Received: 20 April 2020 | Revised: 28 November 2020 | Accepted: 3 December 2020

DOI: 10.1002/fld.4950

RESEARCH ARTICLE

WILEY

A fully coupled high-order discontinuous Galerkin solver for viscoelastic fluid flow

Anne Kikker[1,2] | **Florian Kummer**[1,3] | **Martin Oberlack**[1,3]

[1]Chair of Fluid Dynamics, Technical University of Darmstadt, Darmstadt, Germany

[2]Graduate School of Computational Engineering, Technical University of Darmstadt, Darmstadt, Germany

[3]Centre for Computational Engineering, Technical University of Darmstadt, Darmstadt, Germany

Correspondence
Florian Kummer, Chair of Fluid Dynamics, Technical University of Darmstadt, Otto-Berndt-Str. 2, 64287 Darmstadt, Germany.
Email: kummer@fdy.tu-darmstadt.de

Funding information
Deutsche Forschungsgemeinschaft, Grant/Award Numbers: 265191195, 330615302

Abstract

A fully coupled high order discontinuous Galerkin (DG) solver for viscoelastic Oldroyd B fluid flow problems is presented. Contrary to known methods combining DG for the discretization of the convective terms of the material model with standard finite element methods (FEM) and using elastic viscous stress splitting (EVSS) and its derivatives, a local discontinuous Galerkin (LDG) formulation first described for hyperbolic convection-diffusion problems is used. The overall scheme is described, including temporal and spatial discretization as well as solution strategies for the nonlinear system, based on incremental increase of the Weissenberg number. The solvers suitability is demonstrated for the two-dimensional confined cylinder benchmark problem. The cylinder is immersed in a narrow channel with a blocking ratio of 1:2 and the drag force of is compared to results from the literature. Furthermore, steady and unsteady calculations give a brief insight into the characteristics of instabilities due to boundary layer phenomena caused by viscoelasticity arising in the narrowing between channel and cylinder.

KEYWORDS

artificial viscosity, confined cylinder, discontinuous Galerkin, local discontinuous Galerkin, Oldroyd B, viscoelastic flow

1 | INTRODUCTION

In many flow processes, the behavior of the fluid is non-Newtonian. Especially in biological flow and environmental or chemical processes, such as blood flow,[1,2] sedimentation processes,[3] or polymer melts,[4] we can observe viscoelastic flow phenomena like the characteristic fading memory effect.

The simulation of fluids with viscoelastic behavior is particularly challenging due to two major issues: First, the proper modeling of the physical characteristics and the right choice of a viscoelastic model is crucial in order to not develop unphysical models or even ill-posed problems with non-unique or even without solutions.[5] Existing models such as the Oldroyd B model used in this study are convection-dominated with a missing diffusion term in the constitutive equations. Thus, they are of hyperbolic nature.[6] Second, the numerical method has to deal with stability and accuracy problems caused by a strong variation of length scales within the thin boundary layer, where velocity gradients and stresses can rapidly change their values by several orders of magnitude.[7] Both leads to a loss of convergence even for minimal elastic behavior in the fluid.

A good approach is the use of the discontinuous Galerkin (DG) method having discontinuous elements with appropriate flux functions for the edges, which is more robust against numerical oscillations compared to, for example, continuous Galerkin methods. Within the last 25 years, the DG method has been successfully established for solving hyperbolic conservation laws and was first introduced by Fortin and Fortin[8] for viscoelastic fluid flow. It is also strongly emerging in other fields of computational fluid dynamics.[9] There are two reasons for this ascent which obviates essential limitations of classical techniques such as finite volume or finite difference methods. DG cleverly combines an arbitrary order $k \in \mathbb{N}$ in the numerical discretization error $\mathcal{O}(h^{k+1})$ with a local flux evaluation which is at most to be computed from adjacent cells. Here, h refers to the local grid spacing, and k to the order of the DG basis polynomials. This is in strong contrast to the established schemes, which are substantially limited to a convergence order of two on unstructured grids, and even on Cartesian grids it is rather limited to small values because of the larger stencils required for increasing convergence order.[10]

In viscoelastic flow, the DG method is often used to obtain stability for the convection-dominated convection-diffusion problem using a streamline upwinding formulation. To this end, the convective term of the constitutive equations is discretized by a DG scheme whereas the other variables in the momentum and continuum equation are discretized using a standard finite element method (FEM). The DG method allows jumps in the boundary conditions and preconditioning at the elemental level, appropriate flux functions for the edges can be chosen, and velocity-stress compatibility conditions can be easily satisfied.[5] So the DG method is a promising method for convection-dominated problems. However, in the context of viscoelastic flow, there are few approaches fully based on this method. First ideas for high order DG in a decoupled scheme can be found in the newer work of Mirzakhalili and Nejat.[11]

A breakdown in convergence can also occur due to the mixed hyperbolic-elliptic type of the system of equations, whereas the saddle-point problem of the Navier-Stokes system is of elliptic type. The constitutive equations modeling the viscoealstic behavior are hyperbolic. As it is shown in Section 2, the viscous part of the momentum equations and of the constitutive equations are weighted by a material parameter β. If β is close to 1, the contribution of the constitutive equations is small and we have an elliptic system to solve. If $\beta \to 0$ such that we have a highly elastic fluid for increasing Weissenberg number, a change of type from elliptic to hyperbolic occurs and the numerical solution can become unstable unless special care is taken.[6,12,13] The Weissenberg number relates the elastic to viscous forces such that a large Weissenberg number describes significant non-Newtonian behavior if the elastic forces outweigh the viscous forces.

There are several approaches in numerical computation for handling the strong mixed hyperbolic-elliptic coupling between the momentum and constitutive equations by the velocity gradient. In the elastic-viscous stress splitting (EVSS) method and its derivatives, a second order elliptic term is introduced in the constitutive equations and the depending variables are changed such that there is no necessity for additional compatibility conditions for the well-posedness of the discrete system in the Stokesian limit.[5,7,14,15] However, this extends the system of equations to be solved by an additional evolutionary equation for the velocity gradient.

We aim for a solver for viscoelastic flow with an exclusively high order DG scheme for all equations using a local discontinuous Galerkin (LDG) formulation with penalized fluxes in order to solve the hyperbolic constitutive equations and using a streamline upwinding for the convective fluxes of the constitutive equations. The solver is embedded in the open source DG framework called *BoSSS*, currently under development at the chair of fluid dynamics of Technical University of Darmstadt, which can be downloaded under https://github.com/FDYdarmstadt/BoSSS.

Since only few analytical solutions exist for complex viscoelastic flow, several benchmark problems have been established for validating numerical schemes and models. One of these is the confined cylinder problem in which the flow around a cylinder immersed in a narrow channel with a blocking ratio of 2:1 is investigated numerically. This benchmark has been analysed in numerous works, for example, References 7,16-21, but still, the numerical treatment of this test case is challenging because there are problems arising in resolving steep gradients in velocity and stress and very thin boundary layers as well as a very fine nearly singular beam in the wake of the cylinder.[20] Without stabilizing techniques, most numerical approaches diverge for Weissenberg numbers (Wi) greater than 0.7, for example, Reference 14. The lack of convergence is due to either numerical errors propagating into the wake of the cylinder or physical instabilities inherent in the viscoelastic model which lead to an exponential increase of the stresses at the rear stagnation point behind the cylinder.[20] To circumvent this, among other methods, a log-conformation formulation of the stress tensor was developed,[22,23] which leads to higher achievable Weissenberg numbers, but also leads to a higher complexity of the system and does not avoid the loss of convergence in the wake of the cylinder.

In newer times, the physical instabilities in the viscoelastic flow around the cylinder have come into focus of research. Experimentally, such instabilities were confirmed and analyzed in detail earlier.[24] There seems to be an unstable detaching boundary layer on the cusp of the cylinder leading to a transitional flow which is unsteady beyond $\mathrm{Wi} = 1$ and three-dimensional structures arise. This means that with increasing Weissenberg number, a convergent numerical

solution cannot be found, especially when fine meshes and high-order accurate methods try to capture the evolving structures close to the cylinder. This could be the reason why solutions on coarser meshes might have less difficulty to converge.[20]

In this work, we introduce in a first step the high order LDG method for viscoelastic flow. We then present some results for the confined cylinder benchmark problem. If there are problems finding a convergent solution, we can use a cell indicator combined with artificial diffusion. This method was originally developed similar to the LDG formulation for DG methods for compressible Navier-Stokes applications.[25] In the last part of this work, results for the steady and unsteady viscoelastic flow around the cylinder are shown and compared to results obtained by other numerical methods. In the end the reader will find some concluding remarks concerning the solver and the confined cylinder benchmark problem.

2 | GOVERNING EQUATIONS

We consider two-dimensional incompressible viscoelastic Oldroyd B fluid flow in dimensionless variables consisting of the continuity equation

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

the momentum equations including the solvent part of the fluid

$$\mathrm{Re}\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u}\right) = -\nabla p + \beta \Delta \mathbf{u} + \nabla \cdot \boldsymbol{\tau}, \tag{2}$$

and the constitutive equations for the polymeric part of the Oldroyd B fluid

$$\boldsymbol{\tau} + \mathrm{Wi}\, \overset{\nabla}{\boldsymbol{\tau}} = (1-\beta)\left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T\right). \tag{3}$$

In this system, we have the dependent variables $\mathbf{u}$: velocity vector, p: pressure, and $\boldsymbol{\tau}$: extra stress tensor, the independent variables t: time, the Laplace operator $\Delta = \nabla^2$, the nabla operator $\nabla = \frac{\partial}{\partial \mathbf{x}}$, and $\mathbf{x}$: position vector, and the dimensionless parameters Re: Reynolds number, Wi: Weissenberg number, β: ratio between solvent and total viscosity ($\beta = \frac{\eta_s}{\eta_0}$, $(1-\beta) = \frac{\eta_p}{\eta_0}$). We use the upper convected objective derivative:

$$\overset{\nabla}{\boldsymbol{\tau}} = \frac{\partial \boldsymbol{\tau}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\tau} - \nabla \mathbf{u} \cdot \boldsymbol{\tau} - \boldsymbol{\tau} \cdot (\nabla \mathbf{u})^T. \tag{4}$$

The constitutive model reduces to the upper convected Maxwell model (Maxwell fluid B) if $\beta = 0$ and to a Newtonian fluid if $\beta = 1$ which means that the polymeric viscosity is zero. If the Weissenberg number is chosen to be zero, this leads to a Newtonian formulation as well. The system (1)–(3) is of mixed type: the Stokes part in (1), (2), neglecting the left-hand-side of (2) is elliptic. If a temporal derivative is considered, it is parabolic, while the convective part on the left-hand-side of (2) is hyperbolic. The Oldroyd B model (3)) is also of mixed type; Joseph[13] shows that it can be hyperbolic under certain circumstances, but it can also be parabolic in other regions. However, it is obvious that (1)–(3) is a highly nonlinear, coupled problem. In combination with its non-hyperbolic nature, this requires an implicit non-linear solver.

3 | COMPUTATIONAL METHODS

If we want to discretize the governing equations (1)–(3), we need to define initial and boundary conditions on the boundary $\partial\Omega$ of a physical domain $\Omega \subset \mathbb{R}^2$ for the dependent variables in order to obtain a well-posed problem. The initial conditions are:

$$\mathbf{u} = \mathbf{u}_0, \quad \boldsymbol{\tau} = \boldsymbol{\tau}_0 \quad \text{for } t = t_0. \tag{5}$$

On Dirichlet boundaries $\partial\Omega_D$, we define:

$$\mathbf{u} = \mathbf{u}_D \quad \text{on } \partial\Gamma_D, \tag{6}$$

on boundaries with a condition of the Neumann type $\partial\Gamma_N$ we have:

$$\left(-p\boldsymbol{I} + \beta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) + \boldsymbol{\tau}\right)\ \mathbf{n}_{\partial\Omega} = 0 \quad \text{on } \partial\Omega_N, \tag{7}$$

and we introduce a free slip boundary condition on $\partial\Gamma_S$ with:

$$\begin{aligned} \mathbf{u}\cdot\mathbf{n}_{\partial\Omega} &= 0 \quad \text{and} \\ \boldsymbol{P}_S\left(\beta(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) + \boldsymbol{\tau}\right)\ \mathbf{n}_{\partial\Omega} &= 0 \quad \text{on } \partial\Omega_S, \end{aligned} \tag{8}$$

where $\boldsymbol{P}_S = \boldsymbol{I} - \mathbf{n}_{\partial\Omega}\otimes\mathbf{n}_{\partial\Omega}$ denotes the orthogonal projection onto the boundary $\partial\Omega_S$.

3.1 | Discretization

Before describing the DG discretization of the system of equations, we first introduce some definitions.[26]

Corresponding to the physical domain defined above, we have a computational domain $\Omega_h \subset \mathbb{R}^2$ with its boundary $\partial\Omega_h$ which must be polygonal and simply connected. On Ω_h, we have a space filling triangulation as a numerical grid with geometry-conforming non-overlapping elements $K_h = \{K_1, \ldots, K_N\}$ with a characteristic mesh size h. So the physical domain Ω is approximated by the union of all elements: $\Omega \approx \Omega_h = \bigcup_{i=1}^{N} K_i$, whereas $\Gamma = \bigcup_j \partial K_j$ is the union of all edges or the skeleton of the grid and $\Gamma_I = \Gamma \setminus \partial\Omega_h$ is the union of all interior edges. Furthermore, we have $\Gamma_{D,h} \subset \Omega_h$ and $\Gamma_{D,h} \approx \Gamma_D$, $\Gamma_{N,h} \subset \Omega_h$ and $\Gamma_{N,h} \approx \Gamma_N$, and $\Gamma_{S,h} \subset \Omega_h$ and $\Gamma_{S,h} \approx \Gamma_S$ for the Dirichlet, Neumann, and free slip boundary, respectively.

