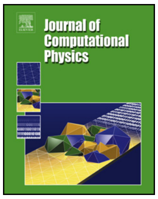
Journal of Computational Physics 442 (2021) 110436

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp



Robust stabilised finite element solvers for generalised Newtonian fluid flows

Richard Schussnig [a,b,*], Douglas R.Q. Pacheco [b,c], Thomas-Peter Fries [a,b]

[a] *Institute of Structural Analysis, Graz University of Technology, Lessingstraße 25/II, 8010 Graz, Austria*
[b] *Graz Center of Computational Engineering, Krenngasse 37/I, 8010 Graz, Austria*
[c] *Institute of Applied Mathematics, Graz University of Technology, Steyrergasse 30/III, 8010 Graz, Austria*

ARTICLE INFO

Article history:
Available online 17 May 2021

Keywords:
Stabilised finite elements
Generalised Newtonian fluid
Incompressible flow
Adaptive time-stepping
Schur complement preconditioner
Navier–Stokes equations

ABSTRACT

Various materials and solid-fluid mixtures of engineering and biomedical interest can be modelled as generalised Newtonian fluids, as their apparent viscosity depends locally on the flow field. Despite the particular features of such models, it is common practice to combine them with numerical techniques originally conceived for Newtonian fluids, which can bring several issues such as spurious pressure boundary layers, unsuitable natural boundary conditions and coupling terms spoiling the efficiency of nonlinear solvers and preconditioners. In this work, we present a finite element framework dealing with such issues while maintaining low computational cost and simple implementation. The building blocks of our algorithm are (i) an equal-order stabilisation method preserving consistency even for lowest-order discretisations, (ii) robust extrapolation of velocities in the time-dependent case to decouple the rheological law from the overall system, (iii) adaptive time step selection and (iv) a fast physics-based preconditioned Krylov subspace solver, to tackle the relevant range of discretisation parameters including highly varying viscosity. Selected numerical experiments are provided demonstrating the potential of our approach in terms of robustness, accuracy and efficiency for problems of practical interest.

1. Introduction

Complex fluids such as blood and polymer melts can respond to stresses in different ways depending on their composition and the flow regime. In hemodynamics, while the assumption of Newtonian behaviour is generally reasonable in large arteries, non-Newtonian characteristics can play an essential role in smaller vessels, especially in the presence of morphological changes such as aneurysms and stenoses [1,2]. The most popular approach when modelling such features is to consider a generalised Newtonian (also known as quasi-Newtonian) behaviour. This consists of allowing for local variations of the viscosity depending on the instant, local shear rate. Alternative models exist that can also account for viscoelasticity [1,3,4], but they introduce an additional tensor-valued field that more than doubles the number of unknowns in the problem. The quasi-Newtonian approach is thus preferred in most practical applications, as it is capable of reproducing important hemodynamic phenomena such as shear thinning and plug flow, while maintaining a low computational cost. Similar models are also popular for polymeric flows [5].

---

[*] Corresponding author at: Institute of Structural Analysis, Graz University of Technology, Lessingstraße 25/II, 8010 Graz, Austria.
E-mail address: schussnig@tugraz.at (R. Schussnig).

https://doi.org/10.1016/j.jcp.2021.110436

Regardless of the viscous model employed, the finite element basis functions used for velocity and pressure must be chosen carefully in incompressible flows. Selecting the same polynomial orders for both flow quantities violates the famous Ladyzhenskaya–Babuška–Brezzi (LBB) condition and therefore results in unstable methods. Nonetheless, since the findings of Hughes and co-workers in the 1980's [6,7], it has become a standard to use residual-based stabilisation methods such as pressure-stabilised Petrov–Galerkin (PSPG) [6] or Galerkin least-squares (GLS) [7] to circumvent the LBB condition and allow equal-order interpolation of all variables. Most stabilisation methods, however, rely on relaxed incompressibility equations devised under the assumption of constant viscosity. Perhaps for that reason, the use of LBB-compatible finite element spaces is somewhat more common in the literature for generalised Newtonian fluids [8–17]. Another stable class of discretisations is that of discontinuous Galerkin (DG) methods, which due to their local formulation are not subject to the classic LBB condition [18]. The use of these methods for quasi-Newtonian flow problems, although still incipient, has been successfully reported [19–22]. High-order DG schemes are nowadays among the most robust tools in computational fluid dynamics (CFD), as they combine the local conservation properties of finite volumes with the great versatility and accuracy of finite elements [23]. Moreover, they provide a natural framework for mesh and degree adaptivity, which is particularly advantageous in under-resolved flows. For recent overviews on high-order and discontinuous Galerkin methods in CFD, see [23–25].

While discontinuous and compatible finite elements offer an ideal setting from a mathematical standpoint (due to e.g. natural stability, optimal convergence and/or local conservation), they require data structures that are not always available in standard open source codes. Hence, there is still a great practical appeal in the use of first-order elements – especially in biomedical applications, where higher-order meshes are rarely an option. There are a few stabilisation methods which are solely based on pressure and can therefore be applied to non-Newtonian problems straightforwardly. The most popular of them, the penalty method, relaxes incompressibility to decouple velocity and pressure. Although used quite often for generalised Newtonian fluids [26–29], this approach yields poor pressure approximations even with quadratic discretisations, as shown by Sobhani et al. [29]. The pressure Poisson stabilisation of Brezzi and Pitkäranta [30] was reported by Knauf et al. [31] to also perform poorly in the non-Newtonian case, inducing spurious pressure boundary layers. This issue can be overcome by the pressure gradient projection method by Codina and Blasco [32], but at the cost of nearly doubling the size of the system to be solved. The local pressure projection method by Dohrmann and Bochev [33] is a more viable alternative and has been used successfully in non-Newtonian hemodynamics [2]. Another simple and efficient equal-order approach is our recently introduced generalised pressure Poisson framework [34] decoupling velocity and pressure. Although residual-based formulations such as PSPG and variational multiscale (VMS) methods may also be employed [35–38], their incomplete residual for linear elements can also lead to numerical artifacts, as we shall discuss later on.

To the best of our knowledge, there are only a few works in the literature dedicated to developing stabilisation techniques for the generalised Newtonian problem. Most of them require treating the viscous stress as an additional tensor-valued unknown [39,40], which leads to prohibitive computational costs in three-dimensional problems. The variational multi-scale approach devised by Masud and Kwack [36] is a more viable alternative, but – as all standard residual-based methods – can suffer from spurious pressure boundary layers and poor conservation in low-order discretisations [41]. In this context, the present work introduces a fully consistent residual-based stabilisation for equal-order finite element approximations of quasi-Newtonian problems. Our method combines a generalised Laplacian form of the momentum equation with a stabilised incompressibility-preserving continuity equation. The resulting stabilisation method can be seen as a modified version of the PSPG formulation to retain the viscous part of the residual even for linear elements. Moreover, even when used for higher-order elements, our method does not require second-order derivatives, which is an advantage from the computational standpoint. Besides the complete residual (i.e., full consistency), a major advantage of our method is not relaxing incompressibility, which leads to accurate results over a wide range of stabilisation parameters.

Regarding the fast solution of the resulting problems, a typical standpoint might be (see e.g. [42,43]) that (block) solvers coupling the velocity and pressure fields within the fluid are preferred for stationary flow, while instationary problems are better tackled using projection schemes decoupling those fields [44]. Nonetheless, one is faced with restrictions on the time step size when using projection schemes, limiting their performance in some cases even if state-of-the-art parallel multigrid solvers are used. On the other hand, preconditioning the resulting linear systems becomes a critical task when considering the coupled approach. In the past decades, however, well-performing algorithms have been developed which are especially efficient in instationary problems (see e.g. [45–48]). Within the class of methods coupling velocity and pressure, multigrid-preconditioned Krylov subspace methods have proven to be among the fastest and most robust solution methods. Therein, the key lies in successfully approximating the Schur complement of the block system, which can be achieved in a variety of ways.

Patankar and Spalding [49] outlined the so-called SIMPLE method, which was later used as a preconditioner by Vuik et al. [50] and further improved by ur Rehman et al. [51]. Satisfactory results were achieved especially for instationary problems or low-Reynolds flows, but performance is somewhat limited for elements with high aspect ratio since the approximation of the velocity-velocity block uses its diagonal components only. Another approach was taken by Silvester et al. [52] and Kay et al. [53], with a convection-diffusion operator on the pressure space defined analogously to the discrete velocity operator to mimic the action of the inverse Schur complement. This approach is directly applicable to stabilised flow problems, while the purely algebraic counterpart [54] was later extended to be used together with stabilised formulations [55]. A third family of methods is based on manipulating the resulting system in a consistent way, such that the difficulty of approximating the Schur complement is simplified considerably, shifting the numerical effort to the approximation of the velocity-velocity block inverse. These augmented-Lagrangian-based methods result in iteration counts independent of discretisation or physical parameters including high-Reynolds flow as shown by Benzi and Olshanskii [56]. Such schemes were also successfully extended to the case of stabilised formulations by Benzi et al. [57]. Similar results were achieved by augmenting the velocity-velocity block with a grad-div term by Heister and Rapin [58].

Studies comparing those approaches revealed that augmented-Lagrangian-based solvers may outperform least-squares-commutator and pressure-convection-diffusion preconditioned ones, which themselves showed to be more robust than SIMPLE and related methods [59–61]. Such a statement is clearly too strict, though: in the time-dependent case, all of the methods showed at least partly satisfactory performance even for stretched grids and high Reynolds numbers. Some of the existing concepts can be straightforwardly extended to effectively account for highly varying rheological parameters, as e.g. in [62–64]. Applying an operator-splitting technique [45], we approximate the action of the Schur complement via operators defined on the pressure space, following Elman et al. [48] and incorporating ideas from [58,64] to account for the viscosity possibly spanning a few orders of magnitude.

Practical applications of fluid flows pose challenges even to high-performance computing clusters, highlighting that preconditioning is merely one (mandatory) part of fast solvers. Only when combined with adaptively chosen time steps to "follow the physics" and appropriate linearisations do such algorithms turn out to be really competitive. Schemes of this kind were presented for convection-diffusion-reaction equations by Gresho et al. [65], for the (Newtonian) incompressible Navier–Stokes equations by Kay et al. [66], and for buoyancy-driven flow by Elman et al. [67]. Moreover, an additional aspect of practical applications is the resulting size of spatial discretisations, easily reaching several million nodes. Taking arterial blood flow simulations as an example, it is obvious that some of these discretely described domains are bounded in an inherently lower-order fashion as a consequence of being reconstructed from voxelated data. Therefore, consistent stabilised formulations are of great relevance, but have not yet been combined with the aforementioned ideas.

Respecting these considerations, we present here a robust framework for non-Newtonian fluid flows combining low-order stabilised methods with a modern physics-based preconditioned linear solver, adaptive time integration schemes and linearisations of the fundamental equations. The remainder of this article is then organised as follows: in Section 2 we state the generalised Newtonian problem; in Section 3, our fully consistent stabilisation method is introduced; Section 4 deals with further aspects of discretisation in space and time including time step selection based on heuristic error measures. Thereafter, Section 5 focuses on the solution techniques applied including linearisation and construction of the preconditioner. Several representative numerical experiments are performed in Section 6, including stationary, periodic and transient problems in two and three dimensions.

2. Problem statement

We consider the balance of linear momentum and mass for an incompressible flow, described by the system

$$\rho\left[\frac{\partial \mathbf{u}}{\partial t} + (\nabla\mathbf{u})\mathbf{u}\right] - \nabla\cdot\mathbb{S} + \nabla p = \rho\mathbf{g} \qquad \text{in } \Omega\times(0,T]\,, \tag{1}$$

$$\nabla\cdot\mathbf{u} = 0 \qquad \text{in } \Omega\times(0,T]\,, \tag{2}$$

where the fluid velocity and pressure are denoted by $\mathbf{u}$ and p, respectively, the fluid density by ρ, the (given) external acceleration by $\mathbf{g}$, and the flow domain by $\Omega \in \mathbb{R}^d$ in the time interval from $t = 0$ to $t = T$. The viscous stress tensor $\mathbb{S}$ for a generalised Newtonian fluid is

$$\mathbb{S} = 2\mu\nabla^s\mathbf{u}\,, \tag{3}$$

in which the dynamic viscosity μ may further depend on $\nabla^s\mathbf{u} := ½[\nabla\mathbf{u} + (\nabla\mathbf{u})^\top]$. This nonlinear dependence is typically expressed in terms of the shear rate $\dot\gamma := \sqrt{½\nabla^s\mathbf{u} : \nabla^s\mathbf{u}}$. Taking blood or polymeric flows as an example, the well-established Carreau model [68]

$$\mu = \eta(\dot\gamma) = \mu_\infty + (\mu_0 - \mu_\infty)\left[1 + (\lambda\dot\gamma)^2\right]^{\frac{\zeta-1}{2}} \tag{4}$$

may be used, in which $\zeta \leq 1$ and the remaining parameters are positive material constants. The system (1)–(2) is supplemented by a generic initial condition for the velocity,

$$\mathbf{u}(\mathbf{x},0) = \mathbf{u}_0 \qquad \text{in } \Omega\,, \tag{5}$$

and is further subject to boundary conditions on the non-overlapping parts Γ_D and Γ_N of the boundary $\Gamma := \partial\Omega$:

$$\mathbf{u} = \mathbf{u}_D \qquad \text{on } \Gamma_D\times(0,T]\,, \tag{6}$$

$$(\mu\nabla\mathbf{u} - p\mathbb{I})\mathbf{n} = \tilde{\mathbf{t}} \qquad \text{on } \Gamma_N\times(0,T]\,, \tag{7}$$

where $\mathbf{n}$ is the unit outward normal vector, $\mathbb{I}$ is the $d\times d$ identity tensor, $\mathbf{u}_D$ is a prescribed velocity and $\tilde{\mathbf{t}}$ is a so-called pseudo-traction, which might either be set to zero (a "do-nothing" condition) or used for prescribing mean pressures over boundary segments [69].

3. Stabilised formulations

We now present the most classical residual-based stabilisation method for continuous equal-order elements, followed by a new global stabilisation approach. For concision, we will present the formulations for the stationary case and drop the body force. The time-dependent setting is considered in Section 4.

3.1. *Pressure-stabilised Petrov–Galerkin method*

The PSPG method is the most popular stabilisation approach for equal-order incompressible flow solvers. It consists of relaxing the divergence-free constraint (2) with an element-weighted residual of the momentum equation. Denoting by $\langle\cdot,\cdot\rangle$, $\langle\cdot,\cdot\rangle_{\Omega_e}$ and $\langle\cdot,\cdot\rangle_{\Gamma_N}$ the L^2 scalar products in Ω, Ω_e and Γ_N, respectively, the discrete variational formulation seeks $(\mathbf{u}_h, p_h) \in {X_h}^d \times X_h$, with $\mathbf{u}|_{\Gamma_D} = \mathbf{u}_D$, such that for all $(\mathbf{w}_h, q_h) \in {X_h}^d \times X_h$, with $\mathbf{w}|_{\Gamma_D} = \mathbf{0}$,

$$\langle \mathbf{w}_h, (\rho\nabla\mathbf{u}_h)\mathbf{u}_h\rangle + \left\langle \nabla^s\mathbf{w}_h, 2\mu(\nabla^s\mathbf{u}_h)\nabla^s\mathbf{u}_h\right\rangle - \langle \nabla\cdot\mathbf{w}_h, p_h\rangle = \langle \mathbf{w}_h, \mathbf{t}\rangle_{\Gamma_N}\,, \tag{8}$$

$$\langle q_h, \nabla\cdot\mathbf{u}_h\rangle + \sum_{e=1}^{N_e} \langle \delta_e\nabla q_h, \nabla p_h - \nabla\cdot\mathbb{S}_h + (\rho\nabla\mathbf{u}_h)\mathbf{u}_h\rangle_{\Omega_e} = 0\,, \tag{9}$$

where X_h is a continuous finite element space, $\mathbf{t} = (2\mu\nabla^s\mathbf{u} - p\mathbb{I})\mathbf{n}$ are normal boundary tractions, N_e is the number of elements Ω_e, and δ_e is a positive parameter dependent on the element size h_e. Let us also define $h = \max\{h_e\}$. For δ_e within an appropriate range, the system is stable for continuous equal-order pairs [70,71]. The term actually responsible for providing stability is the bilinear form $\langle\delta_e\nabla q_h, \nabla p_h\rangle$, which breaks the saddle-point structure of the problem and the LBB requirement [71]. As a matter of fact, the original stabilisation by Brezzi and Pitkäranta [30] uses only this pressure-pressure term. If that is the case, then what are the remaining terms in Eq. (9) for?

The idea behind including the whole residual is that the perturbation of the continuity equation be smaller where the solution is already accurate enough. While this should not affect asymptotic convergence rates, it is critical for the accuracy and quality of the approximations [72,73] – and here arises the issue of using PSPG with linear elements. In that case, and when using simplicial elements, the velocity gradient is piecewise constant and, consequently, so is the viscous stress. Hence:

$$\sum_{e=1}^{N_e} \langle \delta_e\nabla q_h, -\nabla\cdot\mathbb{S}_h\rangle_{\Omega_e} = \sum_{e=1}^{N_e} \langle \delta_e\nabla q_h, \mathbf{0}\rangle_{\Omega_e} = 0\,,$$

i.e., the viscous contribution to the residual is completely lost. This means that, regardless of how fine the mesh is, the added residual will never vanish. Of course, the formulation will still be consistent if $\delta_e \to 0$ as $h_e \to 0$, but an incomplete residual can spoil coarse grid accuracy and restrict the choice of the stabilisation parameter [41]. As a matter of fact, it can be shown that an "inviscid" stabilisation residual induces so-called spurious pressure boundary layers (*cf.* Section 6.1.1). These effects may be milder in convection-dominated flows, but for lower Reynolds numbers there can be considerable loss of accuracy. This is why it is common practice to use projection-based techniques such as the ones proposed by Jansen et al. [73] and Bochev and Gunzburger [74] to reconstruct the viscous term [37,75].

