Monolithic Newton-multigrid solution techniques for incompressible nonlinear flow models ⋆

H. Damanik [a,*], J. Hron [b], A. Ouazzi [a], S. Turek [a]

[a] *Institut für Angewante Mathematik, TU Dortmund, Germany*

[b] *Institute of Mathematics, Charles University Prague, Czech Republic*

Abstract

We present special Newton-multigrid techniques for stationary incompressible nonlinear flow models discretised by the high order LBB-stable Q_2P_1 element pair. We treat the resulting nonlinear and the corresponding linear discrete systems by a fully coupled monolithic approach to maintain high accuracy and robustness, particularly with respect to different rheological behaviour but also regarding different problem sizes and type of nonlinearity. Here, local pressure Schur complement (PSC) techniques are presented as a generalization of the classical Vanka smoother. The discussed methodology is implemented for the well-known flow around cylinder benchmark configuration [1–4] for generalised Newtonian as well as non-Newtonian flows including non-isothermal, shear/pressure dependent and viscoelastic effects.

Key words: Monolithic Newton-multigrid, Vanka smoother, FEM, non-Newtonian

1 Introduction

It is well-known that the efficiency of most incompressible CFD computations heavily relies on the linear and nonlinear solver for the resulting discrete problems,

$$\mathcal{A}\mathbf{x} = \mathbf{b}, \tag{1}$$

⋆ This research was supported by the German Research Foundation (DFG) through the collaborative research center SFB/TR TRR 30 and through the grants TU 102/11-3 (FOR493) and TU 102/21-1 and by Necas Center for Mathematical Modeling through project LC06052 financed by MSMT

* Corresponding author.

Email addresses: hdamanik@math.tu-dortmund.de (H. Damanik), hron@karlin.mff.cuni.cz (J. Hron), ouazzi@math.tu-dortmund.de (A. Ouazzi), ture@featflow.de (S. Turek).

where $\mathcal{A}$ is mostly a (huge) sparse matrix arising from finite element, finite difference or finite volume discretizations. The properties of $\mathcal{A}$ depend on the nonlinearity of the flow problems (the constitutive model) and may range from 'very easy' upto 'very hard' to solve. Thus, any iterative solver faces the challenging task of how to deal with complex fluid flow models. If large computer memory is not a problem, then a direct solver for the sparse systems is in favor, which can be Gauss elimination and its variants. A good candidate is UMFPACK (Unsymmetric MultiFrontal Package) which can solve a sparse matrix problem with reduced fill-in element, see [5]. But if computer memory is expensive, as in most CFD application cases, then iterative methods are the best choice with special preconditioners. Now, excellent candidates are Krylov space methods, for instance BICGSTAB or GMRES, which however show a mesh dependent convergence behaviour. A reasonable alternative is a multigrid solver that offers a rather different approach and substantial benefits such as it is potentially most effficient in terms of computational cost.

This article is structured as follows: Section 2 describes briefly the problem formulations of Stokes/Navier-Stokes type with energy equation or with viscoelastic models of Oldroyd-B and Giesekus type. The variational formulation is described in the following section 3 which is followed by the FEM discretization in section 4. The Newton method for handling the nonlinear equations is described in section 5, then followed by the multigrid solver with Vanka smoother in section 6. Finally, section 7 presents the numerical tests and a summary is drawn in the final section 8.

2 Problem formulation

We consider steady problems of generalised Newtonian flow that satisfy

$$(\mathbf{VP}) \quad \begin{cases} (\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \nabla\cdot\mathbf{T} \\ \nabla\cdot\mathbf{u} = 0 \end{cases} \tag{2}$$

with the constitutive law $\mathbf{T} = 2\eta_s(\dot\gamma, p)\mathbf{D}$, shear rate $\dot\gamma = \sqrt{\mathrm{tr}(\mathbf{D}^2)}$, and symmetric velocity gradient $\mathbf{D} = \frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathrm{T}})$. For convenience, we call it velocity-pressure problem ($\mathbf{VP}$). The fluid viscosity function will lead later to the power law and Cross model. The temperature effects of the flow (non-isothermal) are written in terms of the Boussinesq approximation where the coupling with the above Navier-Stokes equations looks like as follows

$$(\mathbf{VTP}) \quad \begin{cases} (\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \nabla\cdot\mathbf{T} \\ \nabla\cdot\mathbf{u} = 0 \\ (\mathbf{u}\cdot\nabla)\Theta = k_1\nabla^2\Theta + k_2\mathbf{D}:\mathbf{D} \end{cases} \tag{3}$$

with additional friction term $\mathbf{D} : \mathbf{D}$. Here, k_1 and k_2 are the thermal diffusivity and friction parameters. Now, we call it velocity-temperature-pressure problem (**VTP**) where the nonlinear viscosity may additionally depend on temperature, $\eta_s(\dot{\gamma}, \Theta, p)$. Furthermore, complex fluid flow problems in the area of viscoelasticity are also considered and are of the following form

$$(\mathbf{VSP}) \quad \begin{cases} (\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \eta_s \Delta \mathbf{u} + \dfrac{\eta_p}{\Lambda}\nabla\cdot\boldsymbol{\tau} \\ \nabla\cdot\mathbf{u} = 0 \\ (\mathbf{u}\cdot\nabla)\boldsymbol{\tau} - \nabla\mathbf{u}\cdot\boldsymbol{\tau} - \boldsymbol{\tau}\cdot\nabla\mathbf{u}^T + \dfrac{1}{\Lambda}(\boldsymbol{\tau} - \mathbf{I} + \alpha(\boldsymbol{\tau}-\mathbf{I})^2) = 0 \end{cases} \tag{4}$$

where $\boldsymbol{\tau}, \Lambda$ are the conformation tensor and the relaxation time of the polymer. The parameter α is the Giesekus mobility factor which yields to the Oldroyd-B model if it is set to zero. Numerically, the relaxation time describes the complexity of the corresponding flow model which is commonly represented by the non-dimensional Weissenberg number $\mathsf{We} = \Lambda\frac{u_c}{l_c}$. The total stress now consists of viscous and elastic parts as well as the corresponding viscosities, η_s, η_p. The ratio $\beta = \frac{\eta_s}{\eta_0}$ denotes the viscous contribution to the total stress. We call this velocity-stress-pressure problem (**VSP**). Different from the set of equations (2 and 3), the viscoelastic model includes additional equations for describing the stress, hereby using the so-called conformation stress tensor, $\boldsymbol{\tau}$, as the main numerical variable. Consequently, we introduce more numerical complexity into the matrix $\mathcal{A}$.

In this study, we utilize the power law and Cross model for the (**VP**) and (**VTP**) problems as nonlinear viscosity models. These models are well-known and widely used, for instance modelling blood flows, see [6–8].

