td>M2</td>
<td>0.02</td>
<td>0.1</td>
<td>0.3</td>
<td>0.3</td>
<td>13,148</td>
<td>158</td>
</tr>
<tr>
<td>M3</td>
<td>0.01</td>
<td>0.05</td>
<td>0.15</td>
<td>0.15</td>
<td>52,053</td>
<td>315</td>
</tr>
<tr>
<td>M4</td>
<td>0.005</td>
<td>0.025</td>
<td>0.075</td>
<td>0.075</td>
<td>205,951</td>
<td>629</td>
</tr>
</tbody>
</table>

![Fig. 5. Geometry of the flow past a cylinder confined between two walls. Periodical boundary conditions (pbc) for velocity and stress (differential model) or deformation fields are applied between the inflow and outflow boundary.](image-url)
components are similar, i.e. the refined meshes M3–M4 are needed to have visual convergence in the plots. However, the velocity field is hardly affected and there is almost no difference between M0 and M4 for this Weissenberg number.

Next, we solve the same problem using an integral model with the F-approach. The discretization in r-direction is: \( N = 50, \Delta \tau = 0.06\lambda, \tau_e = 15\lambda, f = 1.05546. \) Again we simulate up to \( t = 20\lambda \) with a time step \( \Delta t = 0.1\lambda. \) In Fig. 10, we show the xx-component of the tensor \( \tau \) on (part of) the center line and on the cylinder (\( -1 < x/R < 1 \)) for all the meshes M0–M4. The maximum difference between meshes M3 and M4 is 1.1 (around 1.5 %). Note, that the values shown are the computed stress integrals in all the nodes based on the quadratic elements of the mesh. The projected stress plots (not shown) are somewhere in between the differential model and the integral model for meshes M0–M2. Similar to the differential model implementation, we see that meshes M3–M4 need to have visual convergence, although the error on coarser grids is somewhat different between the two approaches. However, both approaches converge (at least visually) to the same results for the meshes M3–M4. This confirms the consistency between the two methods and also supports that the implementation of the deformation fields method is correct.

### 4.2.4. Integral model with \( \alpha = 0.0001. \) Stability for higher Wi

In this subsection we will be studying the numerical stability of the proposed techniques. We will use the meshes M0–M2 of Table 1 for that, which already show the large difference in numerical stability of the two approaches: the F-approach and the B-approach.

We study a model with a damping function having \( \alpha = 0.0001. \) No corresponding differential model is available for comparison. The viscosity function is given in Fig. 11 and the planar extensional function in Fig. 12. The difference with the UCM model is that the viscosity is now shear thinning and the extensional viscosity is finite for all \( \epsilon. \) It should be noted, that at \( \dot{\gamma} = 129.8 \) the shear stress goes through a maximum value (of 24.79 G) for this single mode model, which turns up as a slope less than \(-1\) in the viscosity curve. Obviously (generalized) shear rates beyond this value should be avoided, since it would lead to extreme strain-rate localization at the cylinder wall when approaching the steady state flow. The planar extensional stress difference goes monotonically to a constant value (of 10000G), and therefore the slope goes to \(-1\) for large extensional rates. Note, that the vertical asymptote for the UCM model is at \( x/R = 0.5. \)

We study the stability for higher Wi of the two approaches using B and F. The discretization in r-direction is: \( N = 50, \Delta \tau = 0.06\lambda, \tau_e = 15\lambda, f = 1.05546. \) The time step used is \( \Delta t = 0.1\lambda. \) The results are given in Table 3. It is clear that the limiting Wi is quite low for the B-approach for this model and mesh-refinement does not seem to improve the stability limit. It should be noted, that the flow seems to become steady for a while and then suddenly there is a catastrophic crash (at \( t = 12.9\lambda, 18.9\lambda, \) and \( 23.9\lambda \) for M0, M1, M2, respectively for the higher value of the range). See Fig. 13 for the maximum value in the flow of the stress component \( \tau_{xx} \) as a function of time for M2 at Wi = 0.6.

We monitored the determinant of the stress tensor in the integration points and find that violation of the positive-definiteness of the stress tensor sets in at the front of the cylinder for only a few integration points and remains so for a while. After numerical stress fluctuations grow, massive violation in the front and wake of the cylinder sets in before the crash. Any conclusions about whether the violations are causing the fluctuations, or are a result of them, cannot be drawn from these observations.