Another shortcoming of the standard formulation (8) is its natural boundary conditions (BCs) in terms of real tractions $\mathbf{t}$. Although very often done [2], setting $\mathbf{t} = \mathbf{0}$ on open boundaries leads to spurious velocity and pressure behaviour [76,77,69]. For fluids with constant viscosity, there is a simple remedy: using the Laplacian form of the stress divergence, that is,

$$\nabla\cdot\mathbb{S} = \mu\nabla\cdot\left(2\nabla^s\mathbf{u}\right) = \mu\left[\Delta\mathbf{u} + \nabla\left(\nabla\cdot\mathbf{u}\right)\right] = \mu\Delta\mathbf{u}\,.$$

Then, integration by parts will result in pseudo-tractions $\tilde{\mathbf{t}} = (\mu\nabla\mathbf{u})\mathbf{n} - p\mathbf{n}$ as natural BCs. The outflow boundary condition $\tilde{\mathbf{t}} \propto \mathbf{n}$ is not only satisfied by Poiseuille and Womersley flows, but also allows vortices to leave the computational domain with minimal upstream disturbance [76]. Therefore, pseudo-tractions are the preferred natural BCs for truncated outlets. When the viscosity is no longer constant, we can formulate the problem in a generalised Laplacian form by writing

$$\nabla\cdot\mathbb{S} = \nabla\cdot\left(2\mu\nabla^s\mathbf{u}\right) = \mu\left[\Delta\mathbf{u} + \nabla\left(\nabla\cdot\mathbf{u}\right)\right] + 2\nabla^s\mathbf{u}\nabla\mu = \mu\Delta\mathbf{u} + 2\nabla^s\mathbf{u}\nabla\mu\,, \tag{10}$$

and using integration by parts only for the term $\mu\Delta\mathbf{u}$ in order to yield the same pseudo-traction BCs as in the classical Newtonian case. For a test function $\mathbf{w}$ such that $\mathbf{w}|_{\Gamma_D} = \mathbf{0}$, we have

$$\begin{aligned}
-\left\langle \mathbf{w}, \mu\Delta\mathbf{u} + 2\nabla^s\mathbf{u}\nabla\mu - \nabla p\right\rangle &= \langle\nabla\mathbf{u}, \nabla(\mu\mathbf{w})\rangle - \langle p, \nabla\cdot\mathbf{w}\rangle - \langle \mathbf{w}, (\mu\nabla\mathbf{u})\mathbf{n} - p\mathbf{n}\rangle_{\Gamma} - \left\langle \mathbf{w}, 2\nabla^s\mathbf{u}\nabla\mu\right\rangle \\
&= \langle\nabla\mathbf{u}, \mu\nabla\mathbf{w} + \mathbf{w}\otimes\nabla\mu\rangle - \langle p, \nabla\cdot\mathbf{w}\rangle - \langle \mathbf{w}, (\mu\nabla\mathbf{u})\mathbf{n} - p\mathbf{n}\rangle_{\Gamma_N} - \left\langle \mathbf{w}, 2\nabla^s\mathbf{u}\nabla\mu\right\rangle \\
&= \langle\nabla\mathbf{w}, \mu\nabla\mathbf{u}\rangle + \langle \mathbf{w}, \nabla\mathbf{u}\nabla\mu\rangle - \langle p, \nabla\cdot\mathbf{w}\rangle - \left\langle \mathbf{w}, \tilde{\mathbf{t}}\right\rangle_{\Gamma_N} - \left\langle \mathbf{w}, 2\nabla^s\mathbf{u}\nabla\mu\right\rangle \\
&= \langle\nabla\mathbf{w}, \mu\nabla\mathbf{u}\rangle - \left\langle \mathbf{w}, (\nabla\mathbf{u})^\top\nabla\mu\right\rangle - \langle \nabla\cdot\mathbf{w}, p\rangle - \left\langle \mathbf{w}, \tilde{\mathbf{t}}\right\rangle_{\Gamma_N}.
\end{aligned}$$

The weak form of the discrete momentum equation then becomes

$$\langle \mathbf{w}_h, (\rho\nabla\mathbf{u}_h)\mathbf{u}_h\rangle + \langle \nabla\mathbf{w}_h, \mu_h\nabla\mathbf{u}_h\rangle - \left\langle \mathbf{w}_h, (\nabla\mathbf{u}_h)^\top \nabla\mu_h \right\rangle - \langle \nabla\cdot\mathbf{w}_h, p_h\rangle = \langle \mathbf{w}_h, \tilde{\mathbf{t}}\rangle_{\Gamma_N}\,, \tag{11}$$

where μ_h is a continuous L^2 projection of the viscosity field, as discussed later on.

3.2. *A fully consistent residual-based stabilisation*

Regardless of whether using the stress-divergence or generalised Laplacian form of the momentum equation, devising a consistent stabilisation method for equal-order pairs is critical for an accurate approximation. In what follows, we introduce a residual-based formulation which preserves the full residual even for lowest-order pairs.

3.2.1. *Strong form*

Before introducing our stabilisation method, we construct an equivalent PDE system to replace the classical momentum-mass system (1)–(2). The proposed boundary value problem reads:

$$(\rho\nabla\mathbf{u})\mathbf{u} - 2\nabla^s\mathbf{u}\nabla\mu - \mu\left[\Delta\mathbf{u} + \chi\nabla(\nabla\cdot\mathbf{u})\right] + \nabla p = \mathbf{0} \qquad \text{in } \Omega\,, \tag{12}$$

$$-\Delta p = \nabla\cdot\left[(\rho\nabla\mathbf{u})\mathbf{u} - 2\nabla^s\mathbf{u}\nabla\mu\right] + \left[\nabla\times(\nabla\times\mathbf{u})\right]\cdot\nabla\mu - \beta\nabla\cdot\mathbf{u} \qquad \text{in } \Omega\,, \tag{13}$$

$$\frac{\partial p}{\partial n} = \mathbf{n}\cdot\left[2\nabla^s\mathbf{u}\nabla\mu - (\rho\nabla\mathbf{u})\mathbf{u} - \mu\nabla\times(\nabla\times\mathbf{u})\right] \qquad \text{on } \Gamma\,, \tag{14}$$

where β is a positive parameter to be defined later, and χ is a factor used simply to switch between the stress-divergence ($\chi=1$) and generalised Laplacian ($\chi=0$) formulations. The momentum equation BCs were omitted because they play no role in the following discussion.

As shall become clear in the remainder of this section, the main advantage of system (12)–(14) is allowing piecewise linear approximation for velocity and pressure by rewriting (13) with only first-order terms. Yet, we first need to show that, for sufficiently regular p and $\mathbf{u}$, the new system is equivalent to the classical one ((1)–(2)).

Lemma 3.1. *For* $\mathbf{u}$ *and* p *sufficiently regular, Eqs.* (1) *and* (2) *imply* (12)–(14).

Proof. The equivalence between Eqs. (12) and (1) for a divergence-free flow has already been shown in Eq. (10). Proving that Eqs. (1) and (2) lead to Eq. (14) is also straightforward: we dot Eq. (1) by $\mathbf{n}$ and use Eq. (2) to write

$$\Delta\mathbf{u} \equiv \nabla(\nabla\cdot\mathbf{u}) - \nabla\times(\nabla\times\mathbf{u}) = -\nabla\times(\nabla\times\mathbf{u})\,. \tag{15}$$

To derive Eq. (13), we apply again relation (15) to Eq. (1), and take minus the divergence of both sides, resulting in

$$-\Delta p = \nabla\cdot\left[(\rho\nabla\mathbf{u})\mathbf{u} - 2\nabla^s\mathbf{u}\nabla\mu - \mu\nabla\times(\nabla\times\mathbf{u})\right], \tag{16}$$

but

$$\nabla\cdot\left[\mu\nabla\times(\nabla\times\mathbf{u})\right] = \left[\nabla\times(\nabla\times\mathbf{u})\right]\cdot\nabla\mu + \mu\nabla\cdot\left[\nabla\times(\nabla\times\mathbf{u})\right] = \left[\nabla\times(\nabla\times\mathbf{u})\right]\cdot\nabla\mu\,,$$

since the divergence of a curl is zero. Furthermore, due to Eq. (2), we can arbitrarily add the term $-\beta\nabla\cdot\mathbf{u}$ to the right-hand side of (16), which completes the proof. For a more general form of the so-called pressure Poisson equation (PPE) (16) allowing compressibility, see [78]. □

Lemma 3.2. *The system* (12)–(14) *implies Eqs.* (1) *and* (2).

Proof. The first step is to apply the divergence operator to Eq. (12), leading to

$$\begin{aligned}\Delta p &= \nabla\cdot\left[\mu\Delta\mathbf{u} + \chi\mu\nabla(\nabla\cdot\mathbf{u}) + 2\nabla^s\mathbf{u}\nabla\mu - (\rho\nabla\mathbf{u})\mathbf{u}\right]\\ &= \nabla\cdot\left[2\nabla^s\mathbf{u}\nabla\mu - (\rho\nabla\mathbf{u})\mathbf{u}\right] + \nabla\mu\cdot\left[\Delta\mathbf{u} + \chi\nabla(\nabla\cdot\mathbf{u})\right] + \mu\left[\nabla\cdot(\Delta\mathbf{u}) + \chi\Delta(\nabla\cdot\mathbf{u})\right],\end{aligned} \tag{17}$$

which when added to Eq. (13) gives

$$\begin{aligned}0 &= \left[\Delta\mathbf{u} + \nabla\times(\nabla\times\mathbf{u}) + \chi\nabla(\nabla\cdot\mathbf{u})\right]\cdot\nabla\mu + \mu\left[\nabla\cdot(\Delta\mathbf{u}) + \chi\Delta(\nabla\cdot\mathbf{u})\right] - \beta\nabla\cdot\mathbf{u}\\ &\equiv (\chi+1)\,\mu\Delta\,(\nabla\cdot\mathbf{u}) + (\chi+1)\,\nabla\mu\cdot\left[\nabla(\nabla\cdot\mathbf{u})\right] - \beta\nabla\cdot\mathbf{u}\\ &\equiv -\beta\nabla\cdot\mathbf{u} + (\chi+1)\,\nabla\cdot\left[\mu\nabla(\nabla\cdot\mathbf{u})\right],\end{aligned} \tag{18}$$

since $\Delta\mathbf{u} + \nabla\times(\nabla\times\mathbf{u}) \equiv \nabla(\nabla\cdot\mathbf{u})$ and $\nabla\cdot(\Delta\mathbf{u}) \equiv \Delta\,(\nabla\cdot\mathbf{u})$.

Therefore, introducing $\phi := \nabla \cdot \mathbf{u}$, we have the diffusion-reaction equation

$$-\nabla \cdot (\mu \nabla \phi) + \frac{\beta}{\chi + 1}\phi = 0 . \tag{19}$$

We can obtain Neumann BCs for this equation by dotting Eq. (12) with $\mathbf{n}$ and substracting from (14), which gives

$$\begin{aligned} 0 &= \mu \mathbf{n} \cdot [\chi \nabla(\nabla \cdot \mathbf{u}) + \Delta \mathbf{u} + \nabla \times (\nabla \times \mathbf{u})] \\ &= (\chi + 1)\, \mu \mathbf{n} \cdot [\nabla (\nabla \cdot \mathbf{u})] \\ &= (\chi + 1)\, \mu \frac{\partial \phi}{\partial n} , \end{aligned} \tag{20}$$

i.e., we have zero Neumann data for ϕ on Γ. The solution of Eq. (19) is thus $\phi \equiv 0$, that is, $\nabla \cdot \mathbf{u} = 0$ in Ω, as we wanted. Now that we have proved incompressibility, the equivalence between Eqs. (12) and (1) is straightforward. □

Remark. We have kept the factor χ in the derivations to show that our modified system is well posed regardless of whether the stress-divergence or (generalised) Laplacian formulation is employed. Since χ does not feature in the pressure Poisson problem (13)–(14), the stabilisation method presented herein is independent of the form selected for the momentum equation. That being said, we shall henceforth stick with the generalised Laplacian description in our algorithms and numerical examples.

3.2.2. Variational formulation

The modified pressure Poisson problem (13)–(14) is the basis for deriving our stabilised formulation. The first step is to use Green's first formula on the PPE (13) to yield

$$\langle \nabla q, \nabla p \rangle - \left\langle q, \frac{\partial p}{\partial n} \right\rangle_\Gamma = \big\langle q, \nabla \cdot \big[(\rho \nabla \mathbf{u})\mathbf{u} - 2\nabla^s \mathbf{u} \nabla \mu\big]\big\rangle + \langle q, [\nabla \times (\nabla \times \mathbf{u})] \cdot \nabla \mu - \beta \nabla \cdot \mathbf{u} \rangle .$$

Integrating by parts the first term on the right-hand side and enforcing the Neumann BC (14) gives

$$\langle q, \beta \nabla \cdot \mathbf{u} \rangle + \langle \nabla q, \nabla p + (\rho \nabla \mathbf{u})\mathbf{u} \rangle = \left\langle \nabla q, \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^\top\right] \nabla \mu \right\rangle + \langle q, [\nabla \times (\nabla \times \mathbf{u})] \cdot \nabla \mu \rangle - \langle q\mathbf{n}, \mu \nabla \times (\nabla \times \mathbf{u}) \rangle_\Gamma , \tag{21}$$

but

$$\begin{aligned} \langle q\mathbf{n}, \mu \nabla \times (\nabla \times \mathbf{u}) \rangle_\Gamma &= \langle q, \mu \nabla \cdot [\nabla \times (\nabla \times \mathbf{u})] \rangle + \langle \nabla (q\mu), \nabla \times (\nabla \times \mathbf{u}) \rangle \\ &= \langle \nabla q, \mu \nabla \times (\nabla \times \mathbf{u}) \rangle + \langle q, \nabla \mu \cdot [\nabla \times (\nabla \times \mathbf{u})] \rangle . \end{aligned}$$

Therefore, the right-hand side of Eq. (21) reduces to

$$\left\langle \nabla q, \left[\nabla \mathbf{u} + (\nabla \mathbf{u})^\top\right] \nabla \mu \right\rangle - \langle \nabla q, \mu \nabla \times (\nabla \times \mathbf{u}) \rangle .$$

As in the PSPG formulation, we still have a second-order term that cannot be approximated by low-order finite element spaces. It is only possible to preserve the full viscous residual if we can rewrite this term using only first-order derivatives. We can in fact use integration by parts once again to write

$$\langle \mu \nabla q, \nabla \times (\nabla \times \mathbf{u}) \rangle = \langle \nabla q \times \mathbf{n}, \mu \nabla \times \mathbf{u} \rangle_\Gamma + \langle \nabla \times (\mu \nabla q) \, , \nabla \times \mathbf{u} \rangle ,$$

and

$$\begin{aligned} \langle \nabla \times (\mu \nabla q) \, , \nabla \times \mathbf{u} \rangle &= \langle \mu \nabla \times (\nabla q) + \nabla \mu \times \nabla q, \nabla \times \mathbf{u} \rangle = \langle \nabla \mu \times \nabla q, \nabla \times \mathbf{u} \rangle \\ &\equiv \langle \nabla q, (\nabla \times \mathbf{u}) \times \nabla \mu \rangle \equiv \left\langle \nabla q, \left[\nabla \mathbf{u} - (\nabla \mathbf{u})^\top\right] \nabla \mu \right\rangle . \end{aligned}$$

Thus, some terms cancel out and the weak form simplifies to

$$\langle q, \beta \nabla \cdot \mathbf{u} \rangle + \left\langle \nabla q, \nabla p + (\rho \nabla \mathbf{u})\mathbf{u} - 2(\nabla \mathbf{u})^\top \nabla \mu \right\rangle + \langle \nabla q \times \mathbf{n}, \mu \nabla \times \mathbf{u} \rangle_\Gamma = 0 ,$$

which is now free of second-order derivatives.

There is one last issue to be addressed. In this weak form, as well as in the generalised Laplacian form of the momentum equation (see Eq. (11)), the term $\nabla \mu$ is present. Since the viscosity in general depends on $\nabla^s \mathbf{u}$, computing $\nabla \mu$ in the variational formulation would increase the regularity requirements on the velocity interpolant and therefore prohibit the use of standard Lagrangian finite elements. This can be avoided by projecting the viscosity onto a continuous space, or, in other words, enforcing the rheological law weakly. Our final variational formulation is then to find $(\mathbf{u}, p, \mu) \in X \times Y \times Z$, with $\mathbf{u}|_{\Gamma_D} = \mathbf{u}_D$, such that for all $(\mathbf{w}, q, r) \in \tilde{X} \times \tilde{Y} \times \tilde{Z}$, with $\mathbf{w}|_{\Gamma_D} = \mathbf{0}$,

$$\langle \mathbf{w}, (\rho \nabla \mathbf{u})\mathbf{u} \rangle + \langle \nabla \mathbf{w}, \mu \nabla \mathbf{u} \rangle - \left\langle \mathbf{w}, (\nabla \mathbf{u})^\top \nabla \mu \right\rangle - \langle \nabla \cdot \mathbf{w}, p \rangle = \langle \mathbf{w}, \tilde{\mathbf{t}} \rangle_{\Gamma_N} , \tag{22}$$

$$\langle q, \beta \nabla \cdot \mathbf{u} \rangle + \left\langle \nabla q, \nabla p + (\rho \nabla \mathbf{u})\mathbf{u} - 2(\nabla \mathbf{u})^\top \nabla \mu \right\rangle + \langle \nabla q \times \mathbf{n}, \mu \nabla \times \mathbf{u} \rangle_\Gamma = 0 , \tag{23}$$

$$\left\langle r, \mu - \eta \left(\dot{\gamma} \left(\nabla^s \mathbf{u} \right) \right) \right\rangle = 0 , \tag{24}$$

with continuous test and trial spaces. We shall refer to this formulation as *boundary vorticity stabilisation* (BVS), as the key to a consistent residual is the first-order boundary term proportional to the vorticity $\nabla \times \mathbf{u}$. For a homogeneous Newtonian fluid, all terms proportional to $\nabla \mu$ vanish, recovering the method recently presented in [41]. Notice that another advantage of weakly coupling μ to $\mathbf{u}$ is that the resulting regularity requirements allow the stabilisation term to be computed globally, instead of the usual restriction to element interiors. Local mesh size effects can be handled by the parameter β in the divergence term, as discussed next.