2.1 Power law model

The power law model is one of the most popular choices for modelling fluids with shear thinning/thickening properties. It can be written as

$$\eta(\dot{\gamma}) = \eta_0\ (\epsilon + \dot{\gamma}^2)^{(\frac{r}{2}-1)}. \tag{5}$$

Here, r is the constant parameter index that describes whether the fluid is in the region of shear thinning ($\frac{r}{2} < 1$) or shear thickening ($\frac{r}{2} > 1$). In this model, the viscosity may be unbounded during the numerical computation such that it no longer has a physical meaning. For this (and numerical) reason, we also test another model that bounds the viscosity, the so-called Cross model.

2.2 Generalized Cross model

This model bounds the viscosity between η_1 and η_0. Here, the jumps between the two viscosities can be affected by physical parameters such as temperature, shear rate and pressure. It is written as follows

$$\eta(\dot{\gamma}) = \eta_0 + \frac{\eta_1 - \eta_0}{(1 + (\lambda\dot{\gamma})^{r_1})^r} \exp(\alpha p)\ e^{a_1 + \frac{a_2}{a_3 + \Theta}} \tag{6}$$

where η_0 is the minimum shear viscosity which represents the behavior at very high shear. The power index r is introduced to generalize the classical cross model to cover both shear-thinning and shear-thickning. The parameter r_1 is known as the Cross Rate Constant (CRC). The CRC is dimensionless and measures the degree of the dependency of viscosity on the shear rate in the shear-thinning or the shear-thickening region. Here, it is set to a value of $r_1 = 1$. The λ parameter is known as the Cross Time Constant which we set to $\lambda = 8.2$.

3 Variational formulations

The corresponding variational formulations for the above problems are given by

$$a(\mathbf{u}, \mathbf{v}) = 2\int_\Omega \eta_s \mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{v}) dx + \int_\Omega \rho(\mathbf{u} \cdot \nabla)\mathbf{u}\ \mathbf{v} dx \tag{7}$$

$$b(\mathbf{v}, q) = -\int_\Omega q\ \ \mathrm{div}\, \mathbf{v} dx \tag{8}$$

for the standard Navier-Stokes equation (2), and

$$e(\Theta, \Phi) = k_1 \int_\Omega \nabla\Theta\, \nabla\Phi\, dx + \int_\Omega (\mathbf{u} \cdot \nabla)\Theta\ \Phi\, dx \tag{9}$$

$$h(\mathbf{u}, \Phi) = -k_2 \int_\Omega [\mathbf{D}(\mathbf{u}) : \mathbf{D}(\mathbf{u})]\Phi\, dx \tag{10}$$

for the energy equation (3) with friction. The variational form due to the stress and velocity coupling in the case of the viscoelastic flow model (4) is given by

$$c(\mathbf{v}, \boldsymbol{\tau}) = -\int_\Omega \boldsymbol{\tau}\ \ : \mathbf{D}(\mathbf{v}) dx, \tag{11}$$

while the variational form for the viscoelastic constitutive law reads as follows

$$d(\boldsymbol{\tau},\boldsymbol{\psi}) = \int_\Omega (\mathbf{u}\cdot\nabla)\,\boldsymbol{\tau} : \boldsymbol{\psi} dx + \frac{1}{\Lambda}\int_\Omega [\boldsymbol{\tau} + \alpha\boldsymbol{\tau}^2 - 2\alpha\boldsymbol{\tau}] : \boldsymbol{\psi} dx$$
$$- \int_\Omega (\nabla\mathbf{u}\cdot\boldsymbol{\tau} - \boldsymbol{\tau}\cdot\nabla\mathbf{u}^T) : \boldsymbol{\psi}\, dx. \tag{12}$$

The variational form of the source terms can be written as

$$\langle F(\tilde{\mathbf{u}},p),(\tilde{\mathbf{v}},q)\rangle = \int_\Omega \mathbf{f}\mathbf{v}\, dx - \frac{\alpha-1}{\Lambda}\int_\Omega \mathbf{I} : \boldsymbol{\psi} dx \tag{13}$$

with $\tilde{\mathbf{u}} := (\mathbf{u},\boldsymbol{\psi},\Phi)$. Let us set $X(\Omega) := [\mathbf{H}_0^1(\Omega)]^2 \times [\mathbf{L}2(\Omega)]^4 \times H^1(\Omega)$, then the coupled nonlinear problem is to find $(\tilde{\mathbf{u}},p) \in X(\Omega)\times L_0^2(\Omega)$ such that

$$\langle K(\tilde{\mathbf{u}},p),(\tilde{\mathbf{v}},q)\rangle = \langle F(\tilde{\mathbf{u}},p),(\tilde{\mathbf{v}},q)\rangle \quad \forall(\tilde{\mathbf{v}},q) \in X(\Omega)\times L_0^2(\Omega) \tag{14}$$

where

$$\begin{aligned}\langle K(\tilde{u},p),(\tilde{v},q)\rangle &= a(\mathbf{u},\mathbf{v}) - b(\mathbf{v},p) + b(\mathbf{u},q)\\ &\quad - c(\mathbf{v},\boldsymbol{\tau}) + d(\boldsymbol{\tau},\boldsymbol{\psi})\\ &\quad + e(\Theta,\Phi) + h(\mathbf{u},\Phi).\end{aligned} \tag{15}$$

4 FEM discretization

The choice of FEM spaces for the Stokes problem is subject to the well-known compatibility condition between the velocity and pressure space, the so-called $\inf - \sup$ condition named after Ladyzhenskaya, Babuška and Brezzi (LBB) [9]

$$\sup_{\boldsymbol{u}\in\mathbf{V}_h^{\boldsymbol{u}}} \frac{\int_\Omega \operatorname{div}\boldsymbol{u}\, q\, dx}{\|\boldsymbol{u}\|_{1,\Omega}} \geq \beta\, \|q\|_{0,\Omega} \quad \text{for all } q \in Q_h. \tag{16}$$

Similarly, the addition of the weak form of the constitutive equation imposes a further compatibility constraint onto the choice of the approximations spaces of the velocity-pressure-stress [10]

$$\sup_{\sigma\in\mathbf{V}_h^{\boldsymbol{\tau}}} \frac{\int_\Omega \sigma : \mathbf{D}(\boldsymbol{u})\, dx}{\|\sigma\|_{0,\Omega}} \geq \gamma\, \|\boldsymbol{u}\|_{1,\Omega} \quad \text{for all } \boldsymbol{u} \in \mathbf{V}_h^{\boldsymbol{u}}, \tag{17}$$

where β and γ are two mesh-independent constants, $\mathbf{V}_h := \mathbf{V}_h^{\boldsymbol{u}} \times \mathbf{V}_h^{\boldsymbol{\tau}} \times \mathbf{V}_h^{\Theta} \subset X(\Omega)$ and $Q_h \subset L_0^2(\Omega)$.