The stability limit for the F-approach is much higher and this method is clearly preferred over the B-approach. The limiting value is mesh dependent, but also the type of unsteady flow for the higher value in the range is different. For M0, and M1 the flow becomes unsteady with periodic oscillations, whereas M2 has growing oscillations that eventually lead to a crash. It should be noted that the maximum value of the effective shear rate \( \gamma_{eff} \) is near 70 for Wi = 2.4 with mesh M1, lower than the value for the shear stress maximum. Note also, that we are only studying stability of the approaches here and that the used meshes do not resolve the flow at these high value of Wi. Much finer meshes would be needed to resolve the boundary layers and the stress wake behind the cylinder at these values of Wi. The observed instability could be a manifestation of an inherent flow/model instability. However, due to the inconsistent type of flow oscillations for different meshes, the observed unsteady/unstable flow is most likely to be a numerical instability. We will not further study this problem here.

### 4.2.5. Integral model with \( \alpha = 0.01, \beta = 0.5. \) High Wi

For simplicity, we want to restrict the memory function to a single discrete mode. In order to obtain a stable steady flow for high Wi for this case, the damping parameter \( \alpha \) needs to be increased to lower the stress growth in elongation. At the same time, this also results in a lower shear rate for the stress maximum in shear, invoking severe stress localisation for lower Wi. Therefore, we need to introduce a solvent

---

**Table 2**

Characteristics of meshes refined near the center line and the cylinder. The meshes are similar to the meshes in Table 1, except that the level of refinement near the cylinder is extended to the complete center line.

<table>
<thead>
<tr>
<th>Mesh</th>
<th>( h_{cy} )</th>
<th>( h_{wall} )</th>
<th>( h_{r150} )</th>
<th>( v_{150, elem} )</th>
<th>( N_{elem} )</th>
<th>( N_{elem, cyl} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>M0r</td>
<td>0.08</td>
<td>0.4</td>
<td>1.2</td>
<td>0.08</td>
<td>3415</td>
<td>40</td>
</tr>
<tr>
<td>M1r</td>
<td>0.04</td>
<td>0.2</td>
<td>0.6</td>
<td>0.04</td>
<td>11,491</td>
<td>79</td>
</tr>
<tr>
<td>M2r</td>
<td>0.02</td>
<td>0.1</td>
<td>0.3</td>
<td>0.02</td>
<td>42,482</td>
<td>158</td>
</tr>
<tr>
<td>M3r</td>
<td>0.01</td>
<td>0.05</td>
<td>0.15</td>
<td>0.01</td>
<td>165,097</td>
<td>315</td>
</tr>
<tr>
<td>M4r</td>
<td>0.005</td>
<td>0.025</td>
<td>0.075</td>
<td>0.005</td>
<td>657,875</td>
<td>629</td>
</tr>
</tbody>
</table>

---

**Fig. 6.** Mesh M0.

**Fig. 7.** Mesh M0r.

**Fig. 8.** Component \( c_{xx} \) of the conformation tensor on (part of) the centerline and on the cylinder (\( -1 < x/R < 1 \)) for all the meshes M0–M4. Differential model. Wi = 0.5.
viscosity $\eta$ to remove the stress maximum altogether. A broader spectrum (e.g. using multiple discrete modes) is also a possibility to achieve that, but this is beyond the scope of the current paper.

We will use a damping parameter of $a = 0.01$ and a solvent content $\beta = 0.5$. The shear viscosity and planar extensional viscosity functions are plotted in Figs. 14 and 15, respectively.

We solve the flow with the $F$-approach using the meshes in Table 2, since we need more resolution near the center line for high Wi. The discretization in $r$-direction is: $N = 50, \Delta r = 0.06 \lambda, \tau_c = 15 \lambda$. It turns out that the computations are stable and no ‘computational’ limit for Wi is present, as long as the time step is limited by the strain, i.e. $\Delta t U / R < 0.25$, or $\Delta t < 0.25 \lambda / Wi$, for high Wi. In fact we use $\Delta t = \min(0.1 \lambda, 0.25 R / U)$, where the ‘cross-over’ point is at $Wi = \lambda U / R = 2.5$. Note, that the number of time steps needed to obtain steady state increases linearly with the Weissenberg number. For example $Wi = 100$ requires 4000 steps to reach $\tau = 10 \lambda$, which can be quite expensive for refined meshes. The time step limitation due to the

---

**Table 3**

Range of Wi in which the flow becomes either unsteady or unstable. The low value is stable, the high value is unsteady/unstable.