We denote a normal field $\mathbf{n}_\Gamma$, on $\partial\Omega_h$ it defines an outer normal. We now can define the direction of information transfer. Therefore, the superscript $\mathbf{u}^-$ defines the information in the interior of an element while the superscript $\mathbf{u}^+$ defines the exterior information of the neighboring cell for a field $\mathbf{u} \in C^0(\Omega_h \setminus \Gamma_I)$ with:

$$\mathbf{u}^- = \lim_{\xi \searrow 0} \mathbf{u}(\mathbf{x} - \xi\mathbf{n}_\Gamma) \quad \text{for } \mathbf{x} \in \Gamma \tag{9}$$

$$\mathbf{u}^+ = \lim_{\xi \searrow 0} \mathbf{u}(\mathbf{x} + \xi\mathbf{n}_\Gamma) \quad \text{for } \mathbf{x} \in \Gamma_I. \tag{10}$$

Since $\mathbf{n}^- = -\mathbf{n}^+$, the jump and mean values of a component of a tensor of arbitrary order on inner edges Γ_I are, respectively:

$$[\![\mathbf{u}]\!] = \mathbf{u}^+ - \mathbf{u}^- \tag{11}$$

$$\{\mathbf{u}\} = \frac{1}{2}\left(\mathbf{u}^- + \mathbf{u}^+\right). \tag{12}$$

The jump and mean values on boundary edges $\partial\Omega_h$ are, respectively:

$$[\![\mathbf{u}]\!] = \mathbf{u}^- \tag{13}$$

$$\{\mathbf{u}\} = \mathbf{u}^-. \tag{14}$$

We now define the broken polynomial space of total degree k as

$$\mathbb{P}_k(K_h) = \{f \in L^2(\Omega_h); \forall K \in K_h : f|_{K_h} \text{ is polynomial and } \deg\left(f|_{K_h}\right) \leq k\}, \tag{15}$$

and the function space for test and trial functions for D_v dependent variables with discretization of different polynomial degree k. Therefore, we define the degree-vector $\mathbf{k} = \left(k_1, \ldots, k_{D_v}\right)$ and get:

$$\mathbb{V}_{\mathbf{k}} = \prod_{i=1}^{D_v} \mathbb{P}_{k_i}(K_h). \tag{16}$$

We define for $u_K, v_K \in \mathbb{V}_{\mathbf{k}}$ a local inner product and a local $L^2(K_h)$ norm on each element:

$$(u_K, v_K)_K = \int_K u_K v_K \, dx \quad \|u_K\|_K^2 = (u_K, u_K)_K, \tag{17}$$

and for $u_h, v_h \in \mathbb{V}_{\mathbf{k}}$, a global broken inner product and a global broken norm:

$$(u_h, v_h)_{\Omega_h} = \sum_{i=1}^{N} (u_h, v_h)_K \quad \|u_h\|_{\Omega_h}^2 = (u_h, u_h)_{\Omega_h}. \tag{18}$$

For $\mathbf{u}_h \in C^1(\Omega_h)$, the broken gradient is defined by $\nabla_h \mathbf{u}_h$ on the domain Ω_h where differentiation at the jumps on Γ is excluded and analogously, $\nabla_h \cdot \mathbf{u}_h$ is the broken divergence.

3.1.1 | Temporal discretization

Although most numerical examples presented here (cf. Section 4) are steady-state solutions, temporally dependent solutions, such as shown in Section 4.3 (see also Figure 11), can be obtained by using a backward difference method of second order (BDF2) for the temporal discretization of Equations (2) and (3). In the examples investigated, we found BDF2 to be a reasonable compromise between temporal accuracy and stability of the temporal integration. Furthermore, it is also easy to implement. This leads to the following system:

$$\begin{aligned}
&\nabla \cdot \mathbf{u}^{n+1} = 0 \\
&\mathrm{Re}\left(\frac{3\mathbf{u}^{n+1}}{2\Delta t} + \mathbf{u}^{n+1} \cdot \nabla \mathbf{u}^{n+1}\right) + \nabla p^{n+1} - \beta \Delta \mathbf{u}^{n+1} - \nabla \cdot \boldsymbol{\tau}^{n+1} = \frac{\mathrm{Re}}{\Delta t}\left(2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1}\right) \\
&\boldsymbol{\tau}^{n+1} + \mathrm{Wi}\left(\frac{3\boldsymbol{\tau}^{n+1}}{2\Delta t} + \mathbf{u}^{n+1} \cdot \nabla \boldsymbol{\tau}^{n+1} - \nabla \mathbf{u}^{n+1} \cdot \boldsymbol{\tau}^{n+1} - \boldsymbol{\tau}^{n+1} \cdot \left(\mathbf{u}^{n+1}\right)^T\right) \\
&\quad - (1-\beta)\left(\nabla \mathbf{u}^{n+1} + \left(\nabla \mathbf{u}^{n+1}\right)^T\right) = \frac{\mathrm{Wi}}{\Delta t}\left(2\boldsymbol{\tau}^n - \boldsymbol{\tau}^{n-1}\right).
\end{aligned} \tag{19}$$

3.1.2 | Spatial discretization

Since we need to satisfy the Ladyzenskaja-Babuska-Brezzi (LBB) condition for the Stokes system with $\beta = 1$, we use polynomials of degree k for velocity and stresses and of degree $k' = k - 1$ for the pressure.[27,28] Detailed work on the stability of the DG method for the Stokes and Navier-Stokes System can be found in Girault et al.[29]

Regarding the temporal discretization, to simplify the notation, let all properties in (19) which are related to times t^n and t^{n-1} (e.g., $\mathbf{u}^n$ and $\mathbf{u}^{n-1}$), be absorbed in the right-hand sides. For steady-state simulations, one can assume the limit $\Delta t \to \infty$, that is, $1/\Delta t = 0$. Then, the DG discretization of the governing system of equations is: Find $(\mathbf{u}_h, p_h, \boldsymbol{\tau}_h) \in \mathbb{V}_{\mathbf{k}}$ such that for all $(\mathbf{v}_h, q_h, \boldsymbol{\sigma}_h = \boldsymbol{\sigma}_h^T) \in \mathbb{V}_{\mathbf{k}}$:

$$b(\mathbf{u}_h, q_h) + s_2(p_h, q_h) = r_1(q_h), \tag{20}$$

$$\frac{3\mathrm{Re}}{2\Delta t}(\mathbf{u}_h, \mathbf{v}_h) + c(\mathbf{u}_h, \mathbf{u}_h, \mathbf{v}_h) + b(p_h, \mathbf{v}_h) - a(\mathbf{u}_h, \mathbf{v}_h) - d'(\mathbf{v}_h, \boldsymbol{\tau}_h) + s_1(\mathbf{u}_h, \mathbf{v}_h) = r_2(\mathbf{v}_h), \tag{21}$$

$$\begin{aligned}
&\left(1 + \frac{3\mathrm{Wi}}{2\Delta t}\right)(\boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) + f(\mathbf{u}_h, \boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) - g(\mathbf{u}_h, \boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) - g(\mathbf{u}_h, \boldsymbol{\tau}_h, \boldsymbol{\sigma}^T{}_h) \\
&\quad - d(\mathbf{u}_h, \boldsymbol{\sigma}_h) - d(\mathbf{u}_h, \boldsymbol{\sigma}_h^T) = r_3(\boldsymbol{\sigma}_h).
\end{aligned} \tag{22}$$

For the trilinear form of the convective term in the momentum equations, we use a Lax-Friedrichs flux:[30]

$$c(\mathbf{w}_h, \mathbf{u}_h, \mathbf{v}_h) = -\int_{\Omega_h} \mathrm{Re}(\mathbf{u}_h \otimes \mathbf{w}_h) \,:\, \nabla_h \mathbf{v}_h \,\mathrm{dV} - \oint_{\Gamma \setminus \Gamma_{D,h} \setminus \Gamma_{S,h}} \mathrm{Re}\left(\{\mathbf{u}_h \otimes \mathbf{w}_h\} \,\mathbf{n}_\Gamma - \frac{\gamma_1}{2} [\![\mathbf{u}_h]\!] \right) \cdot [\![\mathbf{v}_h]\!] \,\mathrm{dS}. \tag{23}$$

Further information about the choice of the Lax-Friedrichs parameter γ_1 in the DG discretization are given in Shahbazi et al.[30,31] For the bilinear form of the pressure gradient as well as for the velocity divergence in the continuity equation, we use

$$b(p_h, \mathbf{v}_h) = -\int_{\Omega_h} p_h \,\nabla_h \cdot \mathbf{v}_h \,\mathrm{dV} - \oint_{\Gamma \setminus \Gamma_{N,h}} [\![\mathbf{v}_h]\!] \cdot \mathbf{n}_\Gamma \,\{p_h\} \,\mathrm{dS}. \tag{24}$$

The Laplacian in the momentum equations is discretized using the symmetric interior penalty (SIP) method:[31]

$$\begin{aligned} a(\mathbf{u}_h, \mathbf{v}_h) = &-\int_{\Omega_h} \beta(\nabla_h \mathbf{u}_h \,:\, \nabla_h \mathbf{v}_h) \,\mathrm{dV} + \oint_{\Gamma \setminus \Gamma_{N,h} \setminus \Gamma_{S,h}} \beta \{\nabla_h \mathbf{u}_h\} \,\mathbf{n}_\Gamma \cdot [\![\mathbf{v}_h]\!] \,\mathrm{dS} \\ &+ \oint_{\Gamma \setminus \Gamma_{N,h} \setminus \Gamma_{S,h}} \beta \{\nabla_h \mathbf{v}_h\} \,\mathbf{n}_\Gamma \cdot [\![\mathbf{u}_h]\!] \,\mathrm{dS} - \oint_{\Gamma \setminus \Gamma_{N,h} \setminus \Gamma_{S,h}} \gamma_2 [\![\mathbf{u}_h]\!] \cdot [\![\mathbf{v}_h]\!] \,\mathrm{dS} + a_S(\mathbf{u}_h, \mathbf{v}_h). \end{aligned} \tag{25}$$

The penalty parameter γ_2 has to be chosen large enough to ensure coercivity while it should be as small as possible since over-penalization increases the approximation error. Further information about the choice of this penalty parameter being proportional to $(p+1)^2$) in the *BoSSS* framework are given in the work of Hillewaert.[32] The free slip boundary condition in the SIP flux $a(\mathbf{u}_h, \mathbf{v}_h)_S$ for $\Gamma_{S,h}$ reads:

$$\begin{aligned} a_S(\mathbf{u}, \mathbf{v}) = &\oint_{\Gamma_{S,h}} \beta \,\mathbf{n}_{\Gamma_{S,h}} \cdot \left(\{\nabla_h \mathbf{u}_h\} \,\mathbf{n}_{\Gamma_{S,h}} \right) [\![\mathbf{v}_h]\!] \cdot \mathbf{n}_{\Gamma_{S,h}} \,\mathrm{dS} \\ &+ \oint_{\Gamma_{S,h}} \beta \,\mathbf{n}_{\Gamma_{S,h}} \cdot \left(\{\nabla_h \mathbf{v}_h\} \,\mathbf{n}_{\Gamma_{S,h}} \right) [\![\mathbf{u}_h]\!] \cdot \mathbf{n}_{\Gamma_{S,h}} \,\mathrm{dS} - \oint_{\Gamma_{S,h}} \gamma_2 [\![\mathbf{u}_h]\!] \cdot \mathbf{n}_{\Gamma_{S,h}} [\![\mathbf{v}_h]\!] \cdot \mathbf{n}_{\Gamma_{S,h}} \,\mathrm{dS}. \end{aligned} \tag{26}$$

For the stress divergence in the momentum equations and the gradient of velocity in the constitutive equations, we have structurally the same weak formulation, for the stress divergence it reads:

$$d'(\boldsymbol{\tau}_h, \mathbf{v}_h) = -\int_{\Omega_h} \boldsymbol{\tau}_h \,:\, \nabla_h \mathbf{v}_h \,\mathrm{dV} + \oint_{\Gamma \setminus \Gamma_{N,h} \setminus \Gamma_{S,h}} \{\boldsymbol{\tau}_h\} \,\mathbf{n}_\Gamma \cdot [\![\mathbf{v}_h]\!] \,\mathrm{dS} + d_S(\boldsymbol{\tau}_h, \mathbf{v}_h), \tag{27}$$

and for the gradient of velocity in the constitutive equations, it is:

$$d(\mathbf{u}_h, \boldsymbol{\sigma}_h) = -\int_{\Omega_h} (1-\beta) \,\mathbf{u}_h \cdot \nabla_h \cdot \boldsymbol{\sigma}_h \,\mathrm{dV} + \oint_{\Gamma} (1-\beta) \,\left(\{\mathbf{u}_h\} \otimes \mathbf{n}_\Gamma \right) \,:\, [\![\boldsymbol{\sigma}_h]\!] \,\mathrm{dS}. \tag{28}$$