We choose the finite element pair Q_2P_1 which is popular for Stokes-like problems [11], see [4, 12] for implementation aspects. Here, the interpolation functions are

$$\mathbf{V}_h = \{\tilde{\boldsymbol{v}}_h := (\boldsymbol{u}_h,\boldsymbol{\tau}_h,\Theta_h) \in X(\Omega_h), \tilde{\boldsymbol{v}}_{h|T} \in [Q_2(T)]^7 \ \ \forall T \in \mathcal{T}_h\}, \tag{18}$$
$$Q_h = \{p_h \in L_0^2(\Omega_h), \quad p_{h|T} \in P_1(T) \ \ \forall T \in \mathcal{T}_h\}. \tag{19}$$

We will show later that, after FEM discretization, we end up with the same type of nonlinear saddle point problems for any set of equations (**VP**,**VTP** or **VSP**) mentioned above, namely

$$\underbrace{\begin{pmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{B} & 0 \end{pmatrix}}_{\text{Nonlinear operator } \mathcal{A}} \begin{pmatrix} \tilde{\mathbf{u}} \\ p \end{pmatrix} = \begin{pmatrix} \text{rhs } \tilde{\mathbf{u}} \\ \text{rhs } p \end{pmatrix} \tag{20}$$

where $\tilde{\mathbf{u}}$ may consist of the coefficient vector representing the velocity, temperature and conformation tensor. Here, $\mathbf{A}$ includes all convection and diffusion operators, including stabilization terms of edge-oriented type (EOFEM) (see [13, 14]).

5 Nonlinear solver

The Newton method is utilized for the linearization of the system (14). We start with a system for the residual of the nonlinear algebraic equations

$$\mathcal{R}(\mathbf{x}) = \mathbf{0}, \tag{21}$$

where $\mathbf{x}$ represents the vector of the coefficients corresponding to all physical unknowns.

To solve this system, let us apply the Newton method with damping which results in iterations of the form

$$\mathbf{x}^{n+1} = \mathbf{x}^n + \omega^n \left[\frac{\partial \mathcal{R}(\mathbf{x}^n)}{\partial \mathbf{x}} \right]^{-1} \mathcal{R}(\mathbf{x}^n). \tag{22}$$

This iteration is repeated until a certain tolerance is reached, which means that a certain norm of the residual $\|\mathcal{R}(\mathbf{x}^n)\|$ is small enough. The damping parameter $\omega^n \in (-1, 0)$ is chosen such that

$$\mathcal{R}(\mathbf{x}^{n+1}) \cdot \mathbf{x}^{n+1} \leq \mathcal{R}(\mathbf{x}^n) \cdot \mathbf{x}^n. \tag{23}$$

In order to satisfy the above inequality, this study implements a line search technique which is based on the backtracking idea, see [15, 16]. Here, a full step ($\omega = -1$) is checked whether $\mathcal{R}(\omega) = \mathcal{R}(\mathbf{x}^n + \omega\delta\mathbf{x})$ is minimized or not. If the starting vector is close to the solution then the above condition is mostly satisfied. If it is not, then at least we can save this information and use it for searching a new damping factor ω by backtracking along the Newton direction $\delta\mathbf{x}$ until $\mathcal{R}(\omega) = \mathcal{R}(\mathbf{x}^n + \omega\delta\mathbf{x})$ is minimized. To achieve this

goal, a quadratic interpolation is used to obtain the corresponding ω since we have 3 informations already: $\mathcal{R}(\omega = -1)$ from the full step test, $\mathcal{R}(0)$ when we construct the residual vector, and $\mathcal{R}'(0)$ when we construct the Jacobian matrix. This can be visualised in Fig. 1 as 1D problem. Then, the new damping

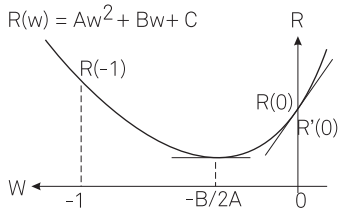


Fig. 1. The quadratic interpolation procedure.

factor reads

$$\omega_1 = \frac{-\mathcal{R}'(0)}{2(\mathcal{R}(1) - \mathcal{R}(0) - \mathcal{R}'(0))}. \tag{24}$$

By replacing $\mathcal{R}(\omega = -1)$ with the last computed information $\mathcal{R}(\omega_1)$, the next damping parameter can be iteratively computed in the same way until $\mathcal{R}(\omega_n)$ meets the criterion. The damping greatly improves the robustness of the Newton iteration in the case when the current approximation $\mathbf{x}^n$ is not close enough to the final solution since the Newton method without damping is not guaranteed to converge.

It is evident that the Newton method needs to compute the first derivative of the residual with respect to the current solution vector (sometimes it is called tangent stiffness matrix) at every Newton step and level. This can be done analytically, by means of taking the Frechet derivative of the nonlinear operator $\mathcal{A}$, or by using divided difference techniques. Here, the Jacobian matrix $\left[\frac{\partial \mathcal{R}(\mathbf{X}^n)}{\partial \mathbf{X}}\right]$ can be approximated using finite differences by

$$\left[\frac{\partial \mathcal{R}(\mathbf{x}^n)}{\partial \mathbf{x}}\right]_{ij} \approx \frac{\mathcal{R}_i(\mathbf{x}^n + \varepsilon \mathbf{e}_j) - \mathcal{R}_i(\mathbf{x}^n - \varepsilon \mathbf{e}_j)}{2\varepsilon}, \tag{25}$$

where the vector $\mathbf{e}_j$ is the standard Kronecker symbol. The parameter ε can be fixed or can be modified according to some norm of the solution $\|\mathbf{x}^n\|$ or the norm of the update in the previous step $\|\mathbf{x}^n - \mathbf{x}^{n-1}\|$. In practise, we set this parameter to be constant, i.e., $\varepsilon = \sqrt{\text{DBL\_Machine}}$.

The structure of the complete Jacobian matrix in each Newton step reads

$$\begin{bmatrix} A_{\boldsymbol{u}} & C^T & \tilde{H}^T_{\boldsymbol{u}} & \tilde{B}^T \\ \tilde{C} & A_{\boldsymbol{\tau}} & 0 & 0 \\ H_{\boldsymbol{u}} & 0 & A_\Theta & 0 \\ B & 0 & 0 & 0 \end{bmatrix} \tag{26}$$

where $A_{\boldsymbol{u}}, A_{\boldsymbol{\tau}}, A_\Theta$ correspond to the linearized operators of the velocity, stress, and temperature block of equations. The divergence operators are denoted by B and C^T for velocity and stress, respectively, while $\tilde{C}$ comes from the linearization of the stretching term. $\tilde{H}_{\boldsymbol{u}}$ and $\tilde{B}$ arise from the linearization of the nonlinear viscosity with respect to the temperature and pressure.