3.2.3. *The stabilisation parameter*

A quick comparison between our stabilised form (23) and the relaxed continuity equation (9) in the PSPG method leads to the simple choice $\beta = 1/\delta_e$, with δ_e being the local PSPG stabilisation parameter. Different definitions exist for δ_e, a popular one being

$$\delta_e = \left[\left(\frac{\alpha_1 \mu_h}{{h_e}^2} \right)^2 + \left(\frac{\alpha_2 \rho |\mathbf{u}_h|}{h_e} \right)^2 + \left(\frac{\alpha_3 \rho}{\Delta t} \right)^2 \right]^{-1/2} , \tag{25}$$

where Δt is the time step size used for the temporal discretisation, $|\mathbf{u}_h|$ denotes the Euclidian norm of $\mathbf{u}_h$, and α_1, α_2, α_3 are stabilisation factors. A common choice is $\alpha_1 = 2\alpha_2 = 2\alpha_3 = 4$ [35]. We use $\alpha_2 = 0$ in our examples, since our focus are diffusion-dominated flows where convective stabilisation is not needed. For linear elements, $\alpha_1 = 12$ is also often recommended [70], and in stationary diffusive flows one can use simply [71]

$$\beta|_{\Omega_e} = \frac{1}{\delta_e} = \frac{\mu_h}{\alpha {h_e}^2} , \tag{26}$$

with α being a positive parameter. We will use this expression in Section 6.1 to assess the influence of the stabilisation parameter on the performance of our method. In this case, our stabilised continuity equation reads

$$\left\langle \nabla q_h, \nabla p_h + (\rho \nabla \mathbf{u}_h)\mathbf{u}_h - 2(\nabla \mathbf{u}_h)^\top \nabla \mu_h \right\rangle + \langle \nabla q_h \times \mathbf{n}, \mu_h \nabla \times \mathbf{u}_h \rangle_\Gamma + \frac{1}{\alpha} \sum_{e=1}^{N_e} {h_e}^{-2} \langle q_h, \mu_h \nabla \cdot \mathbf{u}_h \rangle_{\Omega_e} = 0 ,$$

that is, the stabilisation term is computed globally, whereas the divergence term is computed in an element-weighted manner, conversely to classical residual-based formulations. Having laid out the basic concept for our new stabilisation approach, we proceed with discretisation aspects in the more general time dependent case.

4. Discretisation aspects

The discretisation in time is done with the help of the generalised θ-scheme as a basic ingredient. So, let us divide the time interval $(0, T]$ into N^t steps, $\{t_i\}_{i=1}^{N^t}$, and abbreviate $\mathbf{u}^n = \mathbf{u}(\mathbf{x}, t_n)$. For the current time step Δt_n from t_n to t_{n+1}, i.e., $\Delta t_n = t_{n+1} - t_n$, the semi-discrete problem is the following: Given $(\mathbf{u}^n, p^n)$ and boundary data $\mathbf{u}_D^{n+1}$ and $\tilde{\mathbf{t}}^{n+1}$, compute the solution $(\mathbf{u}^{n+1}, p^{n+1})$ at time t_{n+1} via

$$\frac{\rho}{\Delta t_n} \left(\mathbf{u}^{n+1} - \mathbf{u}^n \right) + \theta \mathbf{f}^{n+1} + \theta' \mathbf{f}^n = 0 \qquad \text{in } \Omega , \tag{27}$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0 \qquad \text{in } \Omega , \tag{28}$$

$$\mu_\infty + (\mu_0 - \mu_\infty) \left(1 + 0.5 \lambda^2 \nabla^s \mathbf{u}^{n+1} : \nabla^s \mathbf{u}^{n+1} \right)^{\frac{\zeta - 1}{2}} = \mu^{n+1} \qquad \text{in } \Omega , \tag{29}$$

$$\mathbf{u}^{n+1} = \mathbf{u}_D^{n+1} \qquad \text{on } \Gamma_D , \tag{30}$$

$$\left(\mu^{n+1} \nabla \mathbf{u}^{n+1} - p^{n+1} \mathbb{I} \right) \mathbf{n} = \tilde{\mathbf{t}}^{n+1} \qquad \text{on } \Gamma_N , \tag{31}$$

with the shorthand notation $\mathbf{f} := (\rho \nabla \mathbf{u})\mathbf{u} - \rho \mathbf{g} + \nabla p - 2\nabla^s \mathbf{u} \nabla \mu - \mu \Delta \mathbf{u}$. As can be seen from Eq. (29), the rheological law may be exchanged effortlessly, which is why we limit the discussion to the Carreau model (4). Fixing the parameters θ and θ' in (27), one can choose a specific time integration scheme, e.g., $\theta = 1$ and $\theta' = 0$ corresponding to the implicit Euler scheme or $\theta = \theta' = 1/2$ for the Crank–Nicolson (CN) scheme. Note, however, that the pressure is integrated consistently in time applying a so-called pressure-corrected θ-scheme [79] and as a consequence, for $\theta \neq 1$, the pressure p^n from the previous time step or an initial condition for the pressure $p(\mathbf{x}, 0) = p_0$ is needed. The simplest and most usual way to dodge this complication is to use a (small) initial implicit Euler step. Moreover, the divergence-free condition (28) is enforced on the discrete velocity $\mathbf{u}^{n+1}$ at every point in time. In doing so, errors in mass conservation do not accumulate over time, while (28) is equivalent to a time-integrated form in the case of an initially divergence-free velocity field [80,81]. Similar reasoning lies behind the choice of (29), enforcing the time-independent rheological law at every point in time.

Rothe's method is applied to discretise first in time and then in space based on the variational formulation presented in Section 3. The fully discrete problem is to find $(\mathbf{u}_h^{n+1}, p_h^{n+1}, \mu_h^{n+1}) \in X_h \times Y_h \times Z_h$, with $\mathbf{u}_h^{n+1}|_{\Gamma_D} = \mathbf{u}_D^{n+1}$, such that for all $(\mathbf{w}_h, q_h, r_h) \in \tilde{X}_h \times \tilde{Y}_h \times \tilde{Z}_h$, with $\mathbf{w}_h|_{\Gamma_D} = \mathbf{0}$,

$$\begin{aligned}
&\left\langle \mathbf{w}_h, \rho \mathbf{u}_h^{n+1} + \tilde{\theta}\left[(\rho \nabla \mathbf{u}_h^{n+1})\mathbf{v}_h^{n+1} - (\nabla \mathbf{u}_h^{n+1})^\top \nabla \mu_h^{n+1}\right]\right\rangle + \left\langle \nabla \mathbf{w}_h, \tilde{\theta}\mu_h^{n+1} \nabla \mathbf{u}_h^{n+1}\right\rangle - \left\langle \nabla \cdot \mathbf{w}_h, \tilde{\theta} p_h^{n+1}\right\rangle = \\
&\left\langle \mathbf{w}_h, \rho(\mathbf{u}_h^n + \tilde{\theta}\mathbf{g}^{n+1}) - \tilde{\theta}'\left[(\rho \nabla \mathbf{u}_h^n)\mathbf{u}_h^n - \rho \mathbf{g}^n - (\nabla \mathbf{u}_h^n)^\top \nabla \mu_h^n\right]\right\rangle - \left\langle \nabla \mathbf{w}_h, \tilde{\theta}' \mu_h^n \nabla \mathbf{u}_h^n\right\rangle + \left\langle \nabla \cdot \mathbf{w}_h, \tilde{\theta}' p_h^n\right\rangle \\
&\quad + \langle \mathbf{w}_h, \tilde{\theta}\tilde{\mathbf{t}}^{n+1} + \tilde{\theta}'\tilde{\mathbf{t}}^n\rangle_{\Gamma_N}\,, \qquad (32)
\end{aligned}$$

$$\left\langle q_h, \nabla \cdot \mathbf{u}_h^{n+1}\right\rangle = 0\,, \qquad (33)$$

$$\left\langle r_h, \mu_h^{n+1} - (\mu_0 - \mu_\infty)\left(1 + 0.5\lambda^2 \nabla^s \mathbf{v}_h^{n+1} : \nabla^s \mathbf{v}_h^{n+1}\right)^{\frac{\zeta-1}{2}}\right\rangle = \langle r_h, \mu_\infty\rangle\,, \qquad (34)$$

with $\tilde{\theta} = \theta \Delta t_n$ and $\tilde{\theta}' = \theta' \Delta t_n$ and the (convective) velocity $\mathbf{v}_h^{n+1}$. Taking $\mathbf{v}_h^{n+1}$ equal to $\mathbf{u}_h^{n+1}$ introduces nonlinearities to the weak form of the balance of linear momentum and couples the velocity and viscosity solution components in a nonlinear fashion through the diffusion terms. As a consequence, nonlinear solvers such as a fixed-point or Newton's method have to be applied, resulting in several iterations per time step in each of which a 3 × 3 block-system has to be appropriately preconditioned and solved. Alternatively, we extrapolate the velocity linearly, which gives

$$\mathbf{v}_h^{n+1} = \left(1 + \frac{\Delta t_n}{\Delta t_{n-1}}\right)\mathbf{u}_h^n - \frac{\Delta t_n}{\Delta t_{n-1}}\mathbf{u}_h^{n-1}. \qquad (35)$$

This is much simpler, since it decouples the viscosity from the overall system and linearises the convective term. Thus, only an L^2 projection for the viscosity, followed by a velocity-pressure system solve are needed per time step. Similar linearisations have been proposed for Newtonian fluids [82,42,83] and buoyancy-driven flow problems [67], where they showed to be a worthwhile compromise between speed, accuracy and stability.

4.1. Equal-order stabilisation

The discrete function spaces are still undefined, since we use either (i) isoparametric Taylor–Hood elements, featuring quadratic velocity interpolation and linear pressure functions, or (ii) linear elements for velocity and pressure together with a modification of (33). For the latter choice, the time dependent counterpart of the PSPG method (8)–(9) replaces the continuity equation (33) in system (32)–(34) by

$$\left\langle q_h, \nabla \cdot \mathbf{u}_h^{n+1}\right\rangle + \sum_{e=1}^{N_e}\left\langle \delta_e \nabla q_h, \rho(\mathbf{u}_h^{n+1} - \mathbf{u}_h^n) + \tilde{\theta}\mathbf{f}_h^{n+1} + \tilde{\theta}'\mathbf{f}_h^n\right\rangle_{\Omega_e} = 0\,, \qquad (36)$$

with $\mathbf{f}_h = \rho\,[(\nabla \mathbf{u}_h)\mathbf{v}_h - \mathbf{g}] + \nabla p_h - 2\nabla^s \mathbf{u}_h \nabla \mu_h - \mu_h \Delta \mathbf{u}_h$ being the discrete counterpart of $\mathbf{f}$ with possibly extrapolated convective velocity $\mathbf{v}_h^{n+1}$ (but of course $\mathbf{v}_h^n = \mathbf{u}_h^n$). Alternatively, applying our BVS formulation, we replace the continuity equation (33) in (32)–(34) by

$$\left\langle \nabla q_h, \rho(\mathbf{u}_h^{n+1} - \mathbf{u}_h^n) + \tilde{\theta}\tilde{\mathbf{f}}_h^{n+1} + \tilde{\theta}'\tilde{\mathbf{f}}_h^n\right\rangle + \langle \nabla q_h \times \mathbf{n}, \tilde{\theta}\mu_h^{n+1} \nabla \times \mathbf{u}_h^{n+1} + \tilde{\theta}'\mu_h^n \nabla \times \mathbf{u}_h^n\rangle_\Gamma + \sum_{e=1}^{N_e}\left\langle {\delta_e}^{-1} q_h, \nabla \cdot \mathbf{u}_h^{n+1}\right\rangle_{\Omega_e} = 0\,, \qquad (37)$$

with $\tilde{\mathbf{f}}_h = (\rho \nabla \mathbf{u}_h)\mathbf{v}_h - \rho \mathbf{g} + \nabla p_h - 2(\nabla \mathbf{u}_h)^\top \nabla \mu_h$ free of second-order derivatives. Comparing (36) and (37) from an algorithmic point of view, we see that either second-order derivatives (PSPG) or domain boundary integration (BVS) must be employed – then, in fact, one ends up with very similar element integration routines.

Mass conservation is further improved by grad-div stabilisation [84]. This penalty term is commonly used for LBB-stable and especially for stabilised formulations in order to enhance conservation of mass, simply adding [71]

$$+\sum_{e=1}^{N_e}\left\langle \tilde{\gamma}_e \nabla\cdot\mathbf{w}_h,\nabla\cdot\mathbf{u}_h^{n+1}\right\rangle_{\Omega_e}\ ,\quad \frac{\tilde{\gamma}_e}{\Delta t_n}=\gamma_e=\begin{cases}0.1 & \text{for } Q_2/Q_1 \text{ pairs,}\\ 0.5\,h_e & \text{for } Q_1/Q_1 \text{ pairs}\end{cases} \tag{38}$$

to the momentum equation (32).

4.2. Time step selection

4.2.1. Predictor-corrector approach

The CN time integration scheme is a particularly popular method, as it is easy to implement, second-order accurate and A-stable [85]. Its non-dissipative nature is an attractive feature when pure convection or convection-dominated problems are considered. Unfortunately though, applying the CN scheme with (large) fixed time steps can be problematic as thoroughly investigated in literature, including works critically comparing CN to other available time-stepping algorithms or operator-splitting methods [86,43]. Various remedies have been presented to consistently introduce slight numerical damping to account for, e.g., rough initial or boundary data. Strategies like introducing intermediate implicit Euler ($\theta = 1$) steps [87,88], periodic averaging [89,90,65,66] or shifting the θ parameter slightly to the implicit side [85] are among the methodologies resorted to. Here, we settle for a variant of the Rannacher time-stepping (R), introducing backward Euler steps in-between CN steps. The discussion regarding this choice is continued in Section 6, where we compare different approaches.

To control the step size, a predictor-corrector approach (see e.g. [91,65–67]) is taken, controlling the time step of the CN method by comparing it to an explicit Adams–Bashforth method of second order (AB2), which computes the velocity as

$$\tilde{\mathbf{u}}^{n+1}=\mathbf{u}^n+\frac{\Delta t_n}{2}\left[\left(2+\frac{\Delta t_n}{\Delta t_{n-1}}\right)\dot{\mathbf{u}}^n-\frac{\Delta t_n}{\Delta t_{n-1}}\dot{\mathbf{u}}^{n-1}\right], \tag{39}$$

approximating the time derivatives using solutions $\mathbf{u}_h$ obtained with the implicit CN scheme in the previous step by

$$\dot{\mathbf{u}}_h^n=\frac{2}{\Delta t_n}(\mathbf{u}_h^{n+1}-\mathbf{u}_h^n)-\dot{\mathbf{u}}_h^{n-1}. \tag{40}$$

Based on the solutions $\tilde{\mathbf{u}}_h^{n+1}$ and $\mathbf{u}_h^{n+1}$, the new time step size is computed as

$$\Delta t_{n+1}=\xi\,\Delta t_n\left(\frac{\epsilon_v}{e_{n+1}}\right)^{\frac{1}{3}}\quad\text{with}\quad e_{n+1}=\left|\frac{\mathbf{u}_h^{n+1}-\tilde{\mathbf{u}}_h^{n+1}}{3(1+\Delta t_{n-1}/\Delta t_n)}\right|, \tag{41}$$

where ϵ_v is some specified tolerance on the L^2 norm of the truncation error in the velocity components and the safety factor $\xi\in(0,1]$ is introduced.

Two issues are encountered when implementing the AB2 scheme. First, it is not self-starting, which is why an initial implicit Euler step of small size is used together with $\dot{\mathbf{u}}_h^{n-1}\approx\dot{\mathbf{u}}_h^n$ or $\dot{\mathbf{u}}_h^{n-1}\approx\mathbf{0}$ if the boundary conditions are ramped up in a consistent way [45]. Second, the explicit AB2 integrator is prone to ringing and therefore stabilised by averaging the solution vectors

$$\dot{\mathbf{u}}_h^{n-1}=\frac{1}{2}\left(\dot{\mathbf{u}}_h^n+\dot{\mathbf{u}}_h^{n-1}\right)\quad\text{and}\quad\dot{\mathbf{u}}_h^n=\frac{1}{\Delta t_n}(\mathbf{u}_h^{n+1}-\mathbf{u}_h^n), \tag{42}$$

in every N^*-th step. This averaging step and the Euler step within the Rannacher scheme are executed at the same time. Similar stabilised predictor-corrector schemes were proposed by [65–67], where an averaging such as (42) is also applied to the velocity components used in the implicit scheme. In order to treat the pressure in the implicit scheme, this averaging step is replaced by an implicit Euler step. N^* is a fixed value being a compromise between accuracy and stability. Choosing N^* too low spoils accuracy, while setting it too high causes the integrator to stall at a certain step size, even if larger time steps might not violate the target tolerance.