In case of the stress divergence, we need to define a free slip boundary condition $d_S(\boldsymbol{\tau}_h, \mathbf{v}_h)$ at $\Gamma_{S,h}$:

$$d_S(\boldsymbol{\tau}_h, \mathbf{v}_h) = \oint_{\Gamma_{S,h}} \mathbf{n}_{\Gamma_{S,h}} \cdot \left(\{\boldsymbol{\tau}_h\} \,\mathbf{n}_{\Gamma_{S,h}} \right) [\![\mathbf{v}_h]\!] \cdot \mathbf{n}_{\Gamma_{S,h}} \,\mathrm{dS}. \tag{29}$$

The trilinear form for the convective term in the constitutive equations including the streamline upwinding[8] is

$$f(\mathbf{w}_h, \boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) = -\int_{\Omega_h} \mathrm{Wi} \,(\mathbf{w}_h \cdot \nabla_h \boldsymbol{\tau}_h) \,:\, \boldsymbol{\sigma}_h \,\mathrm{dV} - \oint_{\Gamma_I} \mathrm{Wi} \left(\{\mathbf{w}_h\} \cdot \mathbf{n}_{\Gamma_I} \right) \left([\![\boldsymbol{\tau}_h]\!] \,:\, \left(f_2 \boldsymbol{\sigma}_h^+ - f_1 \boldsymbol{\sigma}_h^- \right) \right) \,dS, \tag{30}$$

with the following functions depending on the upwind parameter $0 \leq \alpha \leq 1$:

$$f_1(\alpha) = \begin{cases} \alpha, & \text{if } \{\mathbf{w}_h\} \cdot \mathbf{n}_{\Gamma_I} < 0, \\ 1-\alpha, & \text{if } \{\mathbf{w}_h\} \cdot \mathbf{n}_{\Gamma_I} \geq 0 \end{cases},$$
$$f_2(\alpha) = \begin{cases} \alpha, & \text{if } \{\mathbf{w}_h\} \cdot \mathbf{n}_{\Gamma_I} < 0, \\ 1-\alpha, & \text{if } \{\mathbf{w}_h\} \cdot \mathbf{n}_{\Gamma_I} \geq 0 \end{cases}. \tag{31}$$

The objective terms of the constitutive equations consist of the following trilinear form:

$$g(\mathbf{w}_h, \boldsymbol{\tau}_h, \boldsymbol{\sigma}_h) = -\int_{\Omega_h} \mathrm{Wi}\,(\nabla_h \mathbf{w}_h \cdot \boldsymbol{\tau}_h) : \boldsymbol{\sigma}_h \,\mathrm{dV}. \tag{32}$$

In the LDG method, we need a penalty flux to ensure the stability of the system.[9] The bilinear form read

$$s_1(\mathbf{u}_h, \mathbf{v}_h) = -\oint_{\Gamma \setminus \Gamma_{N,h} \setminus \Gamma_{S,h}} \frac{\gamma_3}{h_{min}} [\![\mathbf{u}_h]\!] \cdot [\![\mathbf{v}_h]\!] \,\mathrm{dS}. \tag{33}$$

The right-hand sides of the problem include the Dirichlet boundary values and all values from the BDF2 discretization of time level t^n and t^{n-1}. The right-hand side for the continuity equation is:

$$r_1(q_h) = -\oint_{\Gamma_{D,h}} q_h \, \mathbf{u}_D \cdot \mathbf{n}_{\Gamma_{D,h}} \,\mathrm{dS}. \tag{34}$$

The right-hand side of the momentum equations is:

$$\begin{aligned} r_2(\mathbf{v}_h) = &\oint_{\Gamma_{D,h}} ((\mathbf{u}_D \otimes \mathbf{u}_D)\, \mathbf{n}_{\Gamma_{D,h} \cup \Gamma_{S,h}} + \frac{\gamma_1}{2} \mathbf{u}_D) \cdot [\![\mathbf{v}_h]\!] \,\mathrm{dS} - \oint_{\Gamma_{D,h}} \mathbf{u}_D \cdot \left(\beta \{\nabla_h \mathbf{v}_h\}\, \mathbf{n}_{\Gamma_{D,h}} - \gamma_2 [\![\mathbf{v}_h]\!]\right) \,\mathrm{dS} \\ &- \oint_{\Gamma_{D,h}} \frac{\gamma_3}{h_{min}} \mathbf{u}_D \cdot [\![\mathbf{v}_h]\!] \,\mathrm{dS} + \frac{\mathrm{Re}}{\Delta t} \int_{\Omega_h} \left(2\mathbf{u}^n - \frac{1}{2}\mathbf{u}^{n-1}\right) \cdot \mathbf{v}_h \,\mathrm{dV}. \end{aligned} \tag{35}$$

It includes the Dirichlet boundary condition for the convective part, for the Laplace term and for the penalty flux s_1 and the values for the velocities from the BDF2 discretization of time level t^n and t^{n-1}.

The right-hand side for the constitutive equations consists of a Dirichlet boundary for the velocities in the gradient of velocity and the values for the stresses from the BDF2 discretization of time level t^n and t^{n-1}:

$$r_3(\boldsymbol{\sigma}_h) = -\oint_{\Gamma_{D,h}} (1-\beta)\, \mathbf{u}_D \otimes \mathbf{n}_\Gamma : [\![\boldsymbol{\sigma}_h]\!] \,\mathrm{dS} - \oint_{\Gamma_{D,h}} (1-\beta)\, \mathbf{u}_D \otimes \mathbf{n}_\Gamma : [\![\boldsymbol{\sigma}_h^T]\!] \,\mathrm{dS} + \frac{\mathrm{Wi}}{\Delta t} \int_{\Omega_h} \left(2\boldsymbol{\tau}^n - \frac{1}{2}\boldsymbol{\tau}^{n-1}\right) : \boldsymbol{\sigma}_h \,\mathrm{dV}. \tag{36}$$

Since the stress tensor is symmetric and we choose the test function to be $\boldsymbol{\sigma}_h = \boldsymbol{\sigma}_h^T$, we can omit the constitutive equation for τ_{21} and therefore, neglect all terms belonging to τ_{21} in the discretized system.

3.2 | Nonlinear solver

The presented solver, which is exclusively using a DG scheme in the context of viscoelastic flow, is implemented in the software framework *BoSSS*[33] and due to its future application in multiphase flows embedded in the extended discontinuous Galerkin (XDG) solver presented by Kummer.[26] The overall solver consist of several levels with different iteration schemes:

1. a BDF2-time stepping scheme for transient simulations (cf. Section 4.3), where the limit $1/\Delta t = 0$ yields a stationary solver,
2. a homotopy scheme with slowly increasing Weissenberg number, and

3. a Newton scheme, employing a Dogleg-globalization, see References 34 and 35, and a direct sparse solver for the linearized system.

3.2.1 | Preparations

In order to solve the nonlinear variational problem (20)–(22) by a Newton method, it is helpful to first convert it into a more compact notation: First, we note that Equations (20)–(22) can be written in the following form: Find $\mathbf{U}_h \in \mathbb{V}_\mathbf{k}$

$$\mathcal{N}(\mathbf{U}_h, \mathbf{V}_h) = 0 \quad \forall\, \mathbf{V}_h \in \mathbb{V}_\mathbf{k}, \tag{37}$$

where, for $\mathbf{U}_h = (\mathbf{u}_h, p_h, \boldsymbol{\tau}_h)$ and $\mathbf{V}_h = (\mathbf{v}_h, q_h, \boldsymbol{\sigma}_h)$, the form $\mathcal{N}(\mathbf{U}_h, \mathbf{V}_h)$ is the sum over all left-hand-sides minus the sum over all right-hand-sides of Equations (20)–(22). We assume a basis $\underline{\boldsymbol{\Phi}} = (\boldsymbol{\Phi}_1, \dots, \boldsymbol{\Phi}_L)$ of $\mathbb{V}_\mathbf{k}$, written as a row vector, with $L := \dim(\mathbb{V}_\mathbf{k})$. Then $\mathbf{U}_h$ can be represented as $\mathbf{U}_h = \underline{\boldsymbol{\Phi}} \cdot \mathbf{U}$. The nonlinear problem (37) can then be written as

$$\mathcal{A}(\mathbf{U}) = 0, \tag{38}$$

with the nonlinear function $\mathbb{R}^L \ni \mathbf{U} \mapsto \mathcal{A}(\mathbf{U}) \in \mathbb{R}^L$. The ith component of $\mathcal{A}(\mathbf{U})$ can be defined by $\mathcal{N}(-,-)$ through the relation $[\mathcal{A}(\mathbf{U})]_i = \mathcal{N}(\underline{\boldsymbol{\Phi}} \cdot \mathbf{U}, \boldsymbol{\Phi}_i)$.

3.2.2 | The Jacobian of $\mathcal{A}$

In order to formulate a Newton method, one also requires the Jacobian matrix $\partial\mathcal{A}$ of $\mathcal{A}$, defined as

$$\partial\mathcal{A}_{ij}(\mathbf{U}) := \frac{\partial \mathcal{A}_i}{\partial U_j}(\mathbf{U}). \tag{39}$$

Its computation is quite straightforward, but lengthy; the *BoSSS* code is capable of evaluating the Jacobian matrix automatically from the expressions given in Section 3.1.2. We note that one could write $\mathcal{A}(\mathbf{U})$ as

$$[\mathcal{A}(\mathbf{U})]_i = \mathcal{N}(\mathbf{U}_h, \boldsymbol{\Phi}_i) = \int_{\Omega_h} N_1(\vec{x}, \mathbf{U}_h, \nabla\mathbf{U}_h) \cdot \boldsymbol{\Phi}_i + N_2(\vec{x}, \mathbf{U}_h, \nabla\mathbf{U}_h) \cdot \nabla\boldsymbol{\Phi}_i \mathrm{dV} + \oint_\Gamma \dots \mathrm{dS}. \tag{40}$$

The edge integral, which is left out in Equation (40), can be written in analog fashion as the volume integral, that is, as a sum over four nonlinear functions, multiplied by $\boldsymbol{\Phi}_i^+$, $\boldsymbol{\Phi}_i^-$, $\nabla\boldsymbol{\Phi}_i^+$, and $\nabla\boldsymbol{\Phi}_i^-$, respectively. These functions themselves may depend on $\vec{x}$, $\mathbf{U}_h^+$, $\mathbf{U}_h^-$, $\nabla\mathbf{U}_h^+$, and $\nabla\mathbf{U}_h^-$. For sake of compactness, this part is skipped. Realizing that $\frac{\partial \mathbf{U}_h}{\partial \mathbf{U}_j} = \boldsymbol{\Phi}_j$ and by application of the chain rule, one derives

$$\partial\mathcal{A}_{ij}(\mathbf{U}) = \int_{\Omega_h} (\partial_{\mathbf{U}_h} N_1(\vec{x}, \mathbf{U}_h, \nabla\mathbf{U}_h)\boldsymbol{\Phi}_j + \partial_{\nabla\mathbf{U}_h} N_1(\vec{x}, \mathbf{U}_h, \nabla\mathbf{U}_h)\nabla\boldsymbol{\Phi}_j) \cdot \boldsymbol{\Phi}_i + \dots \mathrm{dV} + \oint_\Gamma \dots \mathrm{dS}. \tag{41}$$

All skipped terms in Equation (41) can be derived in the same way as the contributions for N_1. In the *BoSSS* code, derivatives $\partial_{\mathbf{U}_h} N_1(\dots)$ and $\partial_{\nabla\mathbf{U}_h} N_1(\dots)$ are approximated by a finite difference, using a pertubation by $\sqrt{\texttt{eps}}$ in the respective argument, where $\texttt{eps} = 2.22044604925031 \cdot 10^{-16}$ is the floating point accuracy for double precision.

This notation allows to describe the Newton-Dogleg method which has been sufficient to solve all Newtonian setups presented in this work. However, with increasing Weissenberg number, finding a convergent solution without an initial guess becomes difficult: the Newton-Dogleg method may not diverge, but it stalls at a high residual. Therefore, for such cases we use a homotopy method, starting with a Newtonian solution and slowly increasing Weissenberg number within the time step for steady calculations, cf. Section 3.4.

3.3 | Newton-Dogleg method

Given is a linearization of Equation (38) around $\mathbf{U}_n$,

$$\mathcal{A}(\mathbf{U}_n) + \partial\mathcal{A}(\mathbf{U}_n)\underbrace{(\mathbf{U}_{n+1} - \mathbf{U}_n)}_{=:\mathbf{s}'_n} = 0. \tag{42}$$

By repeatedly solving this system, one obtains a standard Newton scheme for Equation (38), yielding a sequence of approximate solutions $\mathbf{U}_0, \mathbf{U}_1, \mathbf{U}_2, \ldots$ obtained from an initial guess $\mathbf{U}_0$ through the iteration scheme $\mathbf{U}_{n+1} = \mathbf{U}_n + \mathbf{s}'_n$. In the (classical, un-damped) Newton method, the correction step $\mathbf{s}n'$ is set to be the Newton-step, that is, $\mathbf{s}n' = \mathbf{s}_n$ with

$$\mathbf{s}_n := -\partial\mathcal{A}(\mathbf{U}_n)^{-1}\mathcal{A}(\mathbf{U}_n). \tag{43}$$

To compute $\mathbf{s}_n$, we use the sparse direct solver PARDISO, originally developed by Schenk et al.,[36-38] from the "Intel(R) Parallel Studio XE 2018 Update 3 Cluster Edition for Windows" library collection to solve the linear system.