For generalized isothermal Newtonian problems, we can set $\tilde{\boldsymbol{u}} = \boldsymbol{u}$ and $A = A_{\boldsymbol{u}}$. For generalized non-isothermal Newtonian problems with $\tilde{\boldsymbol{u}} = (\boldsymbol{u}, \Theta)$, we set

$$A = \begin{bmatrix} A_{\boldsymbol{u}} & \tilde{H}_{\boldsymbol{u}}^T \\ H_{\boldsymbol{u}} & A_\Theta \end{bmatrix}, \quad B := \begin{bmatrix} B & 0 & 0 \end{bmatrix}, \text{ and } \quad \tilde{B} := \begin{bmatrix} \tilde{B} & 0 & 0 \end{bmatrix}. \tag{27}$$

For the generalized isothermal viscoelastic problem $\tilde{\boldsymbol{u}} = (\boldsymbol{u}, \boldsymbol{\tau})$, we write

$$A = \begin{bmatrix} A_{\boldsymbol{u}} & C^T \\ \tilde{C} & A_{\boldsymbol{\tau}} \end{bmatrix}, \quad B := \begin{bmatrix} B & 0 & 0 \end{bmatrix}, \text{ and } \quad \tilde{B} := \begin{bmatrix} \tilde{B} & 0 & 0 \end{bmatrix}. \tag{28}$$

Finally, for a generalized non-isothermal viscoelastic problem with $\tilde{\boldsymbol{u}} = (\boldsymbol{u}, \boldsymbol{\tau}, \Theta)$, we obtain

$$A = \begin{bmatrix} A_{\boldsymbol{u}} & C^T & \tilde{H}_{\boldsymbol{u}}^T \\ \tilde{C} & A_{\boldsymbol{\tau}} & 0 \\ H_{\boldsymbol{u}} & 0 & A_\Theta \end{bmatrix} \text{ and } \quad \begin{aligned} B &:= \begin{bmatrix} B & 0 & 0 & 0 \end{bmatrix} \\ \tilde{B} &:= \begin{bmatrix} \tilde{B} & 0 & 0 & 0 \end{bmatrix}. \end{aligned} \tag{29}$$

Then, the systems of linear equations to be solved for all problems, considered so far, have all the following 2×2 block structure

$$\begin{bmatrix} A & \tilde{B}^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{u}} \\ p \end{bmatrix} = \begin{bmatrix} \mathcal{R}_{\tilde{\boldsymbol{u}}} \\ \mathcal{R}_p \end{bmatrix} \tag{30}$$

where $A \in \mathcal{R}^n \times \mathcal{R}^n$, $B, \tilde{B} \in \mathcal{R}^n \times \mathcal{R}^m, m \leq n$. In order to solve this linearized system, we will apply PSC techniques which are known to be very efficient as described in [17], taking care of the zero diagonal block in the lower right part.

6 Linear multigrid solver

The way how multigrid solves a discrete linear system is facilitated by the construction of lower levels of discrete systems, which is known as standard

geometric multigrid. Multigrid components consist of a smoother, restriction, prolongation and a direct coarse grid solver (UMFPACK). The smoother plays an important role for the multigrid convergence behaviour in dealing with the Jacobian matrix

$$\mathcal{A} := \begin{bmatrix} A & \tilde{B}^T \\ B & 0 \end{bmatrix} \tag{31}$$

of the systems of the linear equations (30) which is non-symmetric and strongly coupled. Therefore, we use the concept of the PSC approach which is to solve **'exactly'** on fixed patches and to perform an outer Gauß-Seidel iteration. This approach can be interpreted as generalization of block-Jacobi/Gauß-Seidel methods for saddle point problems which contains modifications of classical schemes like the Vanka smoother [18]. The 'division technique' involved in one smoothing step can be based on one element or a cluster of some of them which leads to the so-called cell-based or patch-based Vanka smoother. In [19], there is a convergence study of different type of Vanka smoothers. In this paper, we utilize the cell-based Vanka smoother given as the following

$$\begin{bmatrix} \tilde{\boldsymbol{u}}^{l+1} \\ p^{l+1} \end{bmatrix} = \begin{bmatrix} \tilde{\boldsymbol{u}}^{l} \\ p^{l} \end{bmatrix} + \omega^l \sum_{T \in \mathcal{T}_h} [\mathcal{A}]_{|T}^{-1} \begin{bmatrix} \mathcal{R}_{\tilde{\boldsymbol{u}}} \\ \mathcal{R}_p \end{bmatrix}_{|T}, \tag{32}$$

which acts locally in exactly one element T on all levels. The above smoothing step can be explained as follows:

(1) Prepare the local matrix $[\mathcal{A}]_{|T}$ and local residual $[\mathcal{R}_{\tilde{\boldsymbol{u}}}\ \mathcal{R}_p]_{|T}^{\mathrm{T}}$. This step requires to extract a local matrix from the global one. The global matrix contains already information whether an element is in the interior or boundaries. So, no extra work to do. In parallel to this, a local residual is built by including the updated information from the last visited element.
(2) Then we obtain a local correction $[\mathcal{A}]_{|T}^{-1}\ [\mathcal{R}_{\tilde{\boldsymbol{u}}}\ \mathcal{R}_p]_{|T}^{\mathrm{T}}$.
(3) We finish with updating the correction to the global solution described in equation (32) by multiplying it with a damping parameter before going to the next element.

After a few smoothing steps, an approximation $\mathbf{x}^h$ of the solution is obtained. The unknown error of this approximation ($\mathbf{e}^h = \mathbf{x} - \mathbf{x}^h$) may be smooth, but may not necessarily be so on the coarser level. Then, the restriction transfers the approximate error onto the coarser level $\mathbf{e}^{2h}$, which will then be smoothed. This action continues until we obtain the smoothed error on the coarsest grid, $\mathbf{e}^c$. A 'V-cycle' will visit one time the coarsest grid and a 'F-cycle' may be more than just a time. On the coarsest grid, the linear system of the residual equation $\mathcal{A}(\mathbf{x}^n)\mathbf{e}^c = \mathbf{r}(\mathbf{x}^n)$ is solved by a direct sparse solver (UMFPACK). In order to be efficient, the coarsest grid should be coarse enough so that the UMFPACK memory requirement is as low as possible. Then, prolongation is

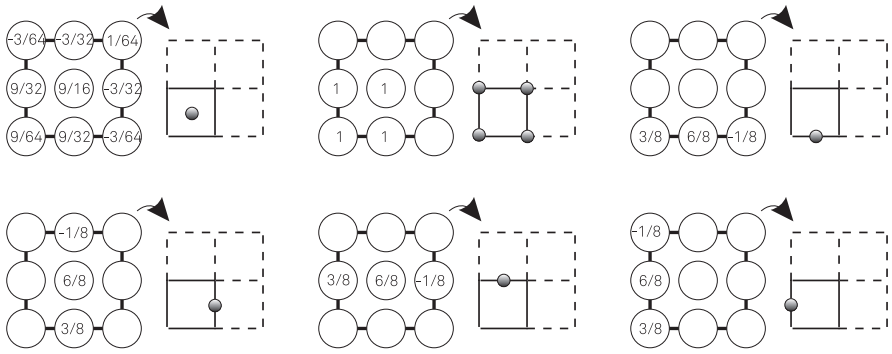


Fig. 2. Prolongation in Q_2 with biquadratic interpolation.

applied which is followed by post-smoothing to give a better error approximation back to the finest level. These steps continue until an 'F-cycle' of multigrid iterations is finished. In the case of conforming finite elements, the lower level space is a subspace of the finer level one. Thus, natural injection can be applied for the restriction operator. The prolongation operator on the other hand needs to be constructed by biquadratic interpolation, see Fig. 2, and analogously for the piecewise linear pressure. In order to demonstrate the solver flexibility with respect to different flow problems, an 'F-cycle' of multigrid is used with 4 smoothing steps in the following numerical tests.