4.2.2. Fractional step θ-scheme

The stabilised predictor-corrector approach is compared to a pressure-corrected fractional step θ-scheme [79] (FS), which is second-order accurate as well, but *strongly* A-stable. Having an implementation of the generalised θ-scheme available, it can be straightforwardly extended to the FS scheme by simply grouping three steps of the generalised θ-scheme to one macro step from t_m to t_{m+1} of size Δt_m and setting the time integration parameters and step lengths according to Table 1, with $\kappa=1-1/\sqrt{2}$, $\tilde{\kappa}=1-2\kappa$, $\alpha=\tilde{\kappa}/(1-\kappa)$ and $\beta=1-\alpha$.

Adaptive time step control is directly enabled by interpreting the FS scheme as a Runge–Kutta method as done by Rang [79], leading to a representation with the Butcher tableau

$$\begin{array}{c|c}\mathbf{c} & \mathbf{A}\\ \hline & \mathbf{b}^\top\end{array}\;=\;\begin{array}{c|cccc}0 & 0 & 0 & 0 & 0\\ \kappa & \kappa\beta & \kappa\alpha & 0 & 0\\ \kappa+\tilde{\kappa} & \kappa\beta & (\kappa+\tilde{\kappa})\alpha & \tilde{\kappa}\beta & 0\\ 1 & \kappa\beta & (\kappa+\tilde{\kappa})\alpha & (\kappa+\tilde{\kappa})\beta & \kappa\alpha\\ \hline & \kappa\beta & (\kappa+\tilde{\kappa})\alpha & (\kappa+\tilde{\kappa})\beta & \kappa\alpha\end{array} \tag{43}$$

Table 1
Time integration parameters and step lengths for the FS scheme.

Step	θ	θ'	Δt_n	t_n	t_{n+1}
1	α	β	$\kappa\Delta t_m$	t_m	$t_m+\kappa\Delta t_m$
2	β	α	$\tilde{\kappa}\Delta t_m$	$t_m+\kappa\Delta t_m$	$t_m+(1-\kappa)\Delta t_m$
3	α	β	$\kappa\Delta t_m$	$t_m+(1-\kappa)\Delta t_m$	t_{m+1}

with an embedded method of order 1 with coefficient vector $\tilde{\mathbf{b}}$ as given by [92]

$$\tilde{\mathbf{b}} = \begin{pmatrix} 0.11785113033497070959 \\ 0.49509379160690495120 \\ 0.29636243203812433921 \\ 0.09069264621404818692 \end{pmatrix}. \tag{44}$$

Already put as an open problem in [79], bundling three steps of the generalised θ-method to one macro step, it is not immediately clear how the embedded scheme can be utilised given intermediate solutions. However, one can recover the coefficient vectors necessary to do so. This is particularly convenient, since the algorithm might then be based on an already existing code using a simple generalised θ-method for time stepping. To make our point clear (and for the convenience of the reader), we briefly introduce Runge–Kutta methods and the connection of the FS scheme to the generalised θ-scheme, while referring to [79,92] for a more detailed discussion.

Let us consider the system

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho}\mathbf{f}(t,\mathbf{u},p) = \mathbf{g} - (\nabla\mathbf{u})\mathbf{u} - \frac{1}{\rho}\left(\nabla p - 2\nabla^s\mathbf{u}\nabla\mu - \mu\Delta\mathbf{u}\right), \tag{45}$$

$$0 = g(t,\mathbf{u},p) = \nabla\cdot\mathbf{u} \tag{46}$$

and assume for now consistent initial conditions $\mathbf{u}(\mathbf{x},0)=\mathbf{u}_0$ and $p(\mathbf{x},0)=p_0$ given. Then, we define an s-stage Runge–Kutta method via the updates

$$\mathbf{u}^{m+1} = \mathbf{u}^m + \Delta t_m \sum_{i=1}^{s} b_i\mathbf{k}^i\,, \quad p^{m+1} = p^m + \Delta t_m \sum_{i=1}^{s} b_i l^i \tag{47}$$

and auxiliary equations for $i=1,\ldots,s$

$$\mathbf{k}^i = -\frac{1}{\rho}\mathbf{f}\left(t_m + c_i\Delta t_m, u^m + \Delta t_m\sum_{j=1}^{s}\mathbf{A}_{ij}\mathbf{k}^j, p^m + \Delta t_m\sum_{j=1}^{s}\mathbf{A}_{ij}l^j\right), \tag{48}$$

$$0 = g\left(t_m + c_i\Delta t_m, u^m + \Delta t_m\sum_{j=1}^{s}\mathbf{A}_{ij}\mathbf{k}^j, p^m + \Delta t_m\sum_{j=1}^{s}\mathbf{A}_{ij}l^j\right). \tag{49}$$

Note however, that due to the absence of a pressure time derivative in the continuity equation, the coefficients l^i are not well defined through (49). Therefore, we insert (47) into (48) and (49) to get

$$\mathbf{k}^1 = -\mathbf{f}\left(t_m,\mathbf{u}^m,p^m\right)/\rho\,, \qquad 0 = g\left(t_m,\mathbf{u}^m,p^m\right), \tag{50}$$

$$\mathbf{k}^2 = -\mathbf{f}\left(t_m+\kappa\Delta t_m,\mathbf{u}^{\mathrm{I}},p^{\mathrm{I}}\right)/\rho\,, \qquad 0 = g\left(t_m+\kappa\Delta t_m,\mathbf{u}^{\mathrm{I}},p^{\mathrm{I}}\right), \tag{51}$$

$$\mathbf{k}^3 = -\mathbf{f}\left(t_m+(\kappa+\tilde{\kappa})\Delta t_m,\mathbf{u}^{\mathrm{II}},p^{\mathrm{II}}\right)/\rho\,, \qquad 0 = g\left(t_m+(\kappa+\tilde{\kappa})\Delta t_m,\mathbf{u}^{\mathrm{II}},p^{\mathrm{II}}\right), \tag{52}$$

$$\mathbf{k}^4 = -\mathbf{f}\left(t_{m+1},\mathbf{u}^{\mathrm{III}},p^{\mathrm{III}}\right)/\rho\,, \qquad 0 = g\left(t_{m+1},\mathbf{u}^{\mathrm{III}},p^{\mathrm{III}}\right), \tag{53}$$

with solutions defined at intermediate steps

$$\mathbf{u}^{\mathrm{I}} = \mathbf{u}^m + \kappa\Delta t_m\left(\beta\mathbf{k}^1 + \alpha\mathbf{k}^2\right), \qquad p^{\mathrm{I}} = p^m + \kappa\Delta t_m\left(\beta l^1 + \alpha l^2\right), \tag{54}$$

$$\mathbf{u}^{\mathrm{II}} = \mathbf{u}^{\mathrm{I}} + \tilde{\kappa}\Delta t_m\left(\alpha\mathbf{k}^2 + \beta\mathbf{k}^3\right), \qquad p^{\mathrm{II}} = p^{\mathrm{I}} + \tilde{\kappa}\Delta t_m\left(\alpha l^2 + \beta l^3\right), \tag{55}$$

$$\mathbf{u}^{\mathrm{III}} = \mathbf{u}^{\mathrm{II}} + \kappa\Delta t_m\left(\beta\mathbf{k}^3 + \alpha\mathbf{k}^4\right), \qquad p^{\mathrm{III}} = p^{\mathrm{II}} + \kappa\Delta t_m\left(\beta l^3 + \alpha l^4\right). \tag{56}$$

With the help of (47) it is then easily verified that $\mathbf{u}^{\mathrm{III}} = \mathbf{u}^{m+1}$ and $p^{\mathrm{III}} = p^{m+1}$ and combining the auxiliary equations for $\mathbf{k}^i$, i.e., (50)–(53) and (54)–(56), we obtain exactly the systems corresponding to the fractional step θ-scheme. Now, the embedded scheme for the velocity components is

$$\tilde{\mathbf{u}}^{m+1} = \mathbf{u}^m + \Delta t_m \sum_{i=1}^{s} \tilde{b}_i \mathbf{k}^i, \tag{57}$$

but since neither the update formula (47) nor the intermediate solution definitions (54)–(56) or $\mathbf{A}_{4j} = \mathbf{b}_j$ can be used to recover the $\mathbf{k}^i$ – even with the solution vectors $\mathbf{u}^m$, $\mathbf{u}^{\mathrm{I}}$, $\mathbf{u}^{\mathrm{II}}$ and $\mathbf{u}^{\mathrm{III}}$ given – a projection is employed. Taking (50) and given the solution at the macro time step start $t = t_m$, i.e., $(\mathbf{u}_h^m, p_h^m, \mu_h^m) \in X_h \times Y_h \times Z_h$, find $\mathbf{k}_h^1 \in Z_h$, such that

$$\left\langle \mathbf{w}_h, \rho \mathbf{k}_h^1 \right\rangle = \left\langle \mathbf{w}_h, \rho \mathbf{g}^m - (\rho \nabla \mathbf{u}_h^m)\mathbf{u}_h^m + (\nabla \mathbf{u}_h^m)^{\top} \nabla \mu_h^m \right\rangle - \left\langle \nabla \mathbf{w}_h, \mu_h^m \nabla \mathbf{u}_h^m \right\rangle + \left\langle \nabla \cdot \mathbf{w}_h, p_h^m \right\rangle + \langle \mathbf{w}_h, \tilde{\mathbf{t}}^m \rangle_{\Gamma_N} \tag{58}$$

for all $\mathbf{w}_h \in \tilde{Z}_h$. Then, we can compute the remaining $\mathbf{k}_h^i$ reformulating (54)–(56) by

$$\mathbf{k}_h^2 = \frac{1}{\alpha}\left(\frac{\mathbf{u}_h^{\mathrm{I}} - \mathbf{u}_h^m}{\kappa\,\Delta t_m} - \beta \mathbf{k}_h^1\right), \quad \mathbf{k}_h^3 = \frac{1}{\beta}\left(\frac{\mathbf{u}_h^{\mathrm{II}} - \mathbf{u}_h^{\mathrm{I}}}{\tilde{\kappa}\,\Delta t_m} - \alpha \mathbf{k}_h^2\right), \quad \mathbf{k}_h^4 = \frac{1}{\alpha}\left(\frac{\mathbf{u}_h^{\mathrm{III}} - \mathbf{u}_h^{\mathrm{II}}}{\kappa\,\Delta t_m} - \beta \mathbf{k}_h^3\right), \tag{59}$$

which finally allows controlling the step size via the PI-controller [93]

$$\Delta t_{m+1} = \xi \, \frac{\Delta t_m^2}{\Delta t_{m-1}} \left(\frac{\epsilon_v \, e_m}{e_{m+1}^2}\right)^{\frac{1}{2}} \quad \text{with} \quad e_{m+1} = \left| \mathbf{u}_h^{m+1} - \tilde{\mathbf{u}}_h^{m+1} \right|, \tag{60}$$

with safety factor $\xi \in (0, 1]$.

Altogether, using the projection (58), i.e., a mass solve for the vector-valued $\mathbf{k}_h^1$, one can enable adaptive time step selection only based on a simple generalised θ-method time stepping. In the present case, the added numerical effort is close to negligible, since the projection is easily computed (at the end of every macro-step), especially when comparing to the effort involved in solving the velocity-pressure system.

Choosing any of the heuristic adaptive time-stepping schemes presented, we introduce an additional safety measure by repeating steps which either resulted in a time step reduction by more than 30% of the current step size or lead to non-convergence of the linear solver in the (final non-linear) step.

5. Linear systems and iterative solution

We aim to obtain a linear system from the consistently stabilised form using (32), (34) and (37), omitting the other forms for brevity. Standard continuous finite element spaces

$$X_h = \operatorname{span}\left\{\boldsymbol{\psi}_i\right\}_{i=1}^{3N^{\mathbf{u}}}, \quad Y_h = \operatorname{span}\left\{\phi_i\right\}_{i=1}^{N^p}, \quad Z_h = \operatorname{span}\left\{\varphi_i\right\}_{i=1}^{N^\mu} \tag{61}$$

are used, denoting by $\boldsymbol{\psi}_i$, ϕ_i and φ_i the vector- or scalar-valued shape functions corresponding to nodal degrees of freedom (DOFs) for velocity, pressure and viscosity fields. The nonlinearities in the system are treated either via Picard iterations or extrapolation (35). The nonlinear solver is used only for comparison, whereas the reason for choosing Picard is two-fold: (i) Newton's method introduces additional coupling terms, which hinder the decoupling of the equation governing viscosity, and (ii) fixed-point iterations have been shown to be as efficient as linearisation via Newton's method for three-dimensional incompressible flow problems in [94]. Then, discretisation naturally leads to the decoupled (scaled) viscosity and velocity-pressure sub-systems

$$\mathsf{M}\underline{\tilde{\mu}} = \underline{h} \qquad \text{and} \qquad \begin{pmatrix} \mathsf{A} & \mathsf{B} \\ \tilde{\mathsf{B}} & \mathsf{C} \end{pmatrix} \begin{pmatrix} \underline{\mathbf{u}}^{n+1} \\ \underline{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \underline{\mathbf{f}} \\ \underline{g} \end{pmatrix} \tag{62}$$

with scaled viscosity $\tilde{\mu} = \mu^{n+1}/\mu_\infty$ and entries of matrices and vectors as defined in Appendix A. This particularly convenient decoupling of the overall system is not possible applying Newton's method, and allows for a sequential solution procedure, where first the scaled viscosity is computed and then plugged into the second system as $\mu_h^{n+1} = \tilde{\mu}\,\mu_\infty$. This also means that exchanging the fluid's rheological law is as easy as adapting the right hand side $\underline{h}$.

To improve the solver's convergence behaviour in the nonlinear case of $\mathbf{v}_h^{n+1} = \mathbf{u}_h^{n+1}$, we apply Aitken's relaxation [95,96]. Therefore, relax the two individual sub-problem's solutions $\underline{\tilde{x}}^{k+1}$ in iteration $k+1$ with ω_{k+1} recursively defined as [97]

$$\underline{x}^{k+1} = \omega_{k+1}\,\underline{\tilde{x}}^{k+1} + (1 - \omega_{k+1})\,\underline{x}^k, \quad \text{with} \quad \omega_{k+1} = -\omega_k \frac{\underline{r}^k \cdot (\underline{r}^{k+1} - \underline{r}^k)}{|\underline{r}^{k+1} - \underline{r}^k|^2} \quad \text{and} \quad \underline{r}^{k+1} = \underline{\tilde{x}}^{k+1} - \underline{x}^k. \tag{63}$$

The linear algebraic systems (62) are solved in each iteration of the nonlinear scheme, or once at each time step in the linearised scheme. For both the viscosity sub-system and the projection to obtain $\mathbf{k}_h^1$ via (58), a conjugate gradient method [98] in combination with a single V-cycle of the algebraic multigrid method (AMG) as preconditioner, provided by [99], is found effective. The iterative solution of the velocity-pressure sub-system is known to be delicate – therefore, we employ a flexible generalised minimal residual method (FGMRES) [100] with the right preconditioner $\mathcal{P}^{-1}$ defined as [47]:

$$\mathcal{P}^{-1} = \begin{pmatrix} \mathrm{A} & \mathrm{B} \\ 0 & \mathrm{S} \end{pmatrix}^{-1} = \begin{pmatrix} \mathrm{A}^{-1} & 0 \\ 0 & \mathrm{I} \end{pmatrix} \begin{pmatrix} \mathrm{I} & -\mathrm{B} \\ 0 & \mathrm{I} \end{pmatrix} \begin{pmatrix} \mathrm{I} & 0 \\ 0 & \mathrm{S}^{-1} \end{pmatrix}, \tag{64}$$

with the Schur complement $\mathrm{S} := \mathrm{C} - \tilde{\mathrm{B}}\mathrm{A}^{-1}\mathrm{B}$. Exact application of $\mathcal{P}^{-1}$ would result in the outer Krylov method converging in at most two iterations, see [47], but is unfeasible in applications – however, the action of S^{-1} and A^{-1} and thus the action of $\mathcal{P}^{-1}$ on a vector within the FGMRES method can be approximated. The inverse of A is accounted for by a single AMG cycle, whereas obtaining a good representation of S^{-1} is based on investigating the contributions to A as defined in (A.4). We interpret A as a sum [45]

$$\mathrm{A} = \rho \mathrm{M}_{\mathbf{u}} + \mathrm{N}_{\mathbf{u}}(\mathbf{v}_h^{n+1}, \nabla \mu_h^{n+1}) + \mathrm{L}_{\mathbf{u}}(\mu_h^{n+1}, \tilde{\gamma}), \tag{65}$$

where $\mathrm{M}_{\mathbf{u}}$ is the velocity mass matrix, $\mathrm{N}_{\mathbf{u}}$ contains the convective and viscosity gradient terms, and $\mathrm{L}_{\mathbf{u}}$ composed of the diffusion and grad-div terms. Each of those constituents, or rather their inverses within the Schur complement, are then considered as

$$\mathrm{S}^{-1} \approx \left(\mathrm{C} - \tilde{\mathrm{B}}\,\mathrm{diag}(\rho \mathrm{M}_{\mathbf{u}})^{-1}\mathrm{B}\right)^{-1} - \mathrm{M}_p^{-1}\mathrm{F}_p\mathrm{L}_p^{-1} + \mathrm{M}_{\mu,\gamma}^{-1}. \tag{66}$$

Here, the matrix composed of reaction and stabilisation terms is explicitly computed to rigorously account for correct boundary conditions and stabilisation terms. The matrices defined on the pressure space are

$$\begin{aligned} &[\mathrm{M}_p]_{ij} = \langle \phi_i, \phi_j \rangle, \quad [\mathrm{L}_p]_{ij} = \langle \nabla\phi_i, \nabla\phi_j \rangle \quad \text{and} \\ &[\mathrm{F}_p]_{ij} = \left\langle \phi_i, \rho\phi_j + \tilde{\theta}\rho\nabla\phi_j \cdot \mathbf{v}_h^{n+1} \right\rangle + \left\langle \nabla\phi_i, \tilde{\theta}\mu_h^{n+1}\nabla\phi_j \right\rangle, \end{aligned} \tag{67}$$

with L_p incorporating homogeneous essential boundary conditions on Neumann boundaries of the flow problem and zero Neumann conditions elsewhere. The discrete operator F_p resembles the convection-diffusion-reaction operator on the velocity space and features the Robin boundary condition [101]

$$-\mu^{n+1}\nabla p^{n+1} \cdot \mathbf{n} + (\mathbf{u}^{n+1} \cdot \mathbf{n}) = 0 \qquad \text{on } \partial\Omega. \tag{68}$$

This whole strategy is a variation of the well-known pressure-convection-diffusion (PCD) preconditioner [53], which is slightly adapted here to tackle highly variable viscosity fields, taking inspiration from [62,58]:

$$[\mathrm{M}_{\mu,\gamma}]_{ij} = \left\langle \phi_i, (\mu_h^{n+1} + \tilde{\gamma})^{-1}\phi_j \right\rangle. \tag{69}$$

All the inverses appearing in the Schur complement approximation are also approximated by single cycles of an AMG method, making them cheap in execution. For each application of the preconditioner on a vector, one thus needs to execute altogether three AMG cycles on pressure matrices, one AMG cycle on the velocity-velocity block and two matrix-vector products.