Unfortunately, convergence of the Newton method for any starting value $\mathbf{U}_0$ is not guaranteed. In order to increase robustness when the distance between $\mathbf{U}_0$ and the exact solution $\mathbf{U}$ is large, we employ a globalization approach, presented by Pawlowski et al.,[34,35] known as the Dogleg-method, respectively, Newton-Dogleg method. Here, we intend to give only the central ideas of method and refer to the original works for further details.

Obviously, the exact solution of Equation (38) is also the minimum of the functional

$$f(\mathbf{U}) := \frac{1}{2}\left\|\mathcal{A}(\mathbf{U})\right\|_2^2. \tag{44}$$

One observes that $\nabla f(\mathbf{U}) = \partial\mathcal{A}(\mathbf{U})^T\mathcal{A}(\mathbf{U})$. For $\mathbf{U}_n$, the approximate Cauchy point, with respect to the 2-norm, is defined as the minimizer $\mathbf{g}_n$ of $\left\|\mathcal{A}(\mathbf{U}_n) + \partial\mathcal{A}(\mathbf{U}_n)\mathbf{g}_n\right\|_2$ in the direction of steepest decent, that is, $\mathbf{g}_n = \lambda\nabla f(\mathbf{U}_n)$, $\lambda \in \mathbb{R}$. Substituting $\mathbf{w} := -\partial\mathcal{A}(\mathbf{U}_n)\nabla f(\mathbf{U}_n)$, $\mathbf{g}_n$ is given as

$$\mathbf{g}_n = \frac{\mathcal{A}(\mathbf{U}_n)\cdot\mathbf{w}}{\mathbf{w}\cdot\mathbf{w}}\nabla f(\mathbf{U}_n). \tag{45}$$

For the Newton-Dogleg method, the correction step $\mathbf{s}'_n$ is chosen along the so-called Dogleg curve, which is the piece-wise linear curve from the origin to $\mathbf{g}_n$ and further to $\mathbf{s}_n$. The selection of $\mathbf{s}'_n$ on this curve is determined by the trust-region diameter $\delta > 0$:

- If $\|\mathbf{s}_n\|_2 \le \delta$, $\mathbf{s}'_n = \mathbf{s}_n$.
- If $\|\mathbf{g}_n\|_2 \le \delta$ and $\|\mathbf{s}_n\|_2 > \delta$, $\mathbf{s}'_n$ is chosen on the linear interpolation from $\mathbf{g}_n$ to $\mathbf{s}_n$ so that $\|\mathbf{s}'_n\|_2 = \delta$: For the Ansatz $\mathbf{s}'_n = \tau\mathbf{s}_n + (1-\tau)\mathbf{g}_n$, the interpolation factor τ is given as $\tau = (a^2 - c + \sqrt{(A^2 + B^2 - 2c)\delta^2 - a^2b^2 + c^2})/(a^2 + b^2 - 2c)$ with $a = \|\mathbf{g}_n\|_2$, $b = \|\mathbf{s}_n\|_2$ and $c = \mathbf{g}_n \cdot \mathbf{s}_n$.
- If $\|\mathbf{g}_n\|_2 > \delta$, $\mathbf{g}_n = (\delta/\|\mathbf{g}_n\|_2)\mathbf{g}_n$.

The choice, respectively, the adaptation of the trust region diameter δ throughout the Newton-Dogleg procedure follows a sophisticated heuristic, mainly based on comparing the actual residual reduction $\text{ared}_n := \|\mathcal{A}(\mathbf{U}_n)\|_2 - \|\mathcal{A}(\mathbf{U}_n + \mathbf{s}'_n)\|_2$ with the predicted residual reduction $\text{pred}_n := \|\mathcal{A}(\mathbf{U}_n)\|_2 - \|\mathcal{A}(\mathbf{U}_n) + \partial\mathcal{A}(\mathbf{U}_n)\mathbf{s}'_n\|_2$; for the direct solver used in this work pred_n simplifies to $\text{pred}_n := \|\mathcal{A}(\mathbf{U}_n)\|_2$. We replicate the algorithm here, for the sake of completeness:

1. Set $n = 0$, $\delta_n = \min(10^{10}, \max(2 \cdot 10^{-6}, \|\mathbf{s}_0\|_2))$.
2. Compute the Newton step $\mathbf{s}_n$ and the Cauchy point $\mathbf{g}_n$ and find $\mathbf{s}'_n$ on the Dogleg curve w.r.t. the recent δ_n.
3. While $\text{ared}_n \le \text{pred}_n$ do: Update trust region diameter $\delta_n \leftarrow 0.5\ \delta_n$ and re-compute $\mathbf{s}'_n$. If $\delta_n < 10^{-6}$ terminate abnormally and mark the computation as failed.
4. If the convergence criterion (see below) is fulfilled, terminate and mark the computation as success.

5. Perform a final update of the trust region: Set

$$\delta_{n+1} = \begin{cases} \max(10^{-6}, \|\mathbf{s}_n\|_2) & \text{if } \text{ared}_n/\text{pred}_n < 0.1 \text{ and } \|\mathbf{s}_n\|_2 \delta_n \\ \max(10^{-6}, 0.25 \cdot \delta_n) & \text{else, if } \text{ared}_n/\text{pred}_n < 0.1 \\ \min(10^{10}, 4 \cdot \delta_n) & \text{else, if } \text{ared}_n/\text{pred}_n > 0.75 \\ \delta_n & \text{otherwise} \end{cases}$$

Set $\mathbf{U}_{n+1} = \mathbf{U}_n + \mathbf{s}'_n$, update $n \leftarrow n+1$ and return to step (2).

All constants used in the algorithm above have been taken from the work of Pawlowski et al. For a detailed description of the underlying ideas, we also refer to these works, which in turn are based on algorithms from Dennis and Schnabel's textbook.

3.3.1 | Termination criterion

A simple approach to determine that the Newton-Dogleg loop can be terminated would be to check whether the residual norm has fallen below a certain threshold, that is, $\|\mathcal{A}(U_n)\| \leq \text{tol}$. A universal choice for tol is indeed difficult, especially for investigations of convergence properties (cf. Section 4.1). If chosen to low, the algorithm may never terminate, since numerical round-of errors are dominating. If chosen to high, the error of the premature termination may dominate the spatial discretization and one cannot take the full advantage from the high-order method. Therefore the goal is to continue the Newton-Dogleg method until the lowest possible limit given by the floating point accuracy is reached. To identify this in a robust way, we first define the residual-norm skyline as

$$\text{sr}_n := \min_{j \leq n} \|\mathcal{A}(\mathbf{U}_j)\| \tag{46}$$

and, for $n \geq 2$, the averaged reduction factor

$$\text{arf}_n := \frac{1}{2} \left(\frac{\text{sr}_{n-2}}{\max\{\text{sr}_{n-1}, 10^{-100}\}} + \frac{\text{sr}_{n-1}}{\max\{\text{sr}_n, 10^{-100}\}} \right). \tag{47}$$

The Newton-Dogleg method is terminated if

$$n \geq 2 \text{ and } \text{sr}_n \leq 10^{-5} + 10^{-5} \|\mathbf{U}_n\|_2 \text{ and } \text{arf}_n < 1.5. \tag{48}$$

At least for the computations in this work, this choice guarantees that the nonlinear system is solved as accurately as possible within the given floating-point accuracy. Thus, the numerical error is dominated by the spatial or temporal discretization and not by the termination criterion of the Newton-Dogleg method. The skyline-approach ensures robustness against oscillations close to the lower limit.

3.4 | Homotopy method

Although the Newton-Dogleg method works pretty well for a variety of cases, it failed to converge (within a maximum of 100 iterations) for several cases in the convergence study presented in Section 4.1. In particular, the Newton-Dogleg method failed for cases with $\text{Wi} = 0.2$ and polynomial degree $k \geq 3$ on all meshes, as well as for $k = 2$ on meshes 2 and 3, cf. Tables 1 and 2. In such cases, we increased the Weissenberg number by the following strategy, which is loosely based on ideas from the textbook of Deuflhard,[39] chapter 5.

Let

$$\mathcal{A}_{\text{wi}^*}(\mathbf{U}) = 0, \tag{49}$$

mesh	NX1	NX2	NX3	NX4	NY1	NY1b	NY2
mesh_0	20	4	6	20	4	6	4
mesh_1	30	8	12	30	8	12	8
mesh_2	45	16	24	45	16	24	16
mesh_3	68	32	48	68	32	48	32
factor	1.5	2	2	1.5	2	2	2

TABLE 1 Amount of nodes of the computational meshes in different regions for the convergence study

mesh	no of cells	$k=1$	$k=2$	$k=3$	$k=4$
mesh_0	320	5120	10,560	17,920	27,200
mesh_1	1208	19,328	39,864	67,648	102,680
mesh_2	4252	68,032	140,316	238,112	361,420
mesh_3	14,712	235,392	485,496	823,872	1,250,520

TABLE 2 Number of cells and of degrees of freedom for different mesh sizes and polynomial degrees with k for the velocity and stresses and $k-1$ for the pressure

be the discretized system for a certain intermediate Weissenberg number wi^*, between 0 and the "target" Weissenberg number Wi, that is, $0 \le \mathrm{wi}^* \le \mathrm{Wi}$. Furthermore, let $\mathbf{U}_{\mathrm{wi},\epsilon}$ be an approximate solution to the problem (49) at Weissenberg number $\mathrm{wi}^* = \mathrm{wi}$, up to a tolerance ϵ, that is,

$$\left\| \mathcal{A}_{\mathrm{wi}}(\mathbf{U}_{\mathrm{wi},\epsilon}) \right\|_2 \le \epsilon. \tag{50}$$

(For the sake of the algorithm which follows below, we distinct between the intermediate Weissenberg number wi for which we assume to already have found an acceptable solution and the next Weissenberg number wi^*, that we are currently trying to find a solution for.) For any $\mathrm{wi}^* < \mathrm{Wi}$, we set $\epsilon = 10^{-5} \left\| \mathcal{A}_{\mathrm{wi}*}(\mathbf{U}_{\mathrm{wi},\epsilon}) \right\|_2$, that is, we aim for a residual norm reduction of at least 5 magnitudes with respect to the initial residual norm. If $\mathrm{wi}^* = \mathrm{Wi}$, the termination criterion presented in Section 3.3 is applied.

An approximate solution for the target Weissenberg number is found by the following recipe:

1. Set $\mathrm{wi} = 0$, that is, start by obtaining an (approximate) solution $\mathbf{U}_{0,\epsilon}$ to the Newtonian problem.
2. Search for a an increased Weissenberg number wi^*: Find the minimal $i \ge 0$ so that for $\mathrm{wi}^* = \frac{1}{2^i}(\mathrm{Wi} - \mathrm{wi}) + \mathrm{wi}$ one has $\left\| \mathcal{A}_{\mathrm{wi}^*}(\mathbf{U}_{\mathrm{wi},\epsilon}) \right\|_2 \le \delta_{\max} \left\| \mathcal{A}_{\mathrm{wi}}(\mathbf{U}_{\mathrm{wi},\epsilon}) \right\|_2$ Here, $\zeta_{\max}$ is the maximal allowed increase of the residual for an increased Weissenberg number wi^*; $\zeta_{\max}$ is adapted in the following steps, as an initial guess we used $\zeta_{\max} = 10^6$.
3. Use the Newton-Dogleg method to compute an approximate solution to the problem (49), for Weissenberg number wi^*, using the solution $\mathbf{U}_{\mathrm{wi},\epsilon}$ as an initial guess.
   - If the Newton-Dogleg method did not converge successfully within 10 steps, the Weissenberg number increase from wi to wi^* was probably to large. Set $\zeta_{\max} \leftarrow 0.2 \cdot \zeta_{\max}$ and go to step 2.
   - If the Newton-Dogleg method reached its convergence criterion and if the target Weissenberg number is reached, that is, $\mathrm{wi}^* = \mathrm{Wi}$, the algorithm has successfully found an approximate solution for $\mathcal{A}_{\mathrm{Wi}}(\mathbf{U}) = 0$ and can terminate.
   - Otherwise, if the Newton-Dogleg method converged successfully, but is below the target Weissenberg number Accept the solution and set $\mathrm{wi} \leftarrow \mathrm{wi}^*$. If the Newton-Dogleg method took less than 3 iterations to reach the convergence criterion, set $\zeta_{\max} \leftarrow 8 \cdot \zeta_{\max}$. Return to step 2.

An exemplary run of the solver is shown in Figure 1.

4 | NUMERICAL RESULTS

We consider the confined cylinder benchmark problem. It is a two dimensional numerical simulation of viscoelastic flow around a cylinder (radius $R = 1$) immersed in a narrow channel (height $H = 4$) with a blocking ratio of $\frac{R}{H} = \frac{1}{2}$ (Figure 2).