7 Numerical tests

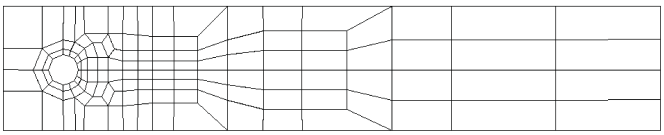
We test the solver behaviour for the geometrical configuration given in Fig. 3 which is motivated by the well-known flow around cylinder benchmark [1,3,4, 17]. The inflow/outflow setting and a lot of details can be further seen also on http://featflow.de/en/benchmarks/cfdbenchmarking.html which leads to the classical case of $\mathsf{Re} = 20$ if we consider solving the (Newtonian) Navier-Stokes equation with constant kinematic viscosity of $\eta_0 = 1e-3$. The accuracy of the solutions is controlled by the drag and lift values

$$C_{\text{drag/lift}} = C \int_s \boldsymbol{\sigma} \cdot \mathbf{n} \; ds \tag{33}$$

with a constant $C = 2F_{d/l}/(\rho u_c^2 l_c)$ obtained from the settings $F_{d/l} = 1$, $u_c = 0.2$, $l_c = 0.1, \rho = 1$, see [1].

7.1 Velocity-pressure problem

In the (**VP**) problem, we analyze the Newton-multigrid behaviour for the steady Stokes and Navier-Stokes equations with both constant and nonlinear



Lev.	NEL	NVT	NMP
L1	520	572	1092
L2	2080	2184	4264
L3	8320	8528	16848
L4	33280	33696	66976
L5	133120	133952	267072

Fig. 3. 'Flow around cylinder' coarse mesh (L0). NEL, NVT and NMP denote the number of elements, vertices and midpoints.

viscosity function. The total number of equations for this problem can be seen in Tab. 1.

Table 1
Total number of equations (NEQ) for the (**VP**) problem

Lev.	NEQ **u**	NEQ p	total NEQ
L0	1144	390	1534
L1	4368	1560	5928
L2	17056	6240	23296
L3	67392	24960	92352
L4	267904	99840	367744
L5	1068288	399360	1467648

The solver behaves as expected for the linear Stokes problem (see Tab. 2 on the left) and for the nonlinear Navier-Stokes problem at $\mathsf{Re} = 20$ (see Tab. 2 on the right). Here, 'N' denotes the number of nonlinear iteration where 'L' gives the average number of multigrid steps per nonlinear cycle, and we set a zero vector as initial solution. For the Stokes problem, the linear solver gains 8 digits of linear precision and the Newton method solves the problem within one step to satisfy 8 digits of nonlinear precision.

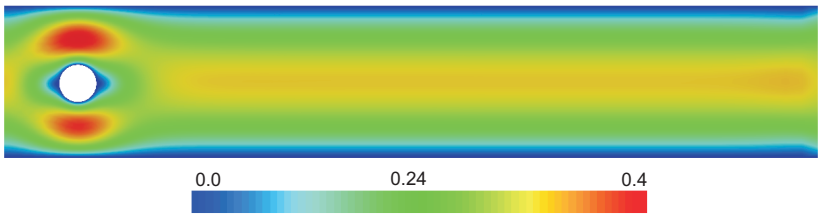


Fig. 4. Velocity magnitude of Stokes flow (Tab. 2).

Table 2
Stokes: Newton-Multigrid behaviour in terms of N/L with constant viscosity

Lev.	Drag/Lift		N/L	Drag/Lift		N/L
	Stokes			Navier-Stokes Re=20		
L1	3.112646	2.965870e-02	1/6	5.540999	9.447473e-03	5/2
L2	3.134342	3.005275e-02	1/7	5.566928	1.046885e-02	5/2
L3	3.140327	3.015909e-02	1/7	5.576088	1.056787e-02	5/2
L4	3.141893	3.018665e-02	1/7	5.578652	1.060398e-02	5/2
L5	3.142292	3.019366e-02	1/7	5.579313	1.061503e-02	5/2

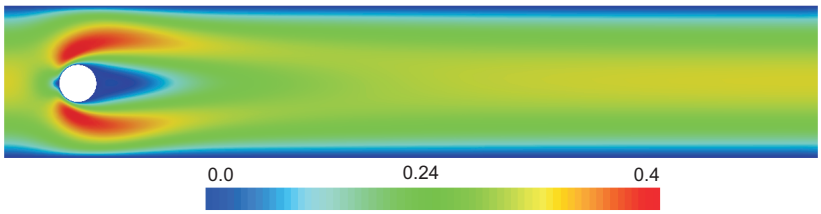


Fig. 5. Velocity magnitude of Navier-Stokes flow at $\mathsf{Re} = 20$ (Tab. 2).

In contrast, the linear solver is required to gain 2 digits for the Navier-Stokes problem so that the Newton method needs 5 cycles to satisfy the same nonlinear precision. The produced drag/lift coefficients recover the results in [1] and lead to mesh independent results.

Next, we increase the nonlinearity of the problem by introducing a nonlinear viscosity function. Here and in the following nonlinear test configurations, we perform a nested start procedure to give a better starting solution for the last 4 levels (L2 upto L5), namely by using the solution from one level below as start values. The required precision is kept the same as before (2 and 8 digits for linear and nonlinear precision, respectively). We present in the following the convergence behaviour when solving the power law model where the typical viscosity distribution is depicted in Fig. 6. In this model, shear thinning and shear thickening can be treated by setting $r = 1.5$ and $r = 3$, respectively, both configurations can be simulated without any problem (see Tab. 3).

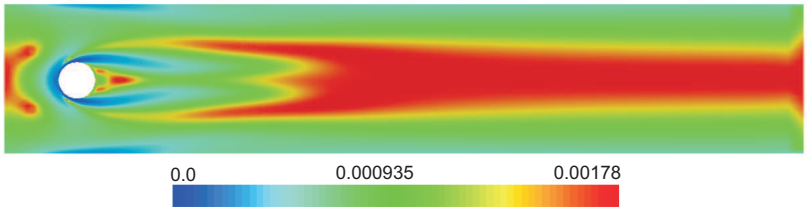


Fig. 6. Typical viscosity distribution of the power law model for $r = 1.5$ (Tab. 3).