6. Computational results

In this section, we assess the accuracy and robustness of our stabilised framework in two parts. First, we focus on establishing a systematic comparison between our BVS formulation and more conventional residual-based stabilisations. For that we consider stationary benchmark problems in two dimensions, focusing solely on the accuracy of the formulations *per se*. Once that is completed, expected convergence rates in space and time are confirmed also for the instationary solver using a manufactured solution. Then, flow in a straight pipe in two and three space dimensions is considered to assess solver performance with regards to time step selection and number of (non-)linear iterations in the case of quasi-stationary and periodic flow conditions. In a final example, we tackle the simulation of blood flow in the idealised geometry of an aneurysmatic vessel, demonstrating solver performance in a realistic setting with physiological inflow conditions and highlighting the importance of rheological modelling in hemodynamics. All examples in the latter part are implemented using the finite element toolbox `deal.II` [102]. In the examples varying α and/or checking convergence rates, parallel direct linear solves are executed [103].

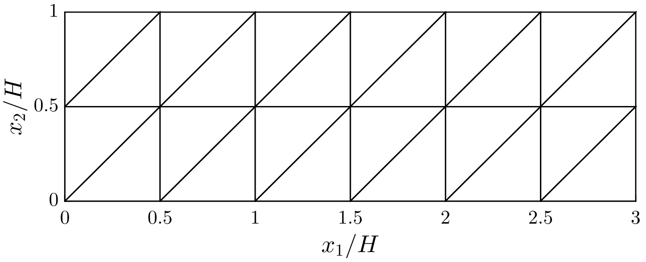


Fig. 1. Carreau channel flow: coarsest mesh used in the refinement study.

6.1. Stationary problems

We begin with two stationary examples to assess the accuracy of our stabilised formulation in comparison to the classical PSPG approach. Since we consider stationary problems with low Reynolds numbers, a stabilisation parameter in the form of Eq. (26) is employed. The approximation errors will be measured in relative L^2 norms:

$$\|p-p_h\|_0 := \frac{\|p-p_h\|_{L^2(\Omega)}}{\|p\|_{L^2(\Omega)}}, \quad \|\mathbf{u}-\mathbf{u}_h\|_0 := \frac{\|\mathbf{u}-\mathbf{u}_h\|_{L^2(\Omega)}}{\|\mathbf{u}\|_{L^2(\Omega)}}.$$

6.1.1. Carreau fluid in straight channel

In a two-dimensional channel $\Omega = (0, L) \times (-H/2, H/2)$ with a constant pressure drop per unit length equal to k, the developed Carreau flow has a known solution:

$$\mathbf{u}(\mathbf{x}) = \begin{pmatrix} u_1(x_2) \\ 0 \end{pmatrix}, \quad p(\mathbf{x}) = |k|(L - x_1), \quad \text{with} \quad u_1(x_2) = \int_{H/2}^{|x_2|} f(y)\,\mathrm{d}y$$

and $f(y)$ solving the nonlinear equation

$$\left\{\mu_\infty + (\mu_0 - \mu_\infty)\left[1 + (\lambda f/2)^2\right]^{\frac{\zeta-1}{2}}\right\} f = -|k|y, \quad y \in \left[0, \frac{H}{2}\right]. \tag{70}$$

The exact solution can then be used for measuring approximation errors in velocity and pressure. We consider a problem with $L = 3H = 3$ mm and a pressure drop of 9 Pa across the channel's length. The hemodynamic parameters for a representative setting are [68]: $\rho = 1050$ kg/m^3, $\mu_\infty = 3.45$ mPa·s, $\mu_0 = 56$ mPa·s, $\zeta = 0.3568$ and $\lambda = 3.313$ s. As for the BCs, we impose no-slip on the walls, i.e., $\mathbf{u}|_{x_2=\pm H/2} = \mathbf{0}$, and enforce the pressure drop by setting $\tilde{\mathbf{t}}|_{x_1=0} = (p_{\text{in}}, 0)^\top$ and $\tilde{\mathbf{t}}|_{x_1=L} = \mathbf{0}$, with $p_{\text{in}} = 9$ Pa.

We first carry out a convergence study with linear triangular elements, starting from the mesh depicted in Fig. 1 and then performing 6 levels of uniform refinement. Fig. 2 shows a comparison between the BVS and PSPG methods for a stabilisation factor $\alpha = 1$. We see that, while the velocity errors are quite similar, the BVS method performs clearly better in the approximation of pressure. At the finest level, the BVS error is already thirty times smaller than that attained by PSPG, and this difference shows an increasing trend if the refinement continues.

In fact, the similarity in the velocity errors does not hold in general; if α is not small, the incomplete PSPG residual ends up over-relaxing incompressibility, which leads to larger velocity errors. Fig. 3 shows the error behaviour for a wide range of stabilisation factors. For small α, the methods perform similarly because the divergence-free constraint dominates over the stabilisation terms. However, increasing the stabilisation factor leads to a much faster increase in the velocity error for PSPG than for our method. For instance, from $\alpha = 0.1$ to $\alpha = 10$ the PSPG method experiences a 1000% increase in the velocity error, in contrast to only 12% in the BVS case. This is due to the improved mass conservation of our approach. As a matter of fact, although we stick here to low-order pairs, a thorough comparison between BVS and PSPG presented by Pacheco et al. [41] for the Newtonian case reveals a similar gain in parameter-robustness also for higher-order elements.

Additionally to the possible over-relaxation of incompressibility, standard low-order residual-based formulations have another well-known shortcoming: the induction of spurious pressure boundary layers – and the straight channel flow is a perfect example for illustrating that. For this problem, the momentum equation simplifies to

$$\nabla p = \mu \Delta \mathbf{u} + \nabla \mathbf{u} \nabla \mu,$$

so that dividing both sides by μ and taking the divergence of the result gives us

$$\nabla \cdot \left(\mu^{-1} \nabla p\right) = \Delta (\nabla \cdot \mathbf{u}) + \frac{\partial}{\partial x_1}\left(\mu^{-1} \frac{\partial u_1}{\partial x_2} \frac{\partial \mu}{\partial x_2}\right) = 0.$$

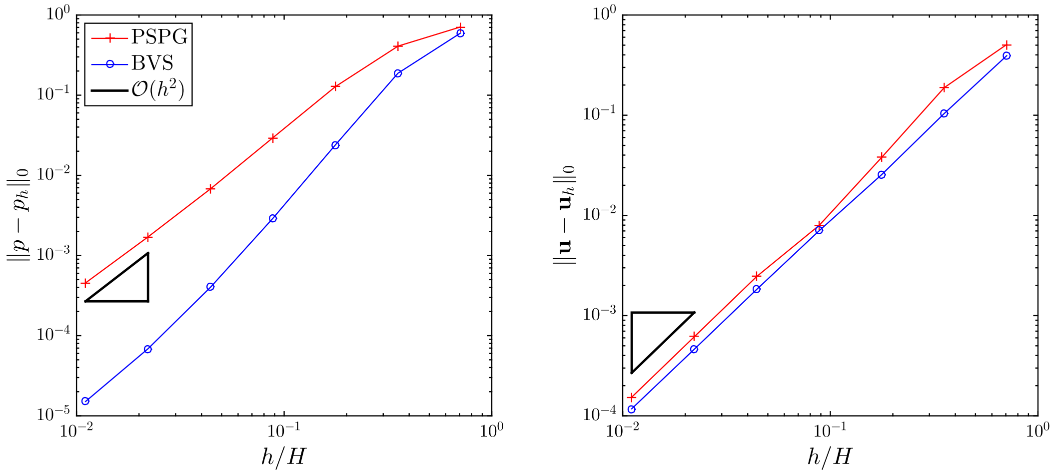


Fig. 2. Carreau channel flow: uniform refinement study (triangles indicate quadratic convergence).

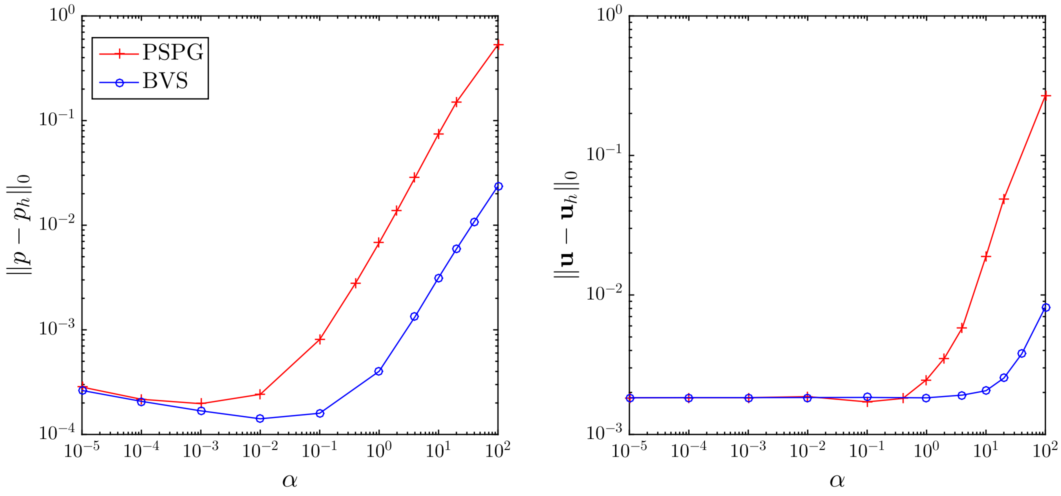


Fig. 3. Carreau channel flow: effect of the stabilisation parameter.

Thus, we can also write

$$\nabla \cdot \mathbf{u} - \delta \nabla \cdot \left(\mu^{-1} \nabla p\right) = 0 \tag{71}$$

for any arbitrary δ. Let us now turn our attention to the stabilised formulation. In a uniform mesh with linear triangular elements, we have $(\nabla \cdot \mathbb{S}_h)|_{\Omega_e} = \mathbf{0}$ and $h_e = h$ for all elements. Hence, the PSPG stabilisation for this example reads simply

$$\langle q_h, \nabla \cdot \mathbf{u}_h \rangle + \alpha h^2 \left\langle \nabla q_h, \mu_h^{-1} \nabla p_h \right\rangle = 0\,, \tag{72}$$

which is the weak form of

$$\nabla \cdot \mathbf{u} - \alpha h^2 \nabla \cdot \left(\mu^{-1} \nabla p\right) = 0 \text{ in } \Omega\,, \tag{73}$$

$$\frac{\partial p}{\partial n} = 0 \text{ on } \Gamma\,, \tag{74}$$

i.e., although Eq. (73) is consistent with Eq. (71), the PSPG term induces zero Neumann BCs (74) for the pressure as a numerical artifact. This results in loss of accuracy close to the boundary, especially if the stabilisation factor α is not small [104]. Therefore, α must be large enough to attain stability, but not so large that spurious pressure boundary layers become relevant – and this, of course, restricts the choice of the stabilisation parameter. In Fig. 4 we plot the pressure field along $x_2 = 0$, for different values of α. The induced Neumann BCs can be clearly seen in the PSPG solution, and the affected region becomes larger as we increase α. Conversely, when using the BVS this is not an issue, since the pressure satisfies appropriate BCs derived from the momentum equation.

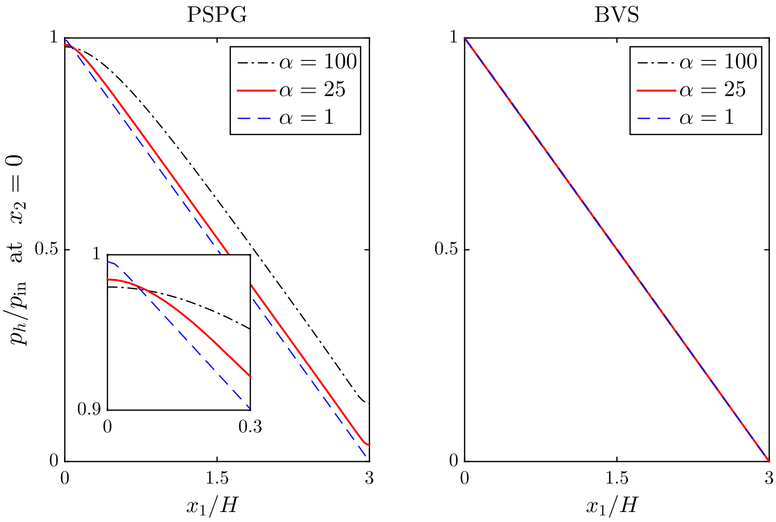


Fig. 4. Carreau channel flow: centreline pressure for different stabilisation factors α, showing the zero Neumann BCs induced by PSPG.

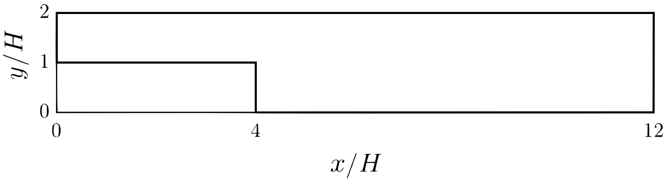


Fig. 5. Geometry used for the backward-facing step benchmark.

6.1.2. *Carreau–Yasuda fluid past a backward-facing step*

We now consider the classical backward-facing step benchmark (*cf.* Fig. 5) with a Carreau–Yasuda fluid [36,105]. This model generalises the Carreau rheology with the law

$$\eta\left(\dot\gamma\right) = \mu_\infty + \left(\mu_0 - \mu_\infty\right)\left[1 + \left(\lambda\dot\gamma\right)^a\right]^{\frac{\zeta-1}{a}} . \tag{75}$$

The hemodynamic parameters used in this example are [36]: $\rho = 1060$ kg/m^3, $\mu_\infty = 3.45$ mPa·s, $\mu_0 = 56$ mPa·s, $\zeta = 0.22$, $\lambda = 3.804$ s and $a = 1.25$. We prescribe a parabolic inflow at $x_1 = 0$, zero pseudo-traction at the outlet $x_1 = L$ and no-slip on the remainder of the boundary. For this example, we can write the Reynolds number as $\text{Re} = \rho Q/\mu_\infty$, with Q being the inflow rate. Results for Re $= 25$ have been reported by Masud and Kwack [36] for widths of $H = 0.5$, 5 and 50 mm. We compare them with respect to the normalised wall shear stress $\overline{\tau_w}$ downstream of the step along the line $x_2 = 0$:

$$\overline{\tau_w} := \frac{\left[\eta\left(\dot\gamma\right)\frac{\partial u_1}{\partial x_2}\right]\Big|_{x_2=0}}{6\mu_\infty Q/(2H)^2} ,$$

whose denominator corresponds to the wall shear stress of a developed Newtonian flow.

For the numerical solution, we use $\alpha = 0.1$ and a uniform mesh with 50,000 bilinear square elements in all three cases. Fig. 6 shows the comparison between our solutions and those attained by Masud and Kwack [36] through a variational multiscale (VMS) method. The results reveal good agreement, especially for the narrower channels. For the widest one there is a discrepancy of around 2%. Fortunately, there is a way to compute the exact wall shear stress of the developed flow due to equilibrium of forces:

$$\tau_w^\infty = -\frac{2H}{2}\frac{\partial p}{\partial x_1}\bigg|_{x_1\to\infty} . \tag{76}$$

Although the pressure gradient is not known right away, it is possible to find it in terms of the flow rate Q (this requires an iterative procedure, which we omit here for concision). Then, using the known value, we are able to verify that our solution differs in less than 0.5% from the exact one, while the VMS approximation differs in around 3%. For completeness, we present in Table 2 the comparison for the other geometries, along with the maximum and minimum values of the wall shear stress in each case.