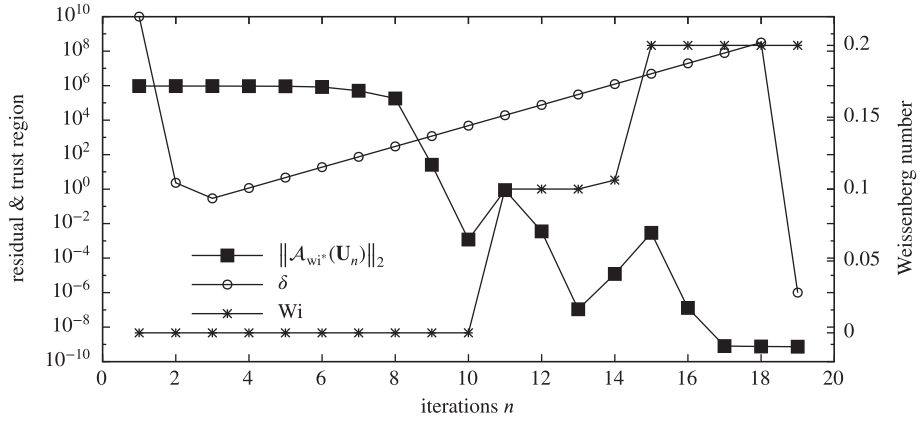


FIGURE 1 Convergence behavior of the homotopy method, for degree $k = 4$ on mesh_3. One observes that the residual $\|\mathcal{A}_{\mathrm{wi}^*}(\mathbf{U}_n)\|_2$ is initially high, and the trust region δ is reduced below 1.0, which forces the Newton-Dogleg method to pick $\mathbf{s}'_n$ close to the origin of the Dogleg curve and later between the Cauchy point and the Newton step. Slowly, the convergent region of the Newton method is approached, while δ is carefully increased. From iteration 8 on, the solver performs full Newton steps, that is, $\mathbf{s}'_n = \mathbf{s}_n$. The Weissenberg number is increased for iterations 11, 14, and 15, causing the residual to grow. From iteration 17 to 19, the lower limit imposed by floating point accuracy is reached

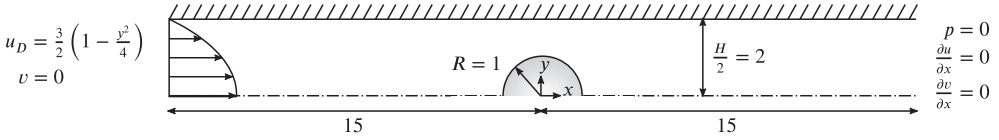


FIGURE 2 Computational domain with boundary conditions for velocities

The inflow Dirichlet boundary condition for the velocity is:

$$u = \frac{3}{2}\left(1 - \frac{y^2}{4}\right), \quad v = 0, \tag{51}$$

such that the mean velocity $\langle u \rangle$ is 1 at the inflow. We have no Dirichlet values for the stresses since in all boundary fluxes we only use the inner value $\boldsymbol{\tau}^-$ resulting from the velocity field. Since we expect the flow to be symmetric for the steady-state case, we consider only half of the channel with a free-slip boundary condition at the centre-line in order to save computational time. At the walls of the channel and on the cylinder surface, we have a no-slip boundary condition with an impermeable wall. The outflow is a pressure outlet, with the pressure and the velocity gradients set to zero.

We use a body-fitted curved elements grid with polynomial order five, which is non-equidistant with gradients toward the cylinder (Figure 3). In case of curved elements grids, the polynomial order of the polygons in the grid needs to be higher than the computational polynomial order.[40] For the convergence study, we have different refinement levels starting from 16 elements dividing the channel height at the inlet up to 128 elements for the finest grid. The amount of nodes of the coarsest grid is doubled for each refinement level. Only for the parts in x-direction of the channel we choose a refinement factor of 1.5. An overview over the different meshes can be seen in Table 1. The resulting degrees of freedom (DOF) are listed in Table 2. Since our system is fully coupled we have, especially for finer meshes and high polynomial orders, increasing DOFs which causes memory issues at computation. This is particularly the case for the direct solvers when decomposing the comparably dense matrix of the non-saddle-point problem. To solve this in future, iterative linear solvers for high-performance computing are currently under development within the *BoSSS* framework.

The dimensionless parameters are defined as follows:

$$\mathrm{Re} = \frac{\rho \langle u \rangle r}{\eta_0}, \quad \mathrm{Wi} = \lambda_1 \frac{\langle u \rangle}{r}. \tag{52}$$

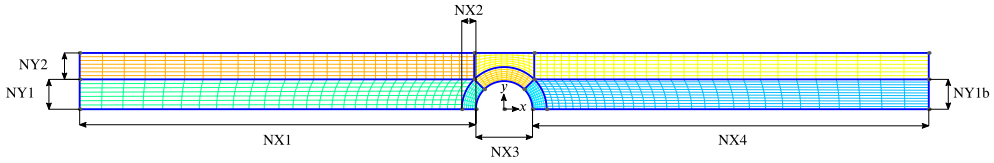


FIGURE 3 Mesh mesh_1 with location of number of nodes as referred to in Table 1 [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 3 Polynomial degree of the discretizaion and experimental order of convergence (EOC) for both Weissenberg numbers and different dependent variables

	EOC_u		EOC_p		EOC_τ	
k	Wi = 0	Wi = 0.2	Wi = 0	Wi = 0.2	Wi = 0	Wi = 0.2
1	2.17	2.20	1.76	1.76	1.11	1.28
2	3.46	2.58	2.03	2.27	2.02	1.87
3	4.12	3.64	2.76	2.79	2.92	2.84
4	5.27	4.92	4.08	4.17	4.03	4.44

They are chosen, if not stated differently, as follows: $\beta = 0.59$, $\mathrm{Re} = 0, 0.01, 0.1, 1$, $\mathrm{Wi} = 0, \ldots, 1$. Apart from the convergence study (Section 4.1), all calculations were made with the comparably coarse mesh_1 and a polynomial degree for the velocity and stresses of $k = 4$. For all calculations apart from Section 4.4, we have $\mathrm{Re} = 0$. For all steady calculations, we use an implicit Euler scheme with one pseudo time step of $\Delta t = 10^6$; for the unsteady calculations in Section 4.3, we use the BDF2 scheme (Equation 19) with a time step size of $\Delta t = 10^{-2}$.

4.1 | Convergence study

We examine our solver in the case of the confined cylinder problem for convergence against the solution of the finest grid for different polynomial degrees in the approximation spaces ($k = 1 \ldots 4$) for $\mathrm{Re} = 0$. For $\mathrm{Wi} = 0$ (i.e., a Newtonian fluid flow) and $\mathrm{Wi} = 0.2$ the results are shown in Figure 4 and the convergence rates are listed in Table 3. For the Newtonian case, all convergence rates for the L^2-norm are equal or greater than $k + 1$ for the velocity and pressure or k for the stress, as is expected for the LDG scheme used for discretization.[9] For the viscoelastic case, there are small deviations in the L^2-error for the finer grids and higher polynomial degrees such that the convergence rates of $k + 1$ and k cannot be satisfied completely for all dependent variables.

4.2 | Results for steady flow simulation

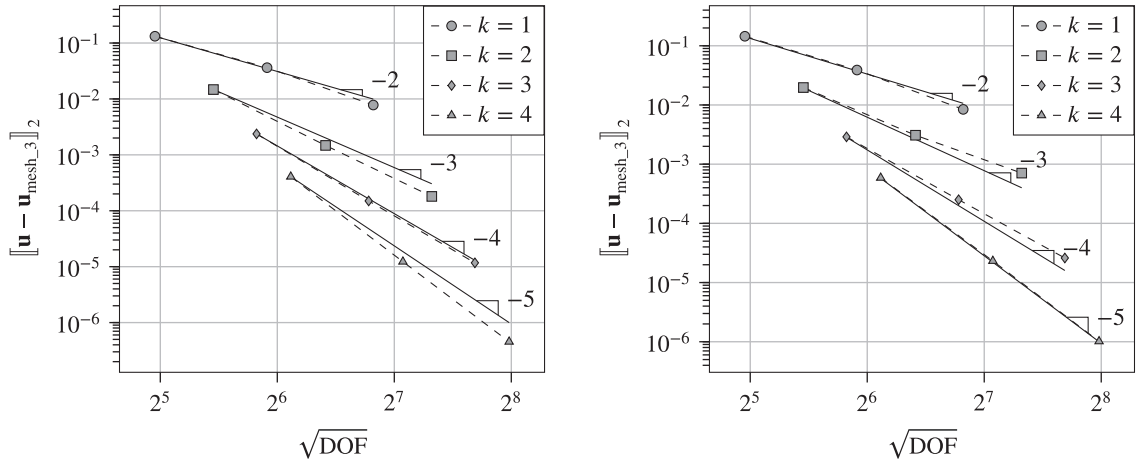
4.2.1 | Cylinder drag

As a measure to compare the accuracy of the method with other methods from literature, the dimensionless drag force of the cylinder in the flow for different Weissenberg numbers is used:

$$F = \int_\Gamma \boldsymbol{T} \cdot \mathbf{n} \, \mathrm{dS}, \tag{53}$$

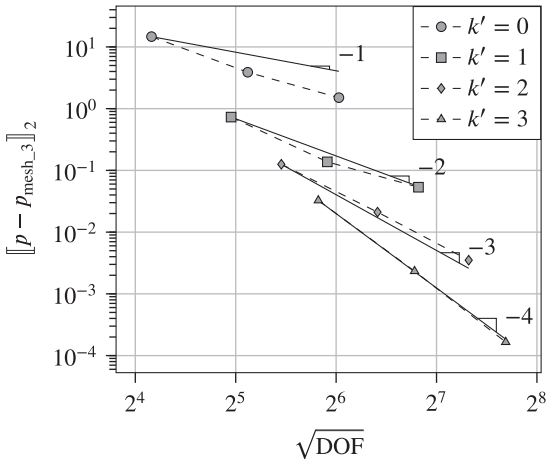
where F is the dimensionless drag force calculated as the surface integral of the total stress tensor

$$\boldsymbol{T} = -p\boldsymbol{I} + \frac{\beta}{\mathrm{Re}} \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) + \frac{1}{\mathrm{Re}} \boldsymbol{\tau}. \tag{54}$$

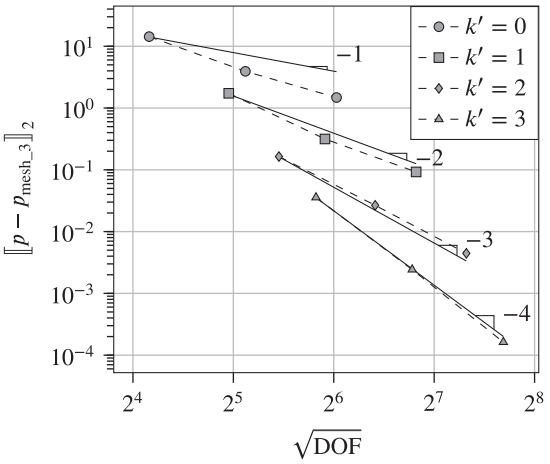


(A) Velocity, Wi = 0.

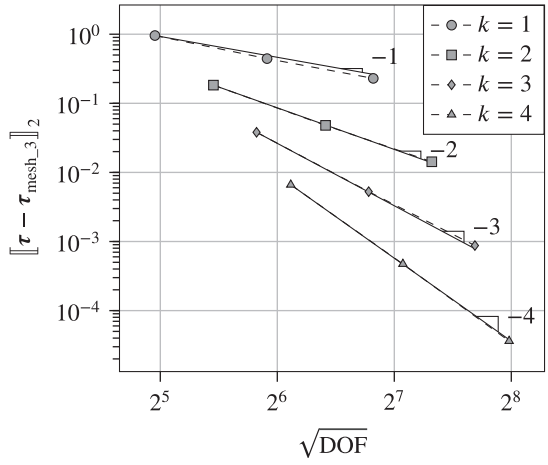
(d) Velocity, Wi = 0.2.



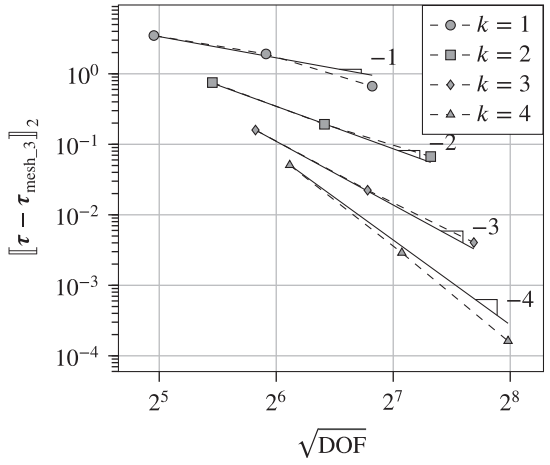
(B) Pressure, Wi = 0.



(E) Pressure,Wi = 0.2.



(C) Stresses, Wi = 0.



(F) Stresses, Wi = 0.2.