Here, all computations lead to level independent solutions with a quite robust iterative behaviour, even with decreasing regularization parameter ϵ. As mentioned in the problem formulation, the nonlinear viscosity can be also presented by the Cross model (see Fig. 7). In the following, we present the corresponding convergence behaviour when using the Cross model which now additionally depends on the shear rate and the pressure and which can be simulated without any problem, too. Here again, all computations lead to mesh independent solutions with a very robust iterative solver behaviour and work quite independently of the nonlinear viscosity function.

Table 3
Power law model: Shear thinning and thickening

Lev.	Drag/Lift		N/L	Drag/Lift		N/L
	r=1.5			r=3.0		
$\epsilon = 10^{-1}$						
L1	3.19922	-0.01238	9/3	13.66925	0.34424	7/2
L2	3.26246	-0.01334	4/2	13.78575	0.35232	3/2
L3	3.27553	-0.01335	3/2	13.81625	0.35148	3/2
L4	3.27781	-0.01332	2/2	13.82490	0.35248	3/1
L5	3.27833	-0.01332	2/2	13.82715	0.35294	2/1
$\epsilon = 10^{-2}$						
L1	3.20071	-0.01256	9/2	13.62575	0.34267	8/2
L2	3.26420	-0.01339	4/2	13.74280	0.35070	3/2
L3	3.27728	-0.01341	3/2	13.77355	0.34963	3/2
L4	3.27956	-0.01338	2/2	13.78220	0.35062	3/1
L5	3.28007	-0.01337	2/2	13.78445	0.35112	2/2
$\epsilon = 10^{-4}$						
L1	3.20082	-0.01261	14/3	13.62085	0.34250	12/3
L2	3.26433	-0.01342	4/2	13.73800	0.35052	3/2
L3	3.27739	-0.01342	3/2	13.76875	0.34941	3/2
L4	3.27968	-0.01339	2/2	13.77740	0.35040	3/1
L5	3.28019	-0.01338	2/2	13.77970	0.35091	2/2

Table 4
Cross model: Shear and pressure dependent viscosity with $\eta_0 = 10^{-3}$

Lev.	Drag/Lift		N/L	Drag/Lift		N/L
	$\eta_1 = 10^{-2}$			$\eta_1 = 10^{-1}$		
$r = 1, \alpha = 0$						
L1	6.28236	0.02352	9/1	14.85985	0.13033	11/2
L2	6.31313	0.02478	3/1	14.91585	0.13672	4/2
L3	6.32337	0.02504	3/2	14.93680	0.13750	4/2
L4	6.32619	0.02509	3/2	14.94270	0.13765	4/3
L5	6.32691	0.02510	2/2	14.94420	0.13769	3/2
$r = 0, \alpha = 0.1$						
L1	32.99952	0.80985	6/2	529.81550	6.43810	10/2
L2	33.22763	0.81901	2/2	534.29750	6.53247	3/2
L3	33.29026	0.82140	2/2	535.48500	6.55813	3/3
L4	33.30657	0.82201	2/2	535.77950	6.56464	3/3
L5	33.31069	0.82217	2/3	535.84800	6.56621	2/2
$r = 1, \alpha = 0.1$						
L1	6.28691	0.02359	11/1	15.10777	0.13223	12/2
L2	6.31772	0.02485	3/1	15.16395	0.13886	4/3
L3	6.32797	0.02512	3/2	15.18516	0.13963	4/3
L4	6.33079	0.02517	3/2	15.19108	0.13978	4/3
L5	6.33151	0.02518	2/2	15.19262	0.13982	3/3

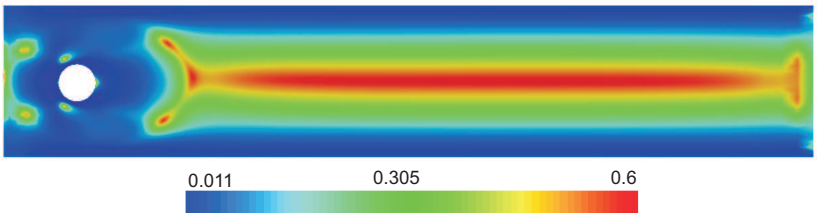


Fig. 7. Typical viscosity distribution of the Cross model with $\eta_0 = 10^{-3}, \eta_1 = 10^{-1}$ (Tab. 4).

7.2 Velocity-temperature-pressure problem

In the following, we analyze the solver behaviour for the (**VTP**) problem. The corresponding number of total unknowns can be seen from Tab. 5 which is half a million more unknowns in comparison to the (**VP**) problem on the finest level or roughly 33% more unknowns on each level.

Table 5
Total number of equations for the (**VTP**) problem

Lev.	NEQ **u**	NEQ Θ	NEQ p	total NEQ
L0	1144	572	390	2106
L1	4368	2184	1560	8112
L2	17056	8528	6240	31824
L3	67392	33696	24960	126048
L4	267904	133952	99840	501696
L5	1068288	534144	399360	2001792

For this example, we firstly include the viscous dissipation and secondly we present a heated cylinder configuration with additional viscous dissipation. In the first example, the boundary setting for the temperature is set in accordance to the inflow as Dirichlet data, hereby prescribing zero temperature. On the other solid walls and the outflow section, the temperature is set to natural boundary conditions.

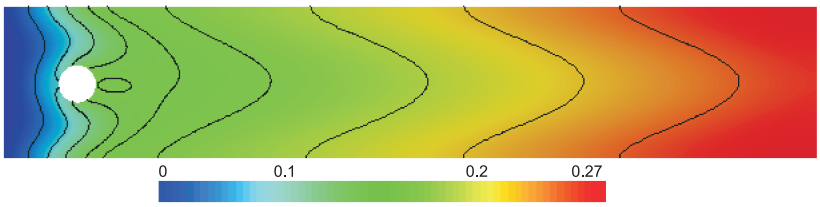


Fig. 8. Temperature field generated by viscous dissipation (Tab. 6).

In the second example, we set the same boundary data as in the first example, but prescribing no-slip condition to solid walls and setting inhomogeneous Dirichlet data at the cylinder.