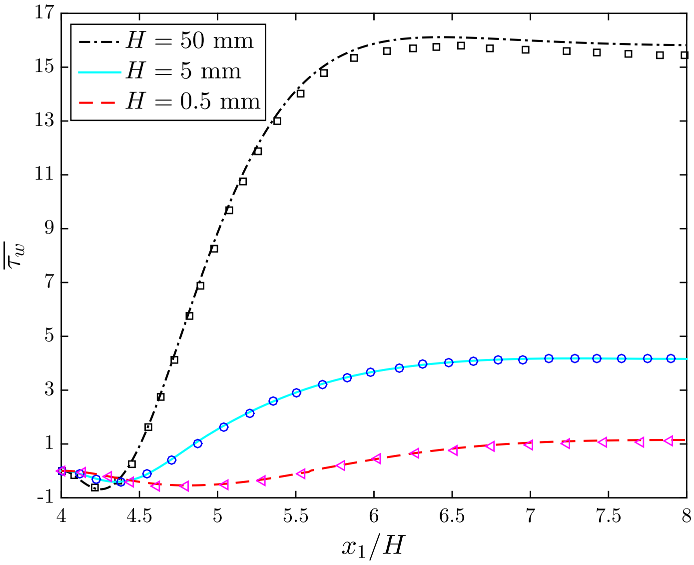


Fig. 6. Backward-facing step with Carreau–Yasuda fluid: wall shear stress ($x_2 = 0$) for the BVS (lines) and VMS (markers [36]) methods.

Table 2
Backward-facing step with Carreau–Yasuda fluid: key wall shear stress values $\overline{\tau_w}$ along the lower wall ($x_2 = 0$).

H [mm]	$\min\{\overline{\tau_w}\}$	$\max\{\overline{\tau_w}\}$	$\overline{\tau_w}\vert_{x_1=L}$	$\overline{\tau_w^\infty}$ (analytical)
0.5	−0.5394	1.147	1.087	1.099
5	−0.4048	4.180	4.139	4.167
50	−0.6814	16.115	15.839	15.897

6.2. Instationary problems

6.2.1. Manufactured solution

In the following, an exact solution to the system (1)–(4) as constructed in Appendix B is used for quantifying convergence. The convective term is not linearised, i.e., $\mathbf{v}_h^{n+1} = \mathbf{u}_h^{n+1}$, and the nonlinear system is solved until the relative criterion

$$|\underline{x}^{k+1} - \underline{x}^k| / |\underline{x}^{k+1}| \le 10^{-6} \tag{77}$$

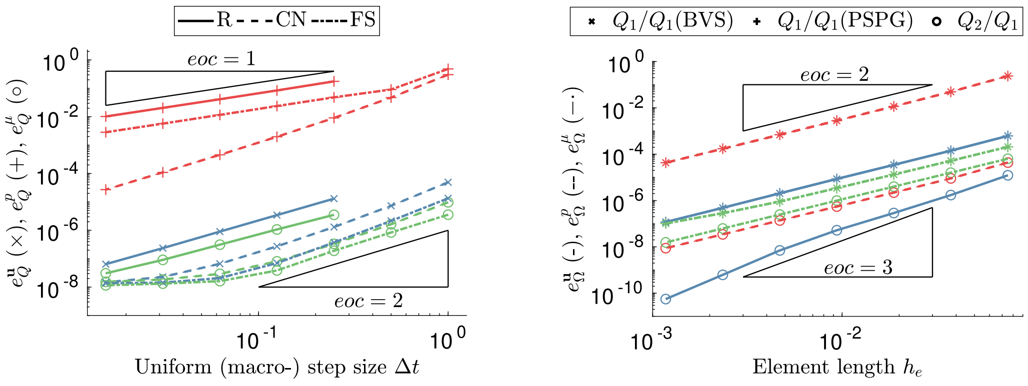
is met. To reduce the ambiguity coming from linear solver tolerance and start-up strategy, a parallel direct solver is employed [103] and the analytical solution for the pressure is simply interpolated at $t = 0$. Further, relative error norms with respect to some solution component are defined, exemplarily taking $\mathbf{u}_h$, as

$$e_Q^{\mathbf{u}} = \frac{||\mathbf{u}_h - \mathbf{u}||_{L^2(0,T;L^2(\Omega))}}{||\mathbf{u}||_{L^2(0,T;L^2(\Omega))}} \quad \text{and} \quad e_\Omega^{\mathbf{u}} = \left(\frac{||(\mathbf{u}_h - \mathbf{u})||_{L^2(\Omega)}}{||\mathbf{u}||_{L^2(\Omega)}} \right)\Bigg|_{t=T} . \tag{78}$$

For comparison, also the LBB-stable Taylor–Hood pair using Q_2/Q_1 elements for the velocity-pressure sub-system is included in the studies. For viscosity, Q_1 elements are used in any case.

Temporal convergence rates In a first numerical test of the time-dependent case, we solve the incompressible flow problem (1)–(2) in $\Omega = (0, 0.1)^2$ for $t \in (0, 1]$ with the parameters $\mu_0 = 10$ mPa·s, $\mu_\infty = 1.0$ mPa·s, $\rho = 10^3$ kg/m^3, $\zeta = 0.5$ and $\lambda = 1$ s. Spatial discretisation is carried out using a uniform grid of 256 × 256 BVS-stabilised Q_1/Q_1 elements to study temporal convergence rates. So, the number of (macro) time steps used is doubled from level to level, starting from a single one. The results are depicted in Fig. 7a, omitting Q_1/Q_1 (PSPG) and Q_2/Q_1 elements for brevity, which give identical results up to the spatial discretisation error.

Regarding the time discretisation, we aim to compare the pressure-corrected Crank–Nicolson integrator, pressure-corrected fractional step θ-method and Rannacher time-stepping scheme, denoted by CN, FS and R respectively. The R scheme is based on the CN method, but introduces intermediate backward Euler steps every N^* steps. Only with a fixed number of intermediate Euler steps can the observed convergence rates be achieved. Thus, when using the R scheme, we start with 4 steps in total and $N^* = 2$, which is doubled when halving the time step size. This leads to linear convergence in the pressure (+) and a shift in the velocity error (×) comparing R (—) with the optimally-convergent CN (- -) scheme. However, the temporal stability is noticeably improved compared to the CN scheme.



(a) Error e_Q using Q_1/Q_1 (BVS). **(b)** Error e_Ω using CN time-stepping.

Fig. 7. Error e_Q (a) using CN, R or FS schemes and error e_Ω (b) obtained with Q_2/Q_1 or Q_1/Q_1 elements stabilised via PSPG or BVS.

Those observed convergence rates match the error bounds established in [85], yet, fully implicit pressure time integration was employed therein. In our case, even a pressure-corrected form of Rannacher stepping did not improve the pressure error observed, while the velocity error convergences quadratically. A comparable approach was taken by [65–67], where instead of intermediate Euler steps, averaging steps were introduced – a stabilizing effect was observed, which is one possibility to counteract the ringing phenomenon when using CN with too large time steps. Introducing intermediate backward Euler steps instead is a worthwhile alternative which additionally dampens possible pressure oscillations in time. In practical applications, though, the parameter N^* controls the amount of numerical diffusion added also when using variable time step sizes, increasing the error.

Another important aspect is the order reduction in the pressure approximation (+) for the FS scheme (—·). Considering the stiff problem at hand, as laid out in [106,79], this is to be expected – a decrease in the convergence rate of the pressure is traded for some increased stability. Note at this point, however, that variable time step sizes lead to linear convergence in the pressure variable even for the CN scheme [107]. The FS scheme suffers from order reduction, also leading to linear convergence of pressure in time [79,107]. Consequently, one ends up with linear convergence in time for pressure using any of the above mentioned methods when adaptively choosing the time step size without including any postprocessing. For the R scheme, a fixed N^* leads to linear convergence of pressure *and* velocity in the observed norms; in practical applications, though, one rather thinks of an error corresponding to N^*.

Spatial convergence rates Within the second numerical test of the time-dependent case, we aim to showcase the spatial convergence rates. Therefore, the scenario is changed slightly by letting $t \in (0, 0.1]$, resolved by 100 time steps of the CN scheme to diminish the influence of time integration errors. The spatial approximation error in the relative $L^2(\Omega)$ norm as defined in (78) at the final time T is obtained, while dividing the domain $\Omega = (0, 0.3)^2$ into a sequence of uniformly refined grids starting from a 4×4 grid up to a 256×256 grid of rectangles. The rheological parameters are chosen as $\mu_0 = 10$ mPa·s, $\mu_\infty = 1$ mPa·s, $\rho = 10$ kg/m^3, $\zeta = 0.5$ and $\lambda = 1$ s, while the stabilisation parameter is defined setting $\alpha_1 = 2\alpha_3 = 4$ and $\alpha_2 = 0$ in (25).

The expected convergence rates are recovered, as seen in Fig. 7b, with the BVS and PSPG stabilisations giving almost identical errors using these settings. Moreover, the difference between stabilised and LBB-stable finite element pairs is also influenced by the physical parameters.

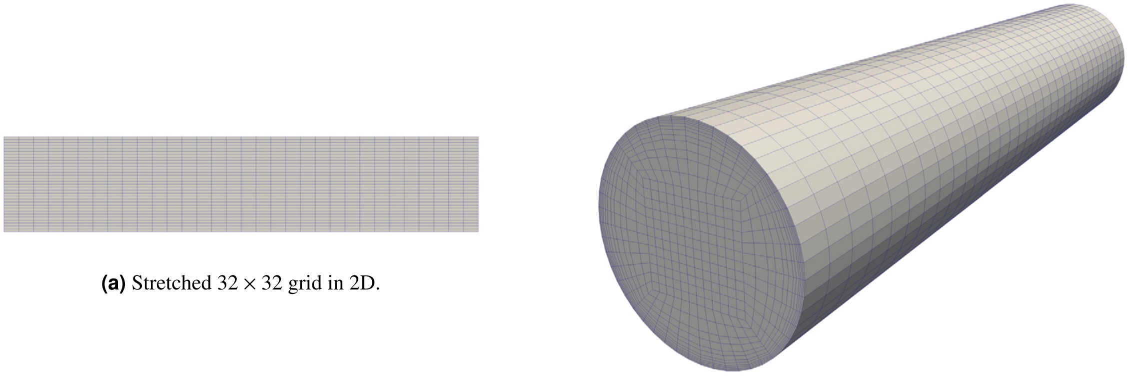
6.2.2. Adaptive time-stepping

The next numerical example demonstrates the solver performance with regards to time step selection in two scenarios resulting from different scaling of the inlet velocity profile in time. Both of the presented scenarios are based on the flow through a straight pipe with radius $R = 5$ mm and length $L = 50$ mm discretised as in Fig. 8a. As initial condition we choose the quiescent state, $\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0 = \mathbf{0}$, and start with a single implicit Euler step of size $\Delta t_0 = 1 \cdot 10^{-5}$ to circumvent the computation of the initial pressure field. On the inlet, i.e., at $x_1 = 0$, we enforce the ramped parabolic inflow profile by setting

$$u_1(\mathbf{x}, t) = \eta_{u_1} \hat{u}_1 \left[1 - \left(\frac{r}{R} \right)^2 \right] \tag{79}$$

$$\text{with} \quad \eta_{u_1} = \eta_r \, \eta_p \, , \quad \eta_r = \begin{cases} \sin^2\left(\frac{\pi t}{2T_r}\right) & \text{if } t < T_r \, , \\ 1 & \text{otherwise,} \end{cases} \tag{80}$$

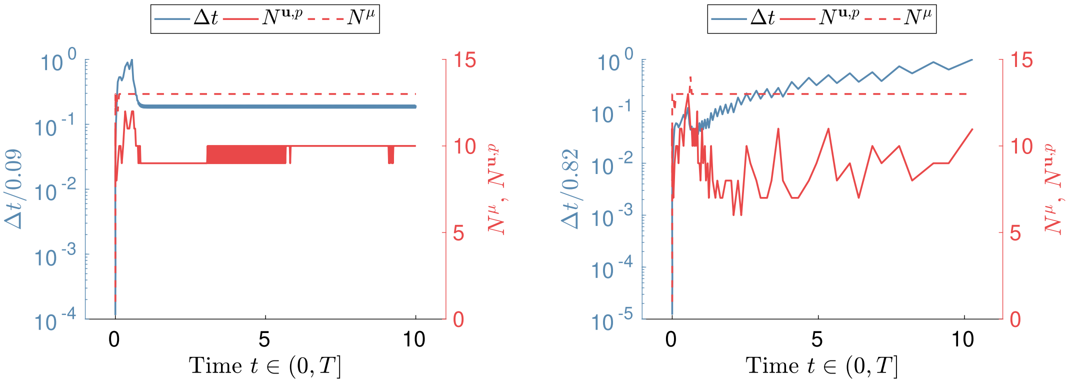
$$\text{and} \quad \eta_p = 1 + \alpha_p \left[\cos^{10}\left(\frac{t - T_r}{T_p} \pi \right) - 1 \right] . \tag{81}$$



(a) Stretched 32×32 grid in 2D.

(b) 3D mesh with $\approx 23 \cdot 10^3$ elements.

Fig. 8. Meshes considered for the first testcase in two (a) and three (b) space dimensions.



(a) CN-AB2, Q_2/Q_1 elements

(b) R-AB2 ($N^* = 10$), Q_2/Q_1 elements

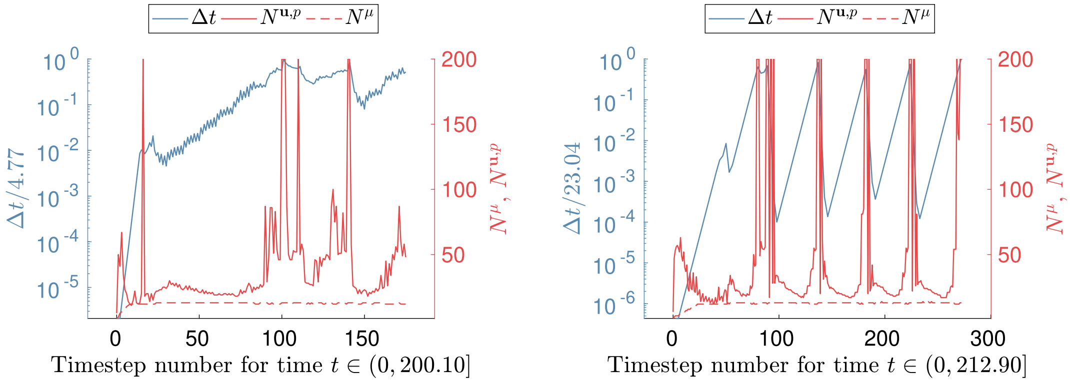
Fig. 9. Time step size (left axis, blue) and linear solver iterations (right axis, red). (For interpretation of the colours in the figure(s), the reader is referred to the web version of this article.)

On the cylinder wall we enforce no-slip BCs and on the outlet $x_1 = L$ the do-nothing condition $\tilde{\mathbf{t}} = \mathbf{0}$. The rheological parameters of the shear-thinning fluid are selected as $\mu_0 = 2.5 \cdot 10^{-1}$ Pa·s, $\mu_\infty = 3.5 \cdot 10^{-3}$ Pa·s, $\rho = 1060$ kg/m^3, $\zeta = 0.25$ and $\lambda = 25$ s. The linearisation (35) is employed and the FGMRES solver to tackle the velocity-pressure system uses a (rather crude) relative convergence tolerance of 10^{-4} using the last computed solution as an initial guess, while the time step selection safety factor $\xi = 0.98$ is used.

Quasi-stationary solution In the first scenario, a quasi-stationary solution in two space dimensions is generated by setting $\hat{u}_1 = 0.01$ m/s, $T_r = 0.5$ s and $\alpha_p = 0$. Therefore, time step selection merely controls how fast the final solution is achieved with a tolerance of $\epsilon_v = 1 \cdot 10^{-3}$ on the velocity error. Additionally, we switch off the re-computation of any time step, regardless of time step reduction factor and allow a maximum of 200 iterations in the linear solvers.

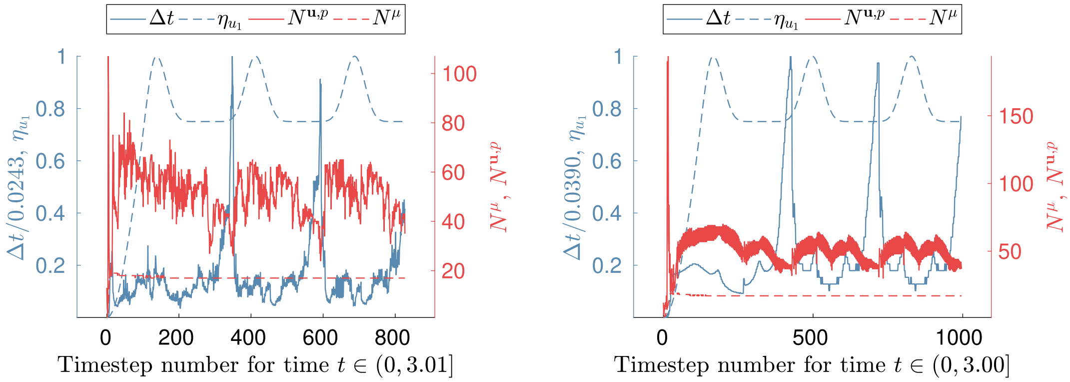
First, the stabilizing effect of the R scheme in comparison to CN is shown. Both schemes are combined with the explicit AB2 predictor to give what we refer to as the CN-AB2 and R-AB2 schemes, respectively. With the R-AB2 scheme, $N^* = 10$ sets the number of stabilisation steps introduced. These stabilisation steps allow the integrator to surpass a certain limit in the time step size encountered with CN-AB2, which for this scenario lies at $\Delta t_n \approx 0.018$ s. The maximum step size reached in the interval of interest with R-AB2 is $\Delta t_n \approx 0.82$ s. At the same time, the iteration counts stay nicely bounded independent of the time step size, with typical values for the velocity-pressure system around $N^{\mathbf{u},p} = 10$ and the viscosity solve $N^{\mu} = 12$. Corresponding graphs in Fig. 9 depict the time step size in logarithmic scale (left axes, blue) together with the iteration counts (right axes, red) in the linear solvers over the physical time t.