FIGURE 4 Convergence study for Wi = 0 (left) and Wi = 0.2 (right) in the L^2-norm compared to the degrees of freedom (DOF) of the finest mesh. The solid lines show the expected convergence rates with $k+1$ for velocity and pressure and k for the stresses

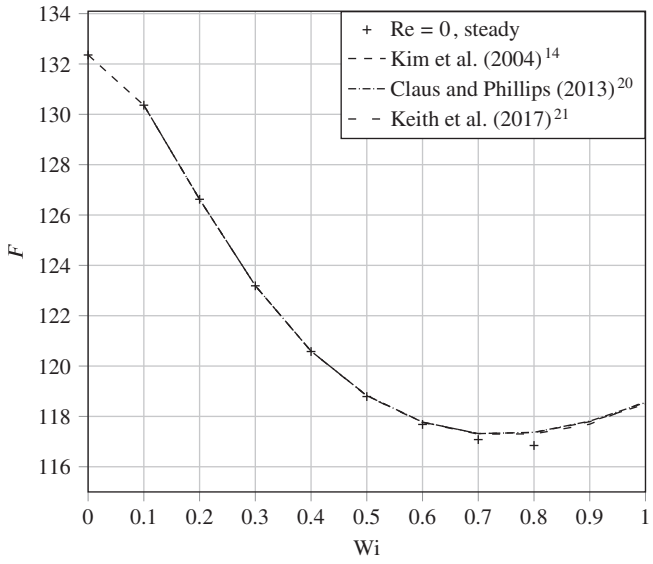


FIGURE 5 Dimensionless drag force of the confined cylinder for different Weissenberg numbers for steady calculations (+) compared with unsteady calculations from exemplary literature.[14,20,21] For $Wi \leq 0.6$, we have good agreement. Afterward, unsteady effects cause errors in the unstable steady state solution as expected

Wi	F	(a)	(b)	(c)
0	132.357	132.36	—	—
0.1	130.363	130.36	130.364	130.3618
0.2	126.625	126.62	126.626	126.6241
0.3	123.188	123.19	123.192	123.1897
0.4	120.577	120.59	120.593	120.5885
0.5	118.789	118.83	118.826	118.8132
0.6	117.680	117.77	117.776	117.7581
0.7	117.079	117.32	117.316	117.2951
0.8	116.844	117.36	117.368	117.3057
0.9	—	117.79	117.812	117.6907
1.0	—	118.49	118.550	118.5970

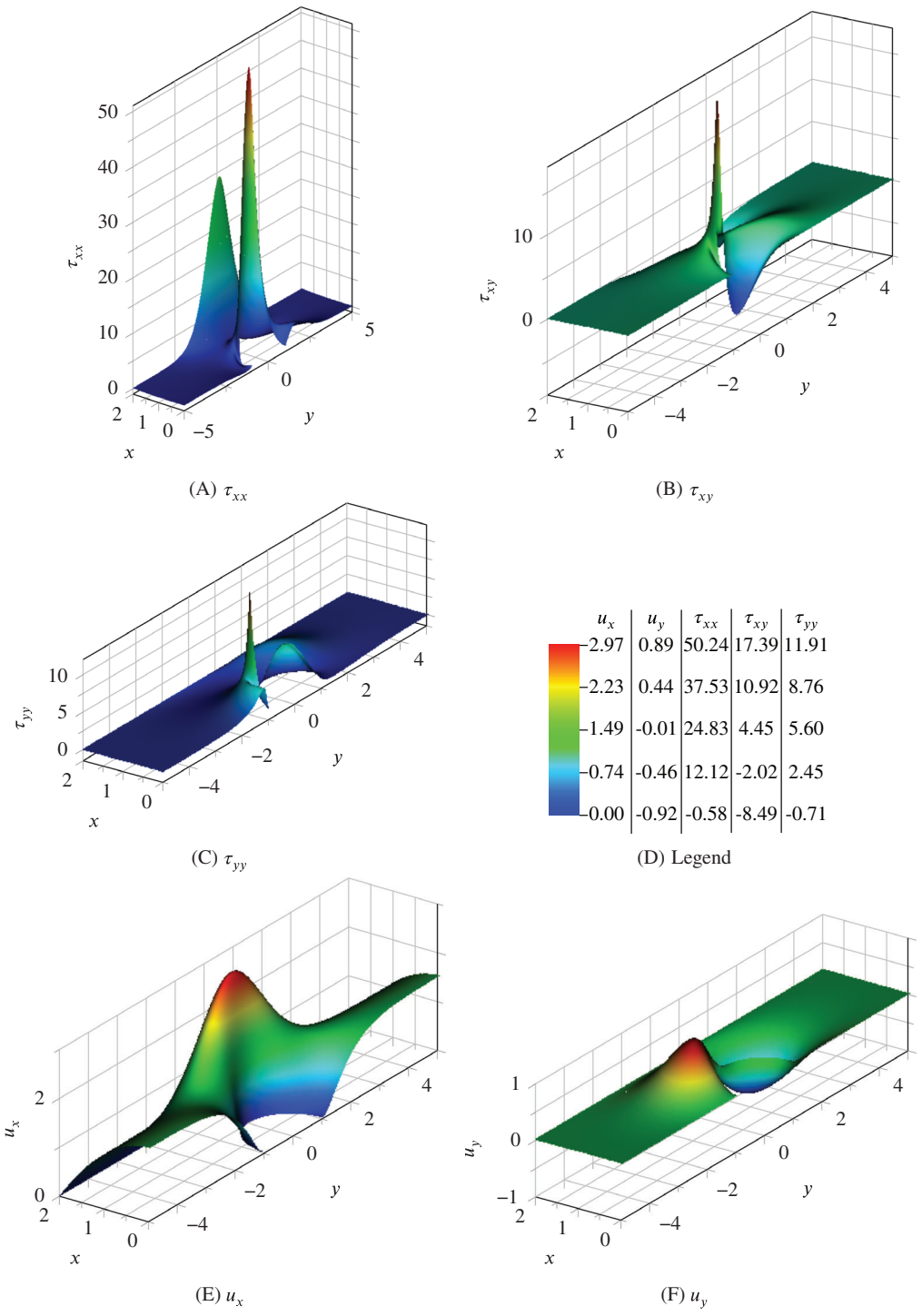
TABLE 4 Dimensionless drag force of the confined cylinder for different Weissenberg numbers for steady calculations compared with unsteady calculations from exemplary literature (a),[14] (b),[20] and (c)[21]

The resulting drag forces for different Weissenberg numbers are plotted in Figure 5, listed in Table 4, and compared to selected literature.[14,20,21] For smaller Weissenberg numbers ($Wi \leq 0.6$), we are in good agreement with the values obtained in other studies. Our drag force values are lower for the higher Weissenberg numbers due to lower absolute values for the normal stresses along the cylinder surface where the drag is calculated (Figure 7). This is due to the fact that our steady simulations naturally do not capture the unsteady behavior which is observed for $Wi > 0.6$, a transitions to unsteady behavior can be observed. For the same reason, we could not conduct a steady convergent solution for $Wi > 0.8$.

4.2.2 | Flow behavior

In Figure 6, color plots for the velocity and the stresses for the whole domain are shown. The velocity u becomes larger in the narrow between the channel wall and the cylinder and we have a small vertical velocity v due to the displacement of the fluid along the cylinder. At the cusp of the cylinder and at the channel wall, we can see high stress peaks, especially in the normal stress τ_{xx}, and a second peak in the wake of the cylinder. The values of the normal stress τ_{xx} are magnitudes greater than the values of τ_{yy}.

The flow behavior of the stress component τ_{xx} along the symmetry line and on the cylinder surface is further investigated. As can be seen in Figure 7, our results are in very good agreement with the work of Claus and Phillips[20] concerning



u_x	u_y	τ_{xx}	τ_{xy}	τ_{yy}
2.97	0.89	50.24	17.39	11.91
2.23	0.44	37.53	10.92	8.76
1.49	-0.01	24.83	4.45	5.60
0.74	-0.46	12.12	-2.02	2.45
0.00	-0.92	-0.58	-8.49	-0.71

FIGURE 6 Color plots of the stress and velocity profiles over the domain $-5 \leq y \leq 5$ for $\mathrm{Wi} = 0.3$. The acceleration of the fluid and the high stress peaks in the narrow at the cusp of the cylinder as well as the stress peak in the wake of the cylinder are clearly visible [Colour figure can be viewed at wileyonlinelibrary.com]

the wake of the cylinder. Along the cylinder surface the resulting normal stress τ_{xx} is for higher Weissenberg numbers not as high as in the work of Claus and Phillips,[20] which can be due to the fact that we are comparing steady solutions with steady-state solutions from an unsteady computation from literature. The stress component τ_{xx} increases along the cylinder surface up to the cusp of the cylinder and then decreases to zero at the rear stagnation point. In the following, a second steep gradient can be seen, forming a tail in the wake of the cylinder with a peak directly behind the rear stagnation point. The two peaks grow for increasing Weissenberg numbers.

Dou and Phan-Thien[7] developed an inflection point theory for a detaching boundary layer at the cusp of the cylinder for increasing $\mathrm{Wi} \geq 0.7$. Claus and Phillips[20] found many indications supporting that theory. We investigated our results conducted from steady calculations for these indications and are in good agreement.

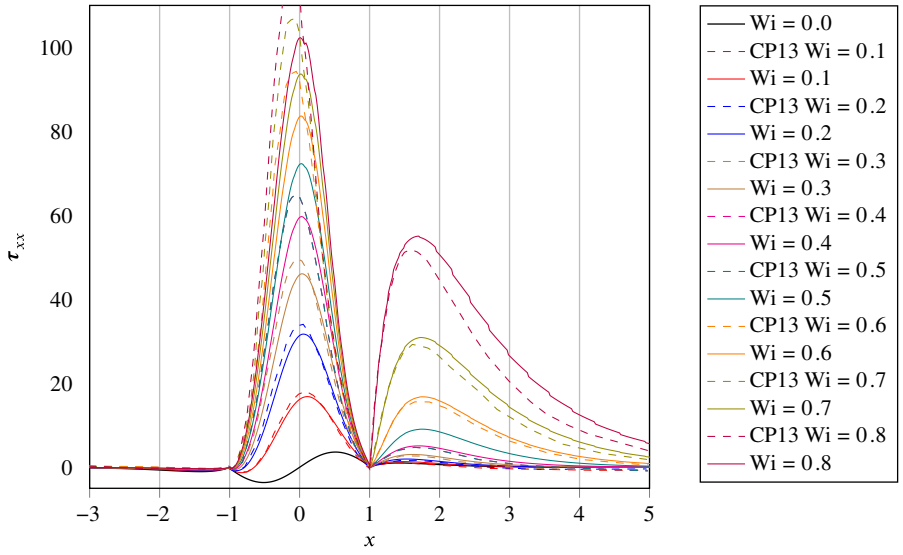


FIGURE 7 Normal stress τ_{xx} at the symmetry line and on the cylinder surface in the interval $[-1 \dots 1]$. Obviously, the steady-state results (solid lines) deviate significantly from the transient calculations of Claus and Phillips[20] (dashed, marked as CP13), since steady simulations naturally do not capture the unsteady behavior which is observed for $\mathrm{Wi} > 0.6$, where a transitions to unsteady behavior can be observed. For larger $\mathrm{Wi} > 0.8$, no steady solution could be accomplished [Colour figure can be viewed at wileyonlinelibrary.com]

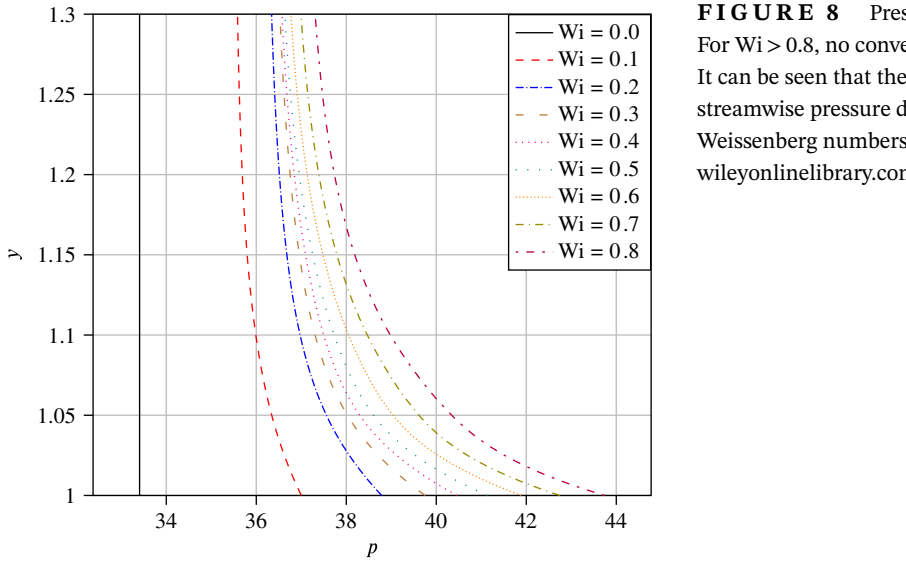


FIGURE 8 Pressure distribution on the cusp of the cylinder. For $\mathrm{Wi} > 0.8$, no convergent steady solution could be accomplished. It can be seen that the angle between the cross-stream and the streamwise pressure derivative becomes larger for higher Weissenberg numbers [Colour figure can be viewed at wileyonlinelibrary.com]

Dou and Phan-Thien explained the appearance of an inflection point in the velocity profile in the boundary layer at the cusp of the cylinder with a non-constant and increasing pressure in the boundary layer for increasing Weissenberg number. For $\mathrm{Wi} > 0.6$, the resulting pressure gradient from within the shear layer to outside ($\frac{\partial p}{\partial y}$) is large and causes a maximum in the velocity gradient with a resulting inflection point in the velocity.[7] This behavior can be measured by the ratio between the streamwise and normal energy gradients, which is proportional to the ratio of the corresponding pressure gradients for negligible kinetic energy. Hence, the angle between cross-stream and streamwise pressure is a measure for the probability for an inflection point in the velocity.[20] In Figure 8, the pressure distribution at the cusp of the cylinder is shown for different Weissenberg numbers. The angle described above increases for increasing Weissenberg numbers, which is in good agreement with Claus and Phillips.[20]

Further indication for the appearance of a velocity inflection is given by the behavior of the velocity itself at the cusp of the cylinder. The distribution of the velocity u_y increases for increasing Weissenberg number close to the cusp of the cylinder and decreases in the upper part close to the channel wall (Figure 9). At $y \approx 1.01$, there is a kink in the slope of the velocity when leaving the boundary layer. This kink is not at a discontinuous element edge but within a cell and was also detected by Dou and Phan-Thien[7] and Claus and Phillips,[20] but at $y \approx 1.02$, so that it is possibly no numerical artifact. This change in slope increases for increasing Weissenberg numbers. Dou and Phan-Thien assume that it causes fluid elements to leave the boundary layer and therefore, disturb the flow. This kind of disturbance can be transported downstream and amplified, leading to a transition to unsteady behavior behind the cylinder.