Table 6
Viscous dissipation: Temperature, shear and pressure dependent viscosity with $\eta_0 = 10^{-3}, k_1 = k_2 = 10^{-2}, a_1 = 0, a_3 = 1$

Lev.	Drag/Lift		N/L	Drag/Lift		N/L
	$\eta_1 = 10^{-2}$			$\eta_1 = 10^{0}$		
Temperature dependent $r = 0, \alpha = 0, a_2 = 1$						
L1	73.79610	1.30217	11/2	7815.320	76.51365	17/2
L2	74.29465	1.31636	2/2	7868.420	77.50415	3/2
L3	74.43290	1.32009	2/2	7883.125	77.77045	3/2
L4	74.46910	1.32105	2/2	7886.975	77.83945	3/2
L5	74.47830	1.32129	2/2	7887.955	77.85700	3/2
Temperature and shear dependent $r = 0.1, \alpha = 0, a_2 = 1$						
L1	53.44090	1.04442	8/2	5543.080	54.16830	8/2
L2	53.78930	1.05488	2/2	5579.355	54.79350	3/2
L3	53.88590	1.05770	3/2	5589.415	54.96815	3/2
L4	53.91125	1.05844	2/2	5592.050	55.01415	3/2
L5	53.91770	1.05863	2/2	5592.725	55.02585	3/2
Temperature, pressure and shear dependent $r = 0.1, \alpha = 10^{-3}, a_2 = 1$						
L1	53.47400	1.04485	8/2	5965.580	59.06025	7/2
L2	53.82265	1.05532	2/2	6005.265	59.75125	3/2
L3	53.91930	1.05814	3/2	6016.220	59.94455	3/2
L4	53.94465	1.05888	2/2	6019.075	59.99535	3/2
L5	53.95115	1.05907	2/2	6019.795	60.00820	3/2

Table 7
Heated cylinder with viscous dissipation: Temperature, shear and pressure dependent viscosity with $\eta_0 = 10^{-3}, r = 0.1, \alpha = 10^{-3}, a_1 = 0, a_2 = 1, a_3 = 1$

Lev.	Drag/Lift		N/L	Drag/Lift		N/L
	$k_1 = k_2 = 10^{-2}$			$k_1 = k_2 = 10^{-3}$		
$\eta_1 = 10^{-2}$						
L1	44.97456	0.89313	6/1	49.28800	0.95093	7/2
L2	45.26969	0.90303	3/2	49.59222	0.96042	2/2
L3	45.35251	0.90563	3/2	49.68249	0.96310	3/2
L4	45.37431	0.90632	2/2	49.70631	0.96381	2/2
L5	45.37988	0.90649	2/2	49.71239	0.96399	2/2
$\eta_1 = 10^{-1}$						
L1	461.05555	4.85446	7/2	509.6785	5.32184	8/2
L2	464.07865	4.90752	3/2	512.7765	5.37640	3/2
L3	464.93045	4.92313	3/2	513.7120	5.39301	3/2
L4	465.15470	4.92724	3/2	513.9585	5.39743	3/2
L5	465.21195	4.92828	3/2	514.0215	5.39856	3/2
$\eta_1 = 10^{0}$						
L1	4955.5470	47.27252	7/2	5495.220	52.56860	8/2
L2	4988.3705	47.80207	3/2	5528.860	53.11685	3/2
L3	4997.5880	47.96040	3/2	5539.025	53.28790	3/2
L4	5000.0000	48.00201	3/2	5541.690	53.33335	3/2
L5	5000.6100	48.01254	5/2	5542.365	53.34490	3/2

In the first example, the resulting temperature field is generated by the viscous dissipation during the flow motion, which contributes to the nonlinear viscosity function. In the second example, the heat is not only arising from the viscous dissipation but also coming from the cylinder. Note that the temperature boundary conditions are different between the two examples, so that qualitative comparisons can not be drawn from Fig. 8 and 9. The solver behaves again very stable when solving such a larger and more complex system, hereby adding other physical effects due to the temperature. The behaviour is tested with different nonlinear viscosity parameters of the Cross model which leads to very satisfactory results. The analogous behaviour can be observed in Tab. 7 when heat is additionally produced at the cylinder.

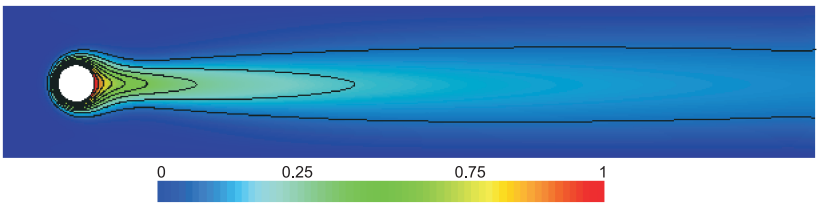


Fig. 9. Temperature field generated by the heated cylinder and viscous dissipation (Tab. 7).

7.3 Velocity-stress-pressure problem

In this subsection, we finally present the resulting solver behaviour when dealing with the above mentioned (**VSP**) problem. Here, the total number of unknowns increases roughly by a factor of 3 compared to the (**VP**) problem as can be seen in Tab. 8.

Table 8
Total number of equations for (**VSP**) problem

Lev.	NEQ $\mathbf{u}$	NEQ $\boldsymbol{\tau}$	NEQ p	total NEQ
L0	1144	1716	390	3250
L1	4368	6552	1560	12480
L2	17056	25584	6240	48880
L3	67392	101088	24960	193440
L4	267904	401856	99840	769600
L5	1068288	1602432	399360	3070080

We test two types of viscoelastic models, namely Oldroyd-B and Giesekus. The first model is well-known to have a limit of We numbers from which on the solutions are no longer mesh-independent, see [20–23]. The later model shows a much more stable behaviour with respect to mesh-independent solutions. Here, Tab. 9 presents the different convergence behaviour for the two models when We gets larger. For small We, i.e., up to $\mathsf{We} = 0.02$, the convergence behaviour between the two models is quite similar. They need 1 or 2 linear

multigrid iterations only. For $\mathsf{We} = 0.2$ the number of nonlinear steps starts to increase for Oldroyd-B at level 1 but the number of linear steps is still quite similar between the two models. For $\mathsf{We} = 1.0$ the Oldroyd-B model needs clearly more linear iteration steps than the Giesekus model, and the mesh convergence solution is deteriorated for the Oldroyd-B model (see [23] for more details).