Secondly, the predictor-corrector scheme R-AB2 is compared to the FS scheme with PI-controller using the same problem parameters, but with Q_1/Q_1 elements stabilised via BVS. In Fig. 10, graphs again show time step size and iteration counts of the linear solvers when computing a considerably longer time interval of 200 s. Due to the aggressive pseudo-time incrementation and tolerances, time steps grow fast and are cut back regularly. These reductions in the time step size are triggered by instabilities resulting from time integration, when the time step control recommends step sizes above the stability limit as a result of a constant solution. Approaching this constant solution, the relative convergence criterion based on the last time step value becomes increasingly hard to satisfy, while the initial guess gives almost zero absolute error. The



(a) R-AB2 ($N^* = 10$), Q_1/Q_1 elements (BVS) **(b)** FS, Q_1/Q_1 elements (BVS)

Fig. 10. Time step size (left axis, blue) and linear solver iterations (right axis, red).



(a) R-AB2 ($N^* = 15$), Q_1/Q_1 elements (BVS) **(b)** FS, Q_1/Q_1 elements (BVS)

Fig. 11. Time step size and scaling $\eta_{\mathbf{u}_1}$ (left axis, blue) and linear solver iterations (right axis, red).

resulting non-convergence, together with temporal instabilities encountered when using large time steps lead to increasing error estimates, which results in a reduction of the time step size again. Nonetheless, after some non-converging linear solves, the suggested time steps tend to rise again, repeatedly reaching the same limit.

Comparing the R-AB2 with the FS scheme, we observe here a maximum time step size to be around $\Delta t_n \approx 4.77$ s for the R-AB2 and around $\Delta t_m \approx 23.04$ s for the macro time step of the FS scheme, respectively. Looking at the sub-steps, rather comparable sizes are obtained. Yet, the FS scheme results in a smoother time step growth (Fig. 10b), whereas the graph corresponding to the R-AB2 integrator (Fig. 10a) clearly shows an increasing step size every $N^* = 10$ steps and slightly oscillatory behaviour in-between. The number of total solves needed until passing the termination time T is higher for the FS scheme, since the macro time step size is controlled every three fractional steps and the stability limit of time integration is reached more often.

Periodic solution In a second scenario, a periodic inlet velocity profile in three space dimensions is enforced with parameters $\hat{u}_1 = 0.05$ m/s, $T_r = 0.5$ s, $\alpha_p = 0.25$ and $T_p = 1.0$ s. Defining the Reynolds number as in the Newtonian case, one arrives at $\mathrm{Re} = \rho u_{\mathrm{mean}} 2R/\mu_\infty \approx 75$. The tolerance on the velocity error is chosen as $\epsilon_v = 1 \cdot 10^{-4}$. Conversely to the first scenario, we now recompute time steps leading to a time step reduction of more than 30% with simply halved (macro) time step size. As before, Q_1/Q_1 elements using BVS stabilisation are employed, but this time in the three-dimensional mesh depicted in Fig. 8b. In Fig. 11 the obtained results using both the FS and R-AB2 ($N^* = 15$) time-stepping schemes are depicted. For the FS scheme, every solve, i.e., sub-step within the FS scheme is plotted, but regarding the time step size, the macro time step is considered for an overall better comparison with R-AB2. Additionally, the temporal scaling η_{u_1} of the inlet velocity profile is included.

Comparing Figs. 11a and 11b, we see that both time step selection schemes adapt to the present flow conditions. The R-AB2 scheme with $N^* = 15$ appears rather jumpy, which is caused by the tolerance ($\epsilon_v = 1 \cdot 10^{-4}$) and the resulting time step size. The depicted macro time step size of the FS scheme is much smoother, but of course adapted every third step only. In total, using the FS scheme 996 solves were completed, whereas with the R-AB2 scheme, 827 solves were performed. Depending on the tolerance set, either one of the presented approaches might come out on top, but the observations made remain unaltered.

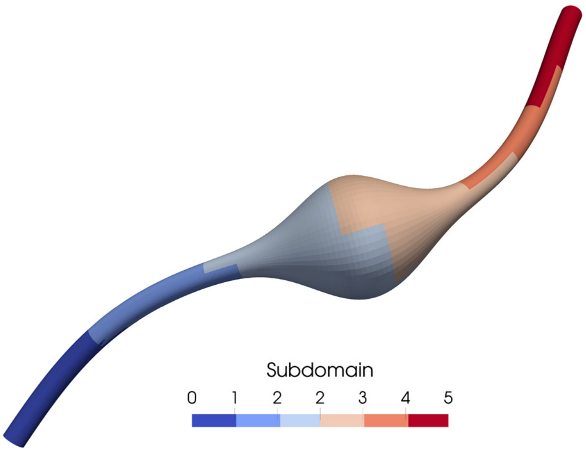


Fig. 12. Subdomains in artificial aneurysm mesh using 6 processors.

Table 3
Fourier coefficients fit to experimental data [109].

k	0	1	2	3	4	5	6	7
a_k	33.29	−5.27	−5.10	−2.81	−2.15	−0.12	0.79	0.27
b_k	–	9.52	0.72	0.29	−3.25	−0.95	−1.17	−0.50

Regarding linear solver performance, it is observed that for both time-stepping schemes the iteration counts $N^{\mu} \approx 17$ are constant throughout the entire simulation and the velocity-pressure sub-system solve does mildly depend on the (sub-) step size, with values around $N^{\mathbf{u},p} \approx 50$. Iteration counts are higher after some initial steps due to (i) largely varying time step sizes and consequently worse extrapolations used as initial guess, combined with (ii) the relative convergence criterion, taking the last solution as reference, which becomes increasingly difficult as $\Delta t_n \to 0$. Smoother variations in time step size and iteration counts would result from using a smaller tolerance ϵ_v.

In general, the same trends as reported in [52,48] are observed: the PCD preconditioner – and thus the variantion presented herein – suffers from a dependence on the Reynolds number when applied to stationary problems, but this effect is not significant or at least largely reduced, when time dependent flows are considered, i.e., a large enough reaction term is present.

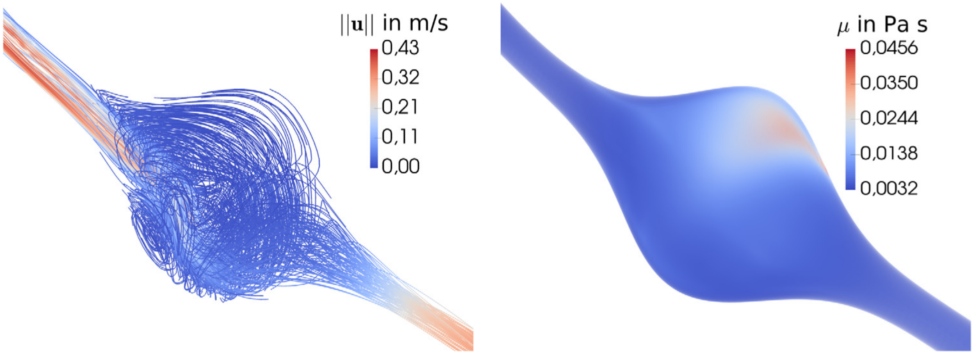
6.2.3. Flow through idealised aneurysm

Finally, solver performance in a more realistic scenario, namely, the flow through an idealised cerebral aneurysm in the internal carotid artery under physiological conditions is showcased to serve as a computational test of practical relevance. In order to construct the geometry, a straight cylinder of length $L = 160$ mm and radius $R = 2.5$ mm is bent and additionally “inflated”: the cross-section remains circular at the inlet and outlet, but is transformed to an ellipse with variable axis lengths of up to $a = 15$ mm at some point along the centerline. In Fig. 12, the final mesh is depicted, which consists of 173,600 hexahedra distributed to 6 subdomains. The number of nodes in the mesh is around $1.8 \cdot 10^5$, which is equal to the number of pressure and viscosity DOFs, and results – due to the equal-order interpolation – in $\approx 5.4 \cdot 10^5$ velocity unknowns. The overall system consists of $\approx 9.0 \cdot 10^5$ DOFs, whereas using Q_2/Q_1 elements would result in over five times as many.

No-slip conditions are enforced on the vessel walls, whereas a do-nothing condition is enforced on the outlet. While the latter choice is not suitable with regards to patient-specific simulations, we restrict ourselves to this simpler choice for the sake of brevity. More sophisticated models incorporating the downstream vasculature resistance and capacitance, e.g., variants of the Windkessel model or 3D-1D couplings (see e.g. [108]) can be straightforwardly considered. Starting from a quiescent state, i.e., $\mathbf{u} = \mathbf{0}$, the inflow is scaled similarly to (79)–(81), but replacing the periodic scaling η_p in (81) with a truncated Fourier series

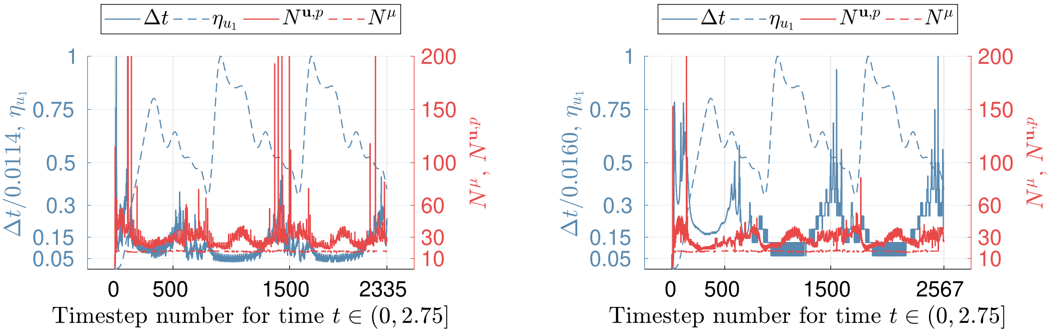
$$\eta_p = \frac{1}{50}\left\{a_0 + \sum_{k=1}^{7}\left[a_k \cos\left(2\pi k \frac{t}{T_p}\right) + b_k \sin\left(2\pi k \frac{t}{T_p}\right)\right]\right\}. \tag{82}$$

The coefficients a_k and b_k, given in Table 3, fit cross sectional peak velocity measurement data [109] and completely describe the inflow velocity profile together with $T_p = 0.917$ s, $T_r = T_p/2$ and peak velocity $\hat{u}_1 = 0.49886$ m/s. Rheological parameters are based on physiological measurements [110], setting $\mu_0 = 45$ mPa·s, $\mu_\infty = 3.2$ mPa·s, $\lambda = 10.03$ s and $\zeta = 0.344$. The Reynolds number computed as in the Newtonian limit, i.e., $\mathrm{Re} = \rho u_{\mathrm{mean}} 2R/\mu_\infty$ is then $\mathrm{Re} \approx 414$. As a consequence, the periodic solution features complex – yet still laminar – flow and recirculation regions with the dynamic viscosity vary-

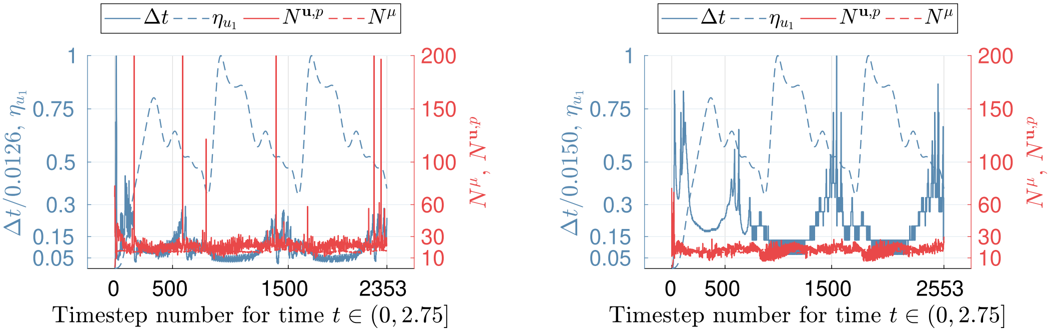


(a) Selected streamlines coloured by velocity norm. **(b)** Dynamic viscosity μ.

Fig. 13. Snapshot of recirculating flow in aneurysm at final third peak systole at time $t = 1.988$ s.



(a) Linearised $\mathbf{v}_h^{n+1}$, R-AB2. **(b)** Linearised $\mathbf{v}_h^{n+1}$, FS.



(c) Nonlinear $\mathbf{v}_h^{n+1}$, R-AB2. **(d)** Nonlinear $\mathbf{v}_h^{n+1}$, FS.

Fig. 14. Time step size and scaling $\eta_{\mathbf{u}_1}$ (left axis, blue) and linear iterations (right axis, red) using linearisation (top row) or Picard scheme (bottom row).

ing vastly in the aneurysm. A representative snapshot of the solution is depicted in Fig. 13, showing selected streamlines coloured by velocity magnitude (13a) and dynamic viscosity (13b) at peak systole in the third cardiac cycle considered, i.e., at $t = 1.988$ s.

Regarding the solver settings, we use the tolerance $\epsilon_v = 5 \cdot 10^{-3}$ for the FS and the R-AB2 ($N^* = 15$) time-stepping scheme, initial step size $\Delta t_0 = 1 \cdot 10^{-5}$, safety factor $\xi = 0.98$ and recompute steps leading to a step size reduction of more than 30%. The last computed solution is taken as initial guess for the linear solver which reduces the relative residual by a factor of 10^{-4} within a maximum of 200 iterations, or the step is recomputed with halved step size. $Q1/Q1$ elements with BVS-stabilisation are employed. The convective velocity $\mathbf{v}_h^{n+1}$ is either extrapolated via (35) or kept nonlinear, in which case the Picard scheme with Aitken relaxation is needed. In the latter case, we use a relative tolerance of 10^{-3} in the convergence criterion of the nonlinear solver (77) and get away with a comparably higher relative convergence criterion in the linear solver of 10^{-2}.

In Fig. 14, the obtained results are summarised. There, time step sizes and linear iteration counts or the mean linear iteration counts of the nonlinear steps per time step are depicted. Comparing the linearised versions (top row) with their nonlinear counterparts (bottom row), an almost identical number of time steps is performed. Furthermore, the chosen time step as well as the maximum step size matches well in pattern. The linear iteration counts are slightly higher for the

Table 4
Performance comparison of (non-)linear R-AB2 ($N^* = 15$) and FS schemes, format: total (**mean**).

	R-AB 2 (linearised)	FS (linearised)	R-AB 2 (nonlinear)	FS (nonlinear)
time steps N^t	2336	2568	2354	2554
system solves	2336 (**1.00**)	2568 (**1.00**)	8525 (**3.62**)	8676 (**3.40**)
$N^{\mathbf{u},p}$	68425 (**29.29**)	73004 (**28.43**)	190972 (**22.40**)	159966 (**18.44**)
N^{μ}	39318 (**16.83**)	43141 (**16.90**)	138714 (**16.27**)	138359 (**15.95**)
mean time/$\mathbf{u}$-p system	39.95 s	30.72 s	21.94 s	18.27 s
relative computing time	1.03	1.00	2.90	2.83

linearised scheme, since a smaller tolerance 10^{-4} is used, whereas the Picard scheme wrapping the linear systems allows for a bigger tolerance 10^{-2}. Moreover, it can be seen that strong gradients in the prescribed inlet velocity scaling $\eta_{\mathbf{u}_1}$ lead to higher linear iteration counts at certain points in time due to a worsening of the computed initial guess.

To further compare the performance of those four schemes, Table 4 lists the mean number of nonlinear and linear iterations over all time steps and the total number of solves performed together with the mean time spent on solving the velocity-pressure systems and overall computing time. The linearised schemes have similar overall computing times with comparable mean number of linear iterations. However, the nonlinear variants have a computing time close to 3 times longer: despite profiting from a less strict convergence criterion used in the linear solves, performing several nonlinear steps is costly – depending of course on the tolerances set. Altogether, the use of linearised schemes can result in considerable speedup, especially if the error tolerance ϵ_v and the CFL condition limit the time step size drastically, but accuracy and stability are of course affected.

7. Conclusion and outlook

Within this contribution, we have presented a new stabilised framework for generalised Newtonian fluid flows preserving consistency even for lower equal-order velocity-pressure pairs. The stabilisation is based on adding to the continuity equation a consistent pressure Poisson equation, and is the centrepiece of fast and easy-to-implement finite element solvers using standard continuous finite element spaces. Comparisons between our stabilisation method and more classical approaches have shown a substantial increase in accuracy, as a result of improving mass conservation and eliminating spurious pressure boundary layers.

To further improve performance, we presented several modifications for the time-dependent setting. Extrapolating the velocity field enables linearising the linear momentum balance and further decouples the viscosity from the main system. As a consequence, the rheological law is effortlessly exchanged and the resulting sub-problems – a simple projection and the usual velocity-pressure system – are only solved once per time step. The time step size is chosen adaptively using either a stabilised Adams–Bashforth predictor with a corrector based on Rannacher time-stepping or, alternatively, by a fractional step θ-scheme interpreted as a generalised θ-method including an embedded scheme. Both methods allow an efficient time step selection in quasi-stationary and transient problems. To solve the resulting velocity-pressure system, we adapt the PCD preconditioner (originally devised for Newtonian fluids) to non-Newtonian flows with highly varying viscosity. We tested our methods on two- and three-dimensional numerical examples, comparing them with existing approaches from literature and highlighting their potential in challenging scenarios.