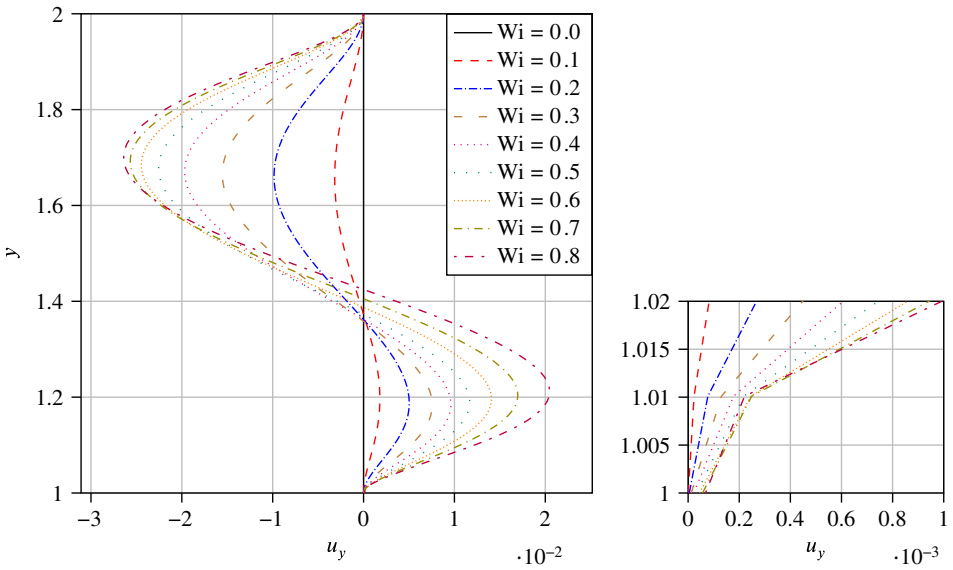


FIGURE 9 Distribution of the velocity u_y on the cusp of the cylinder. For larger Wi > 0.8, no convergent steady solution could be accomplished. It can be seen in the detail plot that there is a kink in the distribution close to the cylinder wall [Colour figure can be viewed at wileyonlinelibrary.com]

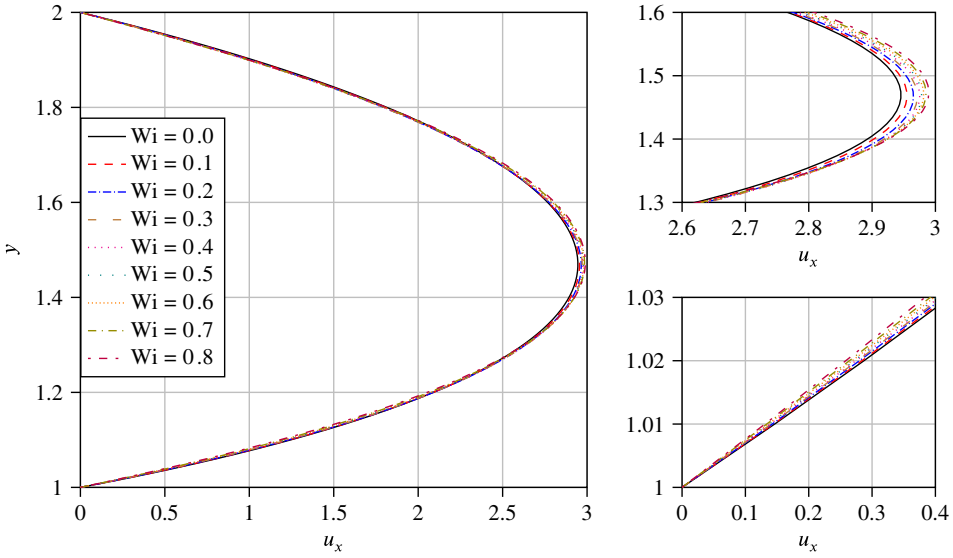


FIGURE 10 Distribution of the velocity u_x on the cusp of the cylinder. For larger Wi > 0.8, no convergent steady solution could be accomplished. On the detail plots, it can be seen that the maximum velocity increases for increasing Wi and the velocity decreases at the cusp of the cylinder [Colour figure can be viewed at wileyonlinelibrary.com]

Considering the distribution of the velocity u_x (Figure 10), it can be seen that the maximum velocity increases with increasing Weissenberg numbers. Furthermore, the velocity is decreased close to the cylinder surface with increasing Weissenberg number.

4.3 | Results for unsteady flow simulation

For Wi ≥ 0.5, we performed unsteady simulations with a BDF2 scheme and $\Delta t = 10^{-2}$. Whereas for Wi = 0.5, a stable steady state solution can be achieved over a long time period; for higher Weissenberg numbers, the simulation stops converging at a certain time step without diverging (Figure 11). For Wi = 0.6, this can be observed at time step 1586, whereas for Wi = 1.0 this time step is already time step 864.

As can be seen in detail in Figure 11, for the lower Weissenberg numbers up to Wi = 0.7, an apparent steady-state solution is reached before the solution stops converging, but for the higher Weissenberg numbers, the drag coefficient starts to oscillate immediately.

Further investigation about the lack of convergence for unsteady calculation with higher Weissenberg numbers is needed.

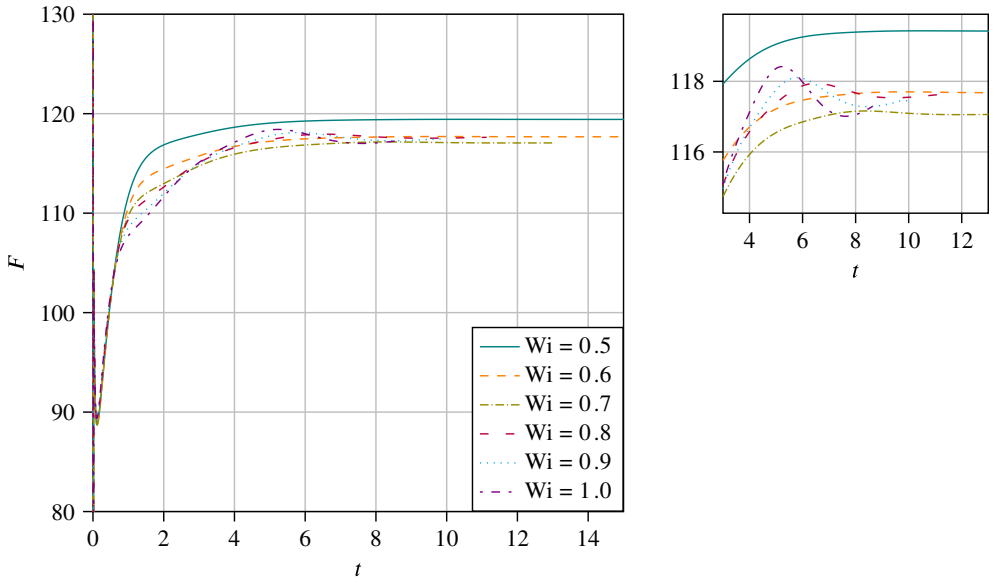


FIGURE 11 Evolution in time of the drag force of the confined cylinder for different Weissenberg numbers for unsteady calculations. For higher $\mathrm{Wi} \geq 0.6$, there is in the detail plot an oscillatory behavior which might be caused by unsteady effects due to velocity inflection in the boundary layer [Colour figure can be viewed at wileyonlinelibrary.com]

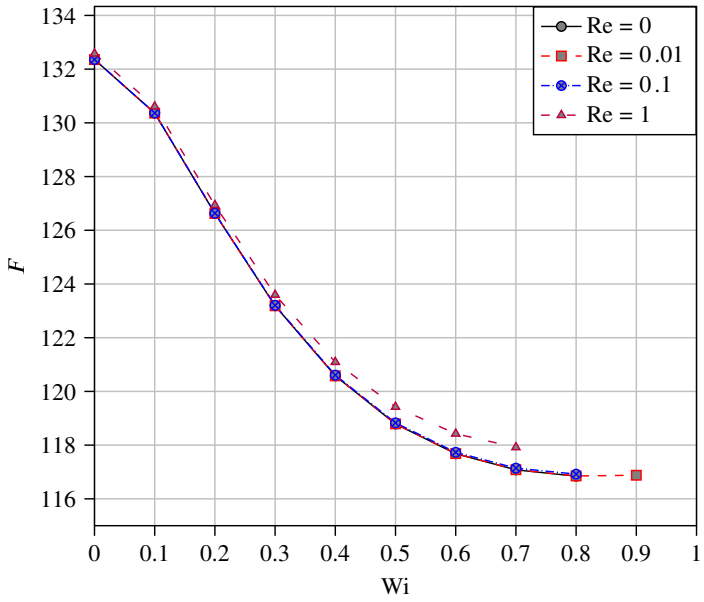


FIGURE 12 Drag coefficient depending on the Weissenberg number for different Reynolds numbers [Colour figure can be viewed at wileyonlinelibrary.com]

4.4 | Results for different Reynolds numbers

In this section, we show the behavior of the steady solution for increasing Reynolds number and, therefore, examine the influence of inertia on the flow. As we can see in Figure 12, the drag of the cylinder increases for higher Reynolds numbers and the drag reduction effect caused by the viscoelasticity is reduced. This is due to the increased velocity gradients caused by the inertia of the fluid. For lower Weissenberg numbers, our steady results are in good agreement with the literature[14,20,21] (Table 5).

All further studies were made for $\mathrm{Wi} = 0.6$. The velocity u_y increases for increasing Reynolds number close to the cusp of the cylinder and decreases with increasing Reynolds number close to the channel wall. This behavior is amplified for viscoelastic flow as showed in Figure 13. Here, the effect is shown for a Newtonian fluid as well as the Oldroyd B fluid with $\mathrm{Wi} = 0.6$.

The normal stress τ_{xx} in Figure 14 decreases only slightly with increasing Reynolds number at the cusp of the cylinder, but in the wake of the cylinder a large decrease is notable for $\mathrm{Re} = 1$.

TABLE 5 Dimensionless drag force of the confined cylinder for different Weissenberg numbers for steady calculations for different Reynolds numbers compared with unsteady calculations from exemplary literature (a)[20] and (b)[21]

Wi	$Re = 0.01$	(a)	(b)	$Re = 0.1$	(a)	(b)	$Re = 1$	(a)	(b)
0.1	130.363	130.364	130.363	130.367	130.368	130.367	130.607	130.609	130.608
0.2	126.626	126.627	126.626	126.635	126.636	126.635	126.936	126.938	126.937
0.3	123.189	123.194	123.192	123.206	123.211	123.210	123.593	123.597	123.596
0.4	120.579	120.595	120.593	120.606	120.622	120.620	121.093	121.106	121.106
0.5	118.792	118.831	118.827	118.830	118.868	118.865	119.423	119.460	119.457
0.6	117.685	117.781	117.775	117.734	117.831	117.823	118.424	118.542	118.538
0.7	117.085	117.323	117.308	117.143	117.387	117.372	117.918	118.233	118.222
0.8	116.851	117.379	117.277	116.920	117.459	117.373	—	118.455	118.397
0.9	116.878	117.827	117.559	—	117.925	117.684	—	119.096	118.874
1.0	—	118.563	118.130	—	118.697	118.224	—	120.057	120.146

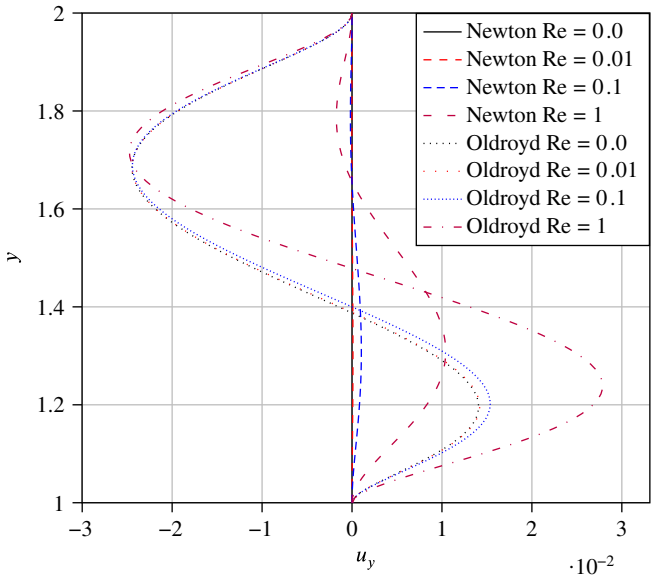


FIGURE 13 Distribution of the velocity u_y on the cusp of the cylinder for Wi = 0.6 and Newtonian flow for different Reynolds numbers [Colour figure can be viewed at wileyonlinelibrary.com]

5 | CONCLUSION

In this work, we present a DG method for solving viscoelastic Oldroyd B fluid flow. We use a LDG formulation with penalization terms in the fluxes to stabilize the system. It is used in combination with an SIP formulation for the Newtonian contribution. The nonlinear system can be solved using an Newton method.