Table 9
Viscoelastic models: Oldroyd-B and Giesekus ($\alpha = 0.1$) at $\beta = 0.5$

Lev.	Drag/Lift		N/L	Drag/Lift		N/L
	Oldroyd-B			Giesekus		
$\mathsf{We} = 0.002$						
L1	5.54787	0.00896	7/3	5.54076	0.00924	8/1
L2	5.57150	0.01031	2/2	5.56474	0.01053	2/2
L3	5.58032	0.01047	3/2	5.57511	0.01064	2/2
L4	5.58285	0.01051	2/2	5.57936	0.01062	2/2
L5	5.58351	0.01052	2/2	5.58131	0.01059	2/2
$\mathsf{We} = 0.02$						
L1	5.57197	0.00826	8/2	5.56721	0.00847	8/1
L2	5.59864	0.00967	2/2	5.59216	0.00975	2/2
L3	5.60914	0.00986	2/2	5.60186	0.00987	2/2
L4	5.61282	0.00990	2/1	5.60515	0.00989	2/1
L5	5.61412	0.00991	2/2	5.60624	0.00989	2/1
$\mathsf{We} = 0.2$						
L1	5.84074	0.00404	14/2	4.93283	-0.00036	9/2
L2	5.89550	0.00358	4/2	4.98376	0.00464	4/2
L3	5.90790	0.00432	3/2	4.98415	0.00507	3/2
L4	5.89463	0.00510	3/2	4.98230	0.00523	3/3
L5	5.87765	0.00576	3/3	4.98286	0.00524	2/2
$\mathsf{We} = 1.0$						
L1	19.64871	0.30280	33/9	5.09704	-0.00266	14/2
L2	18.97834	0.26733	7/8	5.03961	-0.00172	4/2
L3	16.06191	0.17796	6/8	4.93834	-0.00210	4/3
L4	13.56462	0.09464	6/10	4.84483	-0.00252	3/3

8 Summary

This work has concentrated on the very robust and efficient convergence behaviour of a special class of Newton-multigrid solvers for solving stationary incompressible flow models which include a nonlinear viscosity function and additionally non-isothermal as well as viscoelastic flow models. The discretization in space is based on the LBB-stable Q_2P_1 finite element approach to maintain highly accurate solutions while the resulting discrete systems are solved iteratively which is the main aim of this work. More specific, the goal of this study is to analyze the excellent convergence behaviour for solving

complex flow models with the help of Newton-multigrid techniques as iterative solver. The outer Newton method is responsible to provide superlinear up to quadratic convergence for the nonlinear problem when the initial solution is close enough. On the other hand, the proposed geometric multigrid is responsible to treat the linearized algebraic systems in a very efficient way having utilized PSC techniques (see [17,19]). It is shown that this monolithic finite element framework with a high order Stokes element is very robust with respect to convergence and accuracy in dealing with the above mentioned constitutive models so that highly efficient solvers for stationary as well as nonstationary incompressible flow problems with complex rheological behaviour are available. Currently, we extend this work to 3D flow configurations, including also Level Set techniques to couple complex rheological behaviour with multiphase flow.

References

[1] Turek, S., Schäfer, M., Benchmark computations of laminar flow around cylinder, in: E. H. Hirschel (Ed.), Flow Simulation with High-Performance Computers II, Vol. 52 of Notes on Numerical Fluid Mechanics, Vieweg, 1996, pp. 547–566, co. F. Durst, E. Krause, R. Rannacher.

[2] John, V., Higher-order finite element discretizations in a benchmark problem for incompressible flows, Int. J. Numer. Meth. Fluids 37 (2001) 885–903.

[3] John, V., Higher order finite element methods and multigrid solvers in a benchmark problem for the 3-D Navier-Stokes equations, Int. J. Numer. Meth. Fluids 40 (2002) 775–798.

[4] Turek, S., Hron, J., A monolithic FEM solver for an ALE formulation of fluid–structure interaction with configuration for numerical benchmarking, in: P. Wesseling, E. Onate, J. Periaux (Eds.), Books of Abstracts European Conference on Computational Fluid Dynamics, nn, 2006, p. 176, Eccomas CFD 2006.

[5] Davis, T. A., Algorithm 832: Umfpack, an unsymmetric-pattern multifrontal method, ACM Trans. Math. Softw. 30(2) (2004) 196–199.

[6] Hron, J., Malek, J., Rajagopal, K. R., Simple flows of fluids with pressure-dependent viscosities, Proc. Roy. Soc. Lond. A 457 (2001) 1603–1622.

[7] Hron, J., Malek, J., Turek, S., A numerical investigation of flows of shear-thinning fluids with applications to blood rheology, International Journal for Numerical Methods in Fluids 32 (2000) 863–879.

[8] Bodnár, T., Sequeira, A., Numerical study of the significance of the non-newtonian nature of blood in steady flow through a stenosed vessel, Advances in Mathematical Fluid Mechanics (2010) 83–104doi:10.1007 978-3-642-04068-9 6.

[9] Girault, V., Raviart, P. A., Finite Element Methods for Navier-Stokes equations, Springer, 1986, Berlin-Heidelberg.

[10] Brezzi, F., Fortin, M., Mixed and Hybrid Finite Element methods, Springer, Berlin, 1986.

[11] Arnold, D.N., Boffi, D., Falk, R.S., Approximation by Quadrilateral Finite Elements, Math. Comput. 71 (239) (2002) 909–922.

[12] Damanik, H., Hron, J., Ouazzi, A., Turek, S., A monolithic FEM–multigrid solver for non-isothermal incompressible flow on general meshes, Journal of Computational Physics 228 (2009) 3869–3881.

[13] Turek, S., Ouazzi, A., Unified edge–oriented stabilization of nonconforming FEM for incompressible flow problems: Numerical investigations, J. Numer. Math. 15 (2007) 299–322.

[14] Bonito, A., Burman, E., A continuous interior penalty method for viscoelastic flows, Siam Journal of Scientific Computing 30 (2008) 1156–1177.

[15] Dennis, Jr., J. E., Schnabel, R. B., Numerical Methods for Unconstrained Optimization and Nonlinear Equations, SIAM, 1996.

[16] Press, W. H., Teukolsky, S., Vetterling, W. T., Flannery, B. P., Numerical Recipes in C++. The Art of Scientific Computing, Cambridge University Press, 2002.

[17] Turek, S., Efficient solvers for incompressible flow problems: An algorithmic and computational approach, Springer, 1999, LNCSE 6.

[18] Vanka, S. P., Implicit Multigrid Solutions of Navier-Stokes Equations in Primitive Variables, J. of Comp. Phys. 65 (1985) 138–158.

[19] H. Wobker, S. Turek, Numerical studies of Vanka–type smoothers in computational solid mechanics, Advances in Applied Mathematics and Mechanics 1 (1) (2009) 29–55.

[20] Hulsen, M. A., Fattal, R., Kupferman, R., Flow of viscoelastic fluids past a cylinder at high Weissenberg number: Stabilized simulations using matrix logarithms, Journal of Non-Newtonian Fluid Mechanics 127 (2005) 27–39.

[21] Fan, Y., Yang, H., Tanner, R. I., Stress boundary layers in the viscoelastic flow past a cylinder in a channel: limiting solutions, Acta Mech Sinica 21 (2005) 311–321.

[22] Afonso, A., Oliveira, P. J., Pinho, F. T., Alves, M. A., The log-conformation tensor approach in the finite-volume method framework, Journal of Non-Newtonian Fluid Mechanics 157 (2009) 55–65.

[23] Damanik, H., Hron, J., Ouazzi, A., Turek, S., A monolithic FEM approach for the log-conformation reformulation (lcr) of viscoelastic flow problems, Journal of non-Newtonian Fluid Mechanics 165 (2010) 1105–1113.