<table>
<thead>
<tr>
<th></th>
<th>B-approach</th>
<th>F-approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>M0</td>
<td>0.4-0.5</td>
<td>2.2-2.3</td>
</tr>
<tr>
<td>M1</td>
<td>0.5-0.6</td>
<td>2.4-2.5</td>
</tr>
<tr>
<td>M2</td>
<td>0.5-0.6</td>
<td>2.0-2.1</td>
</tr>
</tbody>
</table>
strain is related to our choice to take the $L \cdot F$ term explicit in the time discretization. For not too high values of Wi (we tested up to Wi = 10) the flow becomes steady with relatively smooth stress fields, confirming the stability of the $F$-approach, resembling the stability of the log-conformation representation in differential models [25]. For high Wi (we tested Wi = 100), the flow becomes close to steady after a few relaxation times. Very small stress fluctuations remain, which are probably related to the small scale spatial stress variations remaining present. We did not study the high Wi case further, because it is way beyond the validity of a model with a single discrete mode.

To test the convergence with mesh refinement, we consider the case of Wi = 5. The flow is computed using a time step of $\Delta t = 0.05 \lambda$ using 200 steps (end time is 10$\lambda$). After starting the flow, there is a stress build-up within 1–4 relaxation times and then a transition to a steady state. This can be seen in Fig. 16, where the maximum and average value of the stress components in the domain are plotted as a function of time. The slight wiggle around $t = 5 \lambda$ in the maximum of $\tau_{yy}$ is probably a convection phenomenon due to the periodicity, where the fluid initially near the cylinder reaches the neighbour cylinder. This phenomena is there for all meshes and eventually dies out.

In Fig. 17, we show the distribution of $\tau_{xx}$ in the steady state. Two boundary layers, one on the cylinder and one on the wall, and a long stress wake are present. In Fig. 18, we show the value of $\tau_{xx}$ on the center line and on the cylinder wall for all the meshes used (M0r–M4r). We clearly see mesh convergence on the cylinder wall. However, convergence in the wake has not been achieved yet even for the very refined mesh Mr4. This confirms that convergence in the wake is difficult for high Weissenberg numbers, similar to results found in [25] for a Giesekus fluid. The stress development around the cylinder has an effect on the velocity near the cylinder. In Fig. 19, the velocity component $u_y/U$ is shown on the center line for all the meshes used (M0r–M4r). Also here convergence has not yet been achieved in the wake. Due to the viscoelastic stresses, there are velocity overshoots both in front and in the wake of the cylinder, where the latter is more significant. Note, that the Newtonian value of $u_y/U$ in a channel without the cylinder present would be 1.5, hence a little flattening of the profile is seen due to the mild shear thinning.

5. Discussion and conclusions

In this paper we have ‘upgraded’ the deformation fields method in [2] to have a numerical stability similar to the current state-of-the-art in finite element methods for differential viscoelastic fluid models. Flows in standard geometries (time-dependent shear flow between two plates and flow around a cylinder confined in a channel) have been simulated to study the stability and accuracy of the new implementation of the deformation fields method. The following techniques/methods have been introduced to obtain the improved stability:

- Spatial discretization for the deformation fields is using SUPG for stabilizing convection in real space $(x)$. 
- The convection in real space and in ‘fluid history’ space $(\tau)$ is taken implicitly to allow for larger time steps.
- A novel scheme, based on a differential form of the integral model, that decouples the momentum and mass balance from the constitutive model, allowing for low or zero-solvent viscosity.
- For obtaining the deformation history the ‘projected’ velocity gradient $G$ is used instead of the $(\nabla u)'$, leading to better transient stability.
- The deformation history is represented (and solved) by using the deformation gradient $F$ instead of the (Finger) deformation tensor $B$.

The first four techniques are more or less extensions to integral models of well known similar techniques for differential models. The last technique is different and calls for some more discussion. It basically replaces the log-conformation representation [12] (LCR) for differential models and gives similar stability. The reason for the stability is the same as the remarkable stability of stochastic models based on the method of Brownian configuration fields (BCF) versus conformation tensor models (without LCR). In [26], it has been argued that the stability of BCF comes from the formulation of the models in terms contravariantly deforming vectors $(Q)$:

$$\frac{\partial Q}{\partial t} = -u \cdot \nabla Q + L \cdot Q + \ldots$$

(32)

In elements near stagnation points $(u = 0)$ the first term on the right-
hand side (convection) balances the second term (deformation) quite naturally and numerical stability is obtained even on coarse grids. Conformation tensor models (without LCR) are formulated in terms of contravariantly deforming tensors:

$$\frac{\partial e}{\partial t} = -u \cdot \nabla c + L \cdot c + c \cdot L^T + \ldots$$  \hspace{1cm} (33)

In elements near stagnation points the first term on the right-hand side (convection) balances the second term (deformation), however there is an additional deformation term leading to numerical instability for underresolved grids. Since the deformation gradient $F$ behaves like contravariantly deforming vectors (see Eq. (9)) and the Finger tensor $B$ like contravariantly deforming tensors (see Eq. (12)), we can expect much better stability in the $F$-approach than in the $B$-approach. Note, that when applying the LCR [12] both deformation terms are basically ‘neutralized’, but from a stability point of view, this is not needed.