Because of the new approach using LDG and a fully coupled system, there is no need for hyperbolic-elliptic stabilization methods like (D)EVSS. Hence, there is no need for additional equations for the velocity gradient. Further investigation about the convergence behavior for unsteady calculations with higher Weissenberg numbers is needed. One possible remedy it is the introduction of the log-confromation formulation presented by Fattal and Kupfermann.[41]

Since this work intends to present the DG method, the results are kept short and the physical phenomena are not analyzed in detail. Since a direct sparse solver is used in the Newton iteration, the fully coupled system has a high memory demand for the sparse LU-factorization of the operator matrix, and thus can only be used on high performance computers with a large memory. Furthermore, direct sparse solvers are known to scale not particularly well with more CPU cores, since the parallelization of several stages is rather difficult. The use of iterative linear solvers, which are currently under development, could alleviate the memory demands and should open the door for MPI-parallel computation across multiple compute nodes. Furthermore, an adaptive mesh refinement strategy, which is already implemented in

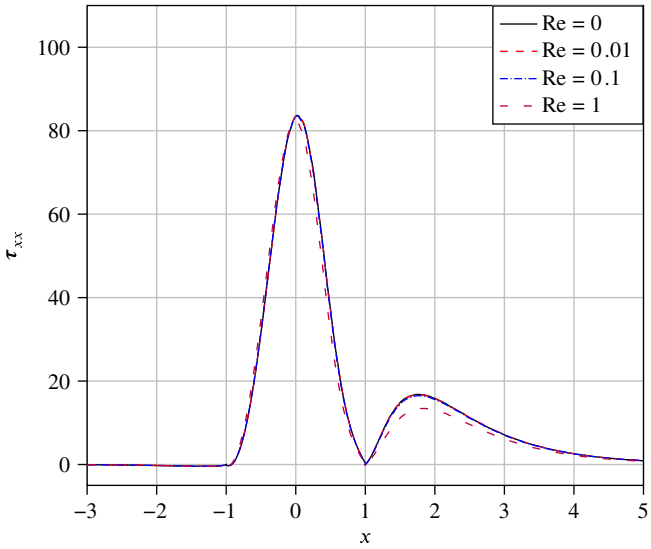


FIGURE 14 Normal stress τ_{xx} at the symmetry line and on the cylinder surface in the interval $[-1 \ldots 1]$ for Wi = 0.6 for different Reynolds numbers [Colour figure can be viewed at wileyonlinelibrary.com]

BoSSS, can be useful in combination an appropriate refinement indicator to provide a reasonable distribution of grid resolution among the computational domain, that is, refinement in regions with boundary layers or other phenomena of interest.

ACKNOWLEDGEMENTS

The work of Anne Kikker is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - Project number: 330615302 and supported by the "Excellence Initiative" of the German Federal and State Governments and the Graduate School of Computational Engineering at Technical University Darmstadt. The work of Florian Kummer has been partially supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - Project-ID 265191195 - SFB 1194.

DATA AVAILABILITY STATEMENT

Research data for this publication is available: https://doi.org/10.25534/tudatalib-355.[42]

ORCID

Anne Kikker https://orcid.org/0000-0003-4989-1759
Florian Kummer https://orcid.org/0000-0002-2827-7576
Martin Oberlack https://orcid.org/0000-0002-5849-3755

REFERENCES

1. Thurston G. Viscoelasticity of human blood. *Biophys J*. 1972;12(9):1205-1217. https://doi.org/10.1016/S0006-3495(72)86156-3.
2. Bodnar T, Sequeira A, Prosi M. On the shear-thinning and viscoelastic effects of blood flow under various flow rates. *Appl Math Comput*. 2011;217(11):5055-5067. https://doi.org/10.1016/j.amc.2010.07.054.
3. Connolly JAD, Podladchikov YY. Temperature-dependent viscoelastic compaction and compartmentalization in sedimentary basins. *Tectonophysics*. 2000;324(3):137-168. https://doi.org/10.1016/S0040-1951(00)00084-6.
4. Hu HW, Granick S. Viscoelastic dynamics of confined polymer melts. *Science*. 1994;258(5086):1339-1342. https://doi.org/10.1126/science.258.5086.1339.
5. Owens RG, Phillips TN. *Computational Rheology*. London, UK/River Edge, NJ: Imperial College Press; Distributed by World Scientific Publishing Company; 2002.
6. Joseph DD, Renardy M, Saut JC. *Hyperbolicity and Change of Type in the Flow of Viscoelastic Fluids*. Berlin/Heidelberg, Germany: Springer; 1985:25-63.
7. Dou HS, Phan-Thien N. Viscoelastic flow past a confined cylinder: instability and velocity inflection. *Chem Eng Sci*. 2007;62(15):3909-3929. https://doi.org/10.1016/j.ces.2007.03.040.
8. Fortin M, Fortin A. A new approach for the FEM simulation of viscoelastic flows. *J Non-Newtonian Fluid Mech*. 1989;32(3):295-310. https://doi.org/10.1016/0377-0257(89)85012-8.
9. Cockburn B, Kanschat G, Schötzau D, Schwab C. Local discontinuous Galerkin methods for the stokes system. *SIAM J Numer Anal*. 2002;40(1):319-343. https://doi.org/10.1137/S0036142900380121.

10. Arnold D, Brezzi F, Cockburn B, Marini L. Discontinuous Galerkin methods. *SIAM J Numer Anal*. 2002;39:1749-1779. https://doi.org/10.1137/S0036142901384162.
11. Mirzakhalili E, Nejat A. High-order solution of viscoelastic fluids using the discontinuous galerkin method. *J Fluids Eng*. 2015;137(3):031205. https://doi.org/10.1115/1.4028779.
12. Joseph DD, Saut JC. Change of type and loss of evolution in the flow of viscoelastic fluids. *J Non-Newtonian Fluid Mech*. 1986;20:117-141. https://doi.org/10.1016/0377-0257(86)80018-0.
13. Joseph DD. *Fluid Dynamics of Viscoelastic Liquids. 84 of Applied Mathematical Sciences*. New York, NY: Springer; 2013.
14. Kim JM, Kim C, Ahn KH, Lee SJ. An efficient iterative solver and high-resolution computations of the Oldroyd-B fluid flow past a confined cylinder. *J Non-Newtonian Fluid Mech*. 2004;123(2):161-173. https://doi.org/10.1016/j.jnnfm.2004.08.003.
15. Sun J, Smith MD, Armstrong RC, Brown RA. Finite element method for viscoelastic flows based on the discrete adaptive viscoelastic stress splitting and the discontinuous Galerkin method: DAVSS-G/DG. *J Non-Newtonian Fluid Mech*. 1999;86(3):281-307. https://doi.org/10.1016/S0377-0257(98)00176-1.
16. Baaijens FPT, Selen SHA, Baaijens HPW, Peters GWM, Meijer HEH. Viscoelastic flow past a confined cylinder of a low density polyethylene melt. *J Non-newtonian Fluid Mech*. 1997;68(2):173-203. https://doi.org/10.1016/S0377-0257(96)01519-4.
17. Fan Y, Yang H, Tanner RI. Stress boundary layers in the viscoelastic flow past a cylinder in a channel: limiting solutions. *Acta Mechanica Sinica*. 2005;21(4):311-321. https://doi.org/10.1007/s10409-005-0040-z.
18. Oliveira PJ, AIP M. A numerical study of steady and unsteady viscoelastic flow past bounded cylinders. *J Non-Newtonian Fluid Mech*. 2005;127(1):51-66. https://doi.org/10.1016/j.jnnfm.2005.02.003.
19. Bonito A, Burman EA. Continuous interior penalty method for viscoelastic flows. *SIAM J Sci Comput*. 2008;30(3):1156-1177. https://doi.org/10.1137/060677033.
20. Claus S, Phillips T. Viscoelastic flow around a confined cylinder using spectral/hp element methods. *J Non-Newtonian Fluid Mech*. 2013;200:131-146. https://doi.org/10.1016/j.jnnfm.2013.03.004.
21. Keith B, Knechtges P, Roberts N, Elgeti S, Behr M, Demkowicz L. An ultraweak DPG method for viscoelastic fluids. *J Non-Newtonian Fluid Mech*. 2017;247:107-122. https://doi.org/10.1016/j.jnnfm.2017.06.006.
22. Hulsen MA, Fattal R, Kupferman R. Flow of viscoelastic fluids past a cylinder at high Weissenberg number: stabilized simulations using matrix logarithms. *J Non-Newtonian Fluid Mech*. 2005;127(1):27-39. https://doi.org/10.1016/j.jnnfm.2005.01.002.
23. Coronado OM, Arora D, Behr M, Pasquali MA. Simple method for simulating general viscoelastic fluid flows with an alternate log-conformation formulation. *J Non-Newtonian Fluid Mech*. 2007;147(3):189-199. https://doi.org/10.1016/j.jnnfm.2007.08.005.
24. GH MK, Armstrong RC, Brown R. The wake instability in viscoelastic flow past confined circular cylinders. *Philos Trans Royal Soc Lond Ser A Phys Eng Sci*. 1671;344:265-304. https://doi.org/10.1098/rsta.1993.0091.
25. Persson PO, Peraire J. Sub-Cell shock capturing for discontinuous Galerkin methods. Paper presented at: Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit; 2006:112; Reno, Nevada.
26. Kummer F. Extended discontinuous Galerkin methods for two-phase flows: the spatial discretization. *Int J Numer Methods Eng*. 2017;109(2):259-289. https://doi.org/10.1002/nme.5288.
27. Babuška I. The finite element method with Lagrangian multipliers. *Numer Math*. 1973;20(3):179-192. https://doi.org/10.1007/BF01436561.
28. Brezzi F. On the existence uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Publications mathématiques et informatique de Rennes*. 1974;8(R2):129-151. https://doi.org/10.1051/m2an/197408R201291.
29. Girault V, Rivière B, Wheeler MA. Discontinuous Galerkin method with nonoverlapping domain decomposition for the stokes and Navier-Stokes problems. *Math Comput*. 2005;74(249):53-84. https://doi.org/10.1090/S0025-5718-04-01652-7.
30. Shahbazi K, Fischer PF, Ethier CRA. High-order discontinuous Galerkin method for the unsteady incompressible Navier–stokes equations. *J Comput Phys*. 2007;222(1):391-407. https://doi.org/10.1016/j.jcp.2006.07.029.
31. Shahbazi K. An explicit expression for the penalty parameter of the interior penalty method. *J Comput Phys*. 2005;205(2):401-407. https://doi.org/10.1016/j.jcp.2004.11.017.
32. Hillewaert K. Development of the Discontinuous Galerkin Method for High-Resolution, Large Scale CFD and Acoustics in Industrial Geometries; 2013.
33. Kummer F. *The BoSSS Discontinuous Galerkin Solver for Incompressible Fluid Dynamics and an Extension to Singular Equations*. Darmstadt: TU Darmstadt; 2012.
34. Pawlowski RP, Shadid JN, Simonis JP, Walker HF. Globalization techniques for Newton–Krylov methods and applications to the fully coupled solution of the Navier–stokes equations. *SIAM Rev*. 2006;48(4):700-721. https://doi.org/10.1137/S0036144504443511.
35. Pawlowski RP, Simonis JP, Walker HF, Shadid JN. Inexact Newton Dogleg methods. *SIAM J Numer Anal*. 2008;46(4):2112-2132. https://doi.org/10.1137/050632166.
36. Schenk O, Gärtner K, Fichtner W. Efficient sparse LU factorization with left-right looking strategy on shared memory multiprocessors. *BIT Numer Math*. 2000;40(1):158-176. https://doi.org/10.1023/A:1022326604210.
37. Schenk O. Two-level dynamic scheduling in PARDISO: Improved scalability on shared memory multiprocessing systems. *Parallel Comput*. 2002;28(2):187-197. https://doi.org/10.1016/S0167-8191(01)00135-1.
38. Schenk O. Solving unsymmetric sparse systems of linear equations with PARDISO. *Futur Gener Comput Syst*. 2004;20(3):475-487. https://doi.org/10.1016/j.future.2003.07.011.
39. Deuflhard P. *Parameter Dependent Systems: Continuation Methods*. Berlin/Heidelberg, Germany: Springer; 2011:233-282.

40. Bassi F, Rebay S. High-order accurate discontinuous finite element solution of the 2D Euler equations. *J Comput Phys*. 1997;138(2):251-285. https://doi.org/10.1006/jcph.1997.5454.
41. Fattal R, Kupferman R. Constitutive laws for the matrix-logarithm of the conformation tensor. *J Non-Newtonian Fluid Mech*. 2004;123(2–3):281-285. https://doi.org/10.1016/j.jnnfm.2004.08.008.
42. Kummer F, Anne K. Publication data collection: a fully coupled high-order discontinuous Galerkin solver for viscoelastic fluid flow. *Int J Numer Methods Fluids*. 2020.

How to cite this article: Kikker A, Kummer F, Oberlack M. A fully coupled high-order discontinuous Galerkin solver for viscoelastic fluid flow. *Int J Numer Meth Fluids*. 2021;93:1736–1758. https://doi.org/10.1002/fld.4950