Future and ongoing work is centred around applying the present framework to patient-specific datasets accounting for non-Newtonian hemodynamics and using our methods to construct efficient fluid-structure interaction solvers. Moreover, a recently published work comparing the performance of our framework with that of state-of-the-art non-Newtonian split-step schemes demonstrates that the present coupled solvers are indeed competitive with fast projection-like methods [34].

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors gratefully acknowledge Graz University of Technology for the financial support of the Lead-project: Mechanics, Modeling and Simulation of Aortic Dissection.

Appendix A. Element matrices and vectors

The discretisation of Equations (32), (37) and (34) naturally leads to the decoupled (scaled) viscosity and velocity-pressure sub-systems

$$\mathrm{M}\underline{\tilde{\mu}} = \underline{h} \quad \text{and} \quad \begin{pmatrix} \mathrm{A} & \mathrm{B} \\ \tilde{\mathrm{B}} & \mathrm{C} \end{pmatrix} \begin{pmatrix} \underline{\mathbf{u}}^{n+1} \\ \underline{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \underline{\mathbf{f}} \\ \underline{g} \end{pmatrix}, \tag{A.1}$$

with scaled viscosity $\tilde{\mu} = \mu^{n+1}/\mu_\infty$ and entries of matrices and vectors defined as

$$[\mathrm{M}_\mu]_{ij} = \left\langle \varphi_i, \varphi_j \right\rangle , \tag{A.2}$$

$$[\underline{h}]_i = \left\langle \varphi_i, 1 + (\mu_0/\mu_\infty - 1)\left(1 + 0.5\lambda^2 \nabla^s \mathbf{v}_h^{n+1} : \nabla^s \mathbf{v}_h^{n+1}\right)^{\frac{\zeta-1}{2}} \right\rangle , \tag{A.3}$$

$$[\mathrm{A}]_{ij} = \left\langle \boldsymbol{\psi}_i, \rho \boldsymbol{\psi}_j + \tilde{\theta}\left[\left(\rho \nabla \boldsymbol{\psi}_j\right)\mathbf{v}_h^{n+1} - (\nabla \boldsymbol{\psi}_j)^\top \nabla \mu_h^{n+1}\right]\right\rangle + \left\langle \nabla \boldsymbol{\psi}_i, \tilde{\theta}\mu_h^{n+1} \nabla \boldsymbol{\psi}_j \right\rangle + \sum_{e=1}^{N_e} \left\langle \tilde{\gamma}_e \nabla \cdot \boldsymbol{\psi}_i, \nabla \cdot \boldsymbol{\psi}_j \right\rangle_{\Omega_e} , \tag{A.4}$$

$$[\mathrm{B}]_{ij} = -\left\langle \nabla \cdot \boldsymbol{\psi}_i, \tilde{\theta}\phi_j \right\rangle , \tag{A.5}$$

$$\begin{aligned} [\underline{\mathbf{f}}]_i = &\left\langle \boldsymbol{\psi}_i, \rho(\mathbf{u}_h^n + \tilde{\theta}\mathbf{g}^{n+1}) - \tilde{\theta}'\left[\left(\rho \nabla \mathbf{u}_h^n\right)\mathbf{u}_h^n - \rho \mathbf{g}^n - (\nabla \mathbf{u}_h^n)^\top \nabla \mu_h^n\right]\right\rangle \\ &- \left\langle \nabla \boldsymbol{\psi}_i, \tilde{\theta}'\mu_h^n \nabla \mathbf{u}_h^n \right\rangle + \left\langle \nabla \cdot \boldsymbol{\psi}_i, \tilde{\theta}' p_h^n \right\rangle + \langle \boldsymbol{\psi}_i, \tilde{\theta}\tilde{\mathbf{t}}^{n+1} + \tilde{\theta}'\tilde{\mathbf{t}}^n \rangle_{\Gamma_N} , \end{aligned} \tag{A.6}$$

$$[\tilde{\mathrm{B}}]_{ij} = \langle \tau_i \nabla \phi_i \times \mathbf{n}, \tilde{\theta}\mu_h^{n+1} \nabla \times \boldsymbol{\psi}_j \rangle_\Gamma + \left\langle \tau_i \nabla \phi_i, \rho \boldsymbol{\psi}_j + \tilde{\theta}\left[(\rho \nabla \boldsymbol{\psi}_j)\mathbf{v}_h - 2(\nabla \boldsymbol{\psi}_j)^\top \nabla \mu_h^{n+1}\right]\right\rangle + \sum_{e=1}^{N_e} \left\langle \frac{\tau_i}{\delta_e}\phi_i, \nabla \cdot \boldsymbol{\psi}_j \right\rangle_{\Omega_e} , \tag{A.7}$$

$$[\mathrm{C}]_{ij} = \left\langle \tau_i \nabla \phi_i, \nabla \phi_j \right\rangle , \tag{A.8}$$

$$[\underline{g}]_i = \left\langle \tau_i \nabla \phi_i, \rho(\mathbf{u}_h^n + \tilde{\theta}\mathbf{g}^{n+1}) - \tilde{\theta}'\left[(\rho \nabla \mathbf{u}_h^n)\mathbf{u}_h^n - \rho \mathbf{g}^n + \nabla p_h^n - 2(\nabla \mathbf{u}_h^n)^\top \nabla \mu_h^n\right]\right\rangle - \langle \tau_i \nabla q_h \times \mathbf{n}, \tilde{\theta}'\mu_h^n \nabla \times \mathbf{u}_h^n \rangle_\Gamma . \tag{A.9}$$

In the rows corresponding to the pressure test functions, a rescaling is performed by the element-averaged factors τ_i defined per node as

$$\tau_i = \left(\sum_{e=1}^{N_i} |\Omega_e|\right)^{-1} \sum_{e=1}^{N_i} \delta_e |\Omega_e| \quad \text{with} \quad |\Omega_e| = \int_{\Omega_e} \mathrm{d}\Omega \tag{A.10}$$

and the number of elements N_i touching vertex i to counteract resulting numerical difficulties when $\delta_e \to 0$ as $h_e \to 0$.

Appendix B. Manufactured solution

An exact solution to the system comprised of (1)–(4) is constructed by specifying

$$u_1(t, \mathbf{x}) = \sin x_1 \cos x_2 \cos t \quad \text{and} \quad p(t, \mathbf{x}) = \cos(x_1 x_2) \cos t , \tag{B.1}$$

inserting u_1 into the continuity equation (2) to obtain for the second velocity component

$$u_2(t, \mathbf{x}) = -\cos t \cos x_1 \sin x_2 \quad \text{and} \quad \dot{\gamma}(t, \mathbf{x}) = \cos t \cos x_1 \cos x_2 \tag{B.2}$$

for the corresponding shear rate, which is further inserted into the rheological law (4) giving $\mu(t, \mathbf{x}) = \eta(\dot{\gamma})$. Then, in a final step, those expressions are inserted into the balance of linear momentum (1) to determine the (rather lengthy) analytical expression for the body force vector $\mathbf{g}(t, \mathbf{x}) = (g_1, g_2)^\top$ with components defined via

$$\begin{aligned} \rho g_1 = \;& 2 \cos t \cos x_2 \sin x_1 \left(\mu_\infty + (\mu_0 - \mu_\infty)\left[1 + (\lambda \dot{\gamma})^2\right]^{\frac{\zeta-1}{2}}\right) \\ &- \rho \cos x_2 \sin t \sin x_1 - x_2 \sin(x_1 x_2) \cos t + \rho \cos^2 t \cos x_1 \sin x_1 \\ &+ 2\rho(n-1)\,(\lambda \dot{\gamma})^2 \cos t \cos x_2 \sin x_1 (\mu_0 - \mu_\infty)\left[1 + (\lambda \dot{\gamma})^2\right]^{\frac{\zeta-3}{2}} , \end{aligned} \tag{B.3}$$

$$\begin{aligned} \rho g_2 = & -2 \cos t \cos x_1 \sin x_2 \left(\mu_\infty + (\mu_0 - \mu_\infty)\left[1 + (\lambda \dot{\gamma})^2\right]^{\frac{\zeta-1}{2}}\right) \\ &+ \rho \cos x_1 \sin t \sin x_2 - x_1 \sin(x_1 x_2) \cos t + \rho \cos^2 t \cos x_2 \sin x_2 \\ &- 2\rho(n-1)(\lambda \dot{\gamma})^2 \cos t \cos x_1 \sin x_2 (\mu_0 - \mu_\infty)\left[1 + (\lambda \dot{\gamma})^2\right]^{\frac{\zeta-3}{2}} . \end{aligned} \tag{B.4}$$

References

[1] G.P. Galdi, R. Rannacher, A.M. Robertson, S. Turek, Hemodynamical Flows, Oberwolfach Seminars, vol. 37, Birkhäuser, Basel, 2008.
[2] L. John, P. Pustějovská, O. Steinbach, On the influence of the wall shear stress vector form on hemodynamic indicators, Comput. Vis. Sci. 18 (4–5) (2017) 113–122.
[3] P. Nithiarasu, A fully explicit characteristic based split (CBS) scheme for viscoelastic flow calculations, Int. J. Numer. Methods Eng. 60 (5) (2004) 949–978.
[4] I.G. Donev, B.D. Reddy, Time-dependent finite element simulations of a shear-thinning viscoelastic fluid with application to blood flow, Int. J. Numer. Methods Fluids 75 (9) (2014) 668–686.
[5] G. Böhme, J. Broszeit, Numerical flow simulation for Bingham plastics in a single-screw extruder, Theor. Comput. Fluid Dyn. 9 (1) (1997) 65–74.
[6] T.J.R. Hughes, L.P. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuška-Brezzi condition: a stable Petrov-Galerkin formulation of the Stokes problem accommodating equal-order interpolations, Comput. Methods Appl. Mech. Eng. 59 (1) (1986) 85–99.
[7] T.J.R. Hughes, L.P. Franca, A new finite element formulation for computational fluid dynamics: VII. The Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces, Comput. Methods Appl. Mech. Eng. 65 (1) (1987) 85–96.
[8] L. Rubart, G. Böhme, Numerical simulation of shear-thinning flow problems in mixing vessels, Theor. Comput. Fluid Dyn. 3 (2) (1991) 95–115.
[9] H.R. Tamaddon-Jahromi, D. Ding, M.F. Webster, P. Townsend, A Taylor–Galerkin finite element method for non-Newtonian flows, Int. J. Numer. Methods Eng. 34 (3) (1992) 741–757.
[10] J.P.W. Baaijens, A.A. Van Steenhoven, J.D. Janssen, Numerical analysis of steady generalized Newtonian blood flow in a 2D model of the carotid artery bifurcation, Biorheology 60 (1) (1993) 63–74.
[11] C. Tu, M. Deville, Pulsatile flow of non-Newtonian fluids through arterial stenoses, J. Biomech. 29 (7) (1996) 899–908.
[12] N. Arada, M. Pires, A. Sequeira, Viscosity effects on flows of generalized Newtonian fluids through curved pipes, Comput. Math. Appl. 53 (3–4) (2007) 625–646.
[13] W.L. Barth, L.V. Branets, G.F. Carey, Non-Newtonian flow in branched pipes and artery models, Int. J. Numer. Methods Fluids 57 (5) (2008) 531–553.
[14] M.G.H.M. Baltussen, Y.J. Choi, M.A. Hulsen, P.D. Anderson, Weakly-imposed Dirichlet boundary conditions for non-Newtonian fluid flow, J. Non-Newton. Fluid Mech. 166 (17–18) (2011) 993–1003.
[15] H. Lee, Numerical approximation of quasi-Newtonian flows by ALE-FEM, Numer. Methods Partial Differ. Equ. 28 (5) (2012) 1667–1695.
[16] H. Damanik, J. Hron, A. Ouazzi, S. Turek, Monolithic Newton-multigrid solution techniques for incompressible nonlinear flow models, Int. J. Numer. Methods Fluids 71 (2) (2013) 208–222.
[17] X. Zheng, G. Chen, X. Xie, A divergence-free weak Galerkin method for quasi-Newtonian Stokes flows, Sci. China Math. 60 (8) (2017) 1515–1528.
[18] R. Sevilla, A. Huerta, HDG-NEFEM with degree adaptivity for Stokes flows, J. Sci. Comput. 77 (3) (2018) 1953–1980.
[19] S. Congreve, P. Houston, E. Suli, T.P. Wihler, Discontinuous Galerkin finite element approximation of quasilinear elliptic boundary value problems II: strongly monotone quasi-Newtonian flows, IMA J. Numer. Anal. 33 (4) (2013) 1386–1415.
[20] S. Congreve, P. Houston, Two-grid *hp*-version discontinuous Galerkin finite element methods for quasi-Newtonian fluid flows, Int. J. Numer. Anal. Model. 11 (3) (2014) 496–524.
[21] G.N. Gatica, F.A. Sequeira, Analysis of an augmented HDG method for a class of quasi-Newtonian Stokes flows, J. Sci. Comput. 65 (3) (2015) 1270–1308.
[22] G.N. Gatica, F.A. Sequeira, A priori and a posteriori error analyses of an augmented HDG method for a class of quasi-Newtonian Stokes flows, J. Sci. Comput. 69 (3) (2016) 1192–1250.
[23] M. Giacomini, R. Sevilla, Discontinuous Galerkin approximations in computational mechanics: hybridization, exact geometry and degree adaptivity, SN Appl. Sci. 1 (9) (2019) 1047.
[24] R. Abgrall, M. Ricchiuto, High-order methods for CFD, in: Encyclopedia of Computational Mechanics, second edition, John Wiley & Sons, Chichester, UK, 2017, pp. 1–54.
[25] N. Fehn, M. Kronbichler, C. Lehrenfeld, G. Lube, P.W. Schroeder, High-order DG solvers for underresolved turbulent incompressible flows: a comparison of L^2 and $H(\text{div})$ methods, Int. J. Numer. Methods Fluids 91 (11) (2019) 533–556.
[26] G. Böhme, L. Rubart, Non-Newtonian flow analysis by finite elements, Fluid Dyn. Res. 5 (3) (1989) 147–158.
[27] G.F. Carey, K.C. Wang, W.D. Joubert, Performance of iterative methods for Newtonian and generalized Newtonian flows, Int. J. Numer. Methods Fluids 9 (2) (1989) 127–150.
[28] M. Franta, J. Málek, K.R. Rajagopal, On steady flows of fluids with pressure- and shear-dependent viscosities, Proc. Math. Phys. Eng. Sci. 461 (2055) (2005) 651–670.
[29] H. Sobhani, M. Razavi-Nouri, M.H.R. Ghoreishy, Investigation of combination of finite element formulation and element type on the accuracy of 3D modeling of polymeric fluid flow in an extrusion die, J. Appl. Polym. Sci. 120 (3) (2011) 1607–1615.
[30] F. Brezzi, J. Pitkäranta, On the stabilization of finite element approximations of the Stokes equations, in: W. Hackbusch (Ed.), Efficient Solutions of Elliptic Systems, Notes on Numerical Fluid Mechanics, 1984, pp. 11–19.
[31] S. Knauf, S. Frei, T. Richter, R. Rannacher, Towards a complete numerical description of lubricant film dynamics in ball bearings, Comput. Mech. 53 (2) (2014) 239–255.
[32] R. Codina, J. Blasco, A finite element formulation for the Stokes problem allowing equal velocity-pressure interpolation, Comput. Methods Appl. Mech. Eng. 143 (3–4) (1997) 373–391.
[33] C.R. Dohrmann, P.B. Bochev, A stabilized finite element method for the Stokes problem based on polynomial pressure projections, Int. J. Numer. Methods Fluids 46 (2) (2004) 183–201.
[34] D.R.Q. Pacheco, R. Schussnig, T.-P. Fries, An efficient split-step framework for non-Newtonian incompressible flow problems with consistent pressure boundary conditions, Comput. Methods Appl. Mech. Eng. 382 (2021) 113888.
[35] B. Hübner, D. Dinkler, A simultaneous solution procedure for strong interactions of generalized Newtonian fluids and viscoelastic solids at large strains, Int. J. Numer. Methods Eng. 64 (7) (2005) 920–939.
[36] A. Masud, J. Kwack, A stabilized mixed finite element method for the incompressible shear-rate dependent non-Newtonian fluids: variational multiscale framework and consistent linearization, Comput. Methods Appl. Mech. Eng. 200 (5–8) (2011) 577–596.
[37] V.L. Marrero, J.A. Tichy, O. Sahni, K.E. Jansen, Numerical study of purely viscous non-Newtonian flow in an abdominal aortic aneurysm, J. Biomech. Eng. 136 (10) (2014).
[38] L. Gesenhues, J.J. Camata, A.M.A. Côrtes, F.A. Rochinha, A.L.G.A. Coutinho, Finite element simulation of complex dense granular flows using a well-posed regularization of the μ(I)-rheology, Comput. Fluids 188 (2019) 102–113.
[39] F. Zinani, S. Frey, Finite element approximations for quasi-Newtonian flows employing a multi-field GLS method, Comput. Mech. 48 (2) (2011) 139–152.
[40] E. Castillo, R. Codina, Stabilized stress–velocity–pressure finite element formulations of the Navier–Stokes problem for fluids with non-linear viscosity, Comput. Methods Appl. Mech. Eng. 279 (2014) 554–578.
[41] D.R.Q. Pacheco, R. Schussnig, O. Steinbach, T.-P. Fries, A global