An additional benefit of the ‘vector’ based methods (BCF, $F$-approach) versus the ‘tensor’ based methods (conformation tensor, $B$-approach) is the guaranteed positive-(semi)definiteness of the stress tensor. Since positive-definiteness is required in the physical model, instabilities due to negative eigenvalues of the stress tensor can be avoided. Note, that LCR also leads to a positive definite stress tensor.

The next step is extending the range of constitutive models. In particular, we are interested in integral models with long-range memory functions, such as powerlaw functions [27] and powerlaw-like functions as appear in fractional K-BKZ models [5]. This will be part of future work.

Fig. 16. Maximum and average value of the stress tensor components in the domain for mesh M3r as a function of time. $a = 0.01, \eta_s = 1, Wi = 5$.

Fig. 17. Component $\tau_{xx}/G$ of the stress tensor near the cylinder and in the wake ($−2.75 ≤ x/R ≤ 15$) for mesh M3r. $a = 0.01, \eta_s = 1, Wi = 5$.

Fig. 18. Component $\tau_{xx}$ of the stress tensor (relative to $G$) on (part of) the centerline and on the cylinder ($−1 < x/R < 1$) for all the meshes M0r–M4r. Integral model with $F$-approach. $a = 0.01, \eta_s = 1, Wi = 5$.

Fig. 19. Velocity component $u_x$ (relative to $U$) on (part of) the centerline and on the cylinder ($−1 < x/R < 1$) for all the meshes M0r–M4r. Integral model with $F$-approach. $a = 0.01, \eta_s = 1, Wi = 5$. 
Appendix A. Differential form of the integral model

In this appendix, we will derive a ‘differential form’ of the integral model, given by Eq. (3), which we will repeat here for convenience:
\[
\dot{\tau}(t) = \int_{-\infty}^{t} m(t - t')h(t')B dt'.
\]  
(A.1)

For that, we need to differentiate the integral model with respect to time, where a single material particle is considered (Lagrangian frame). Note, that the upper limit, the memory function \(m(t - t')\) and the Finger tensor \(B(t, t') = F(t, t')F(t, t')^{\text{T}}\) (and also the invariant \(I_l\)) will depend on time. To obtain the (material) time derivative of \(\dot{\tau}\) we apply Leibniz’ integral rule:

\[
\frac{d}{dt} \int_{-\infty}^{t} f(t, t') dt' = f(t, t(t)) \frac{d}{dt} - f(t, a(t)) \frac{d}{dt} a(t) + \int_{-\infty}^{t} \frac{d}{dt} f(t, t') dt',
\]
(A.2)

which leads to:

\[
\dot{\tau}(t) = m(0)h(t)I + \int_{-\infty}^{t} \frac{dm(t)}{dt} h(t)B dt' + \int_{-\infty}^{t} m(t - t')h(t)B dt' + \int_{-\infty}^{t} m(t - t)h(t)B dt'.
\]
(A.3)

Using
\[
h(t) = 1, \quad B = L + B^{L'}, \quad \dot{h} = 2B: \dot{D},
\]
we obtain

\[
\dot{\tau}(t) = m(0)I + \int_{-\infty}^{t} \frac{dm(t)}{dt} h(t)B dt' + \int_{-\infty}^{t} m(t - t')h(t)B dt' + \int_{-\infty}^{t} m(t - t)h(t)B dt'L',
\]
(A.5)

which can be written in the compact form

\[\frac{d\tau}{dt} + \mathbf{u} \cdot \nabla \tau = m(0)\mathbf{I} + A + K: \mathbf{L'} \tau + \tau \cdot \mathbf{L'},\]  
(A.6)

with

\[\tau = \int_{-\infty}^{t} m(t)h(t)B dt,\]
(A.7)
\[A = \int_{-\infty}^{t} \frac{dm(t)}{dt} h(t)B dt,\]
(A.8)
\[K = \int_{-\infty}^{t} m(t)\frac{dh(t)}{dt}B^T B dt,\]
(A.9)

where we transformed the memory integrals from the domain \(t' \in (-\infty, 0)\) to \(t \in [0, \infty)\). Note, that we have also splitted the material derivative of the stress tensor into a local and convective part:

\[\dot{\tau} = \frac{d\tau}{dt} + \mathbf{u} \cdot \nabla \tau,\]
(A.10)

since in this paper we solve all equations in the Eulerian frame.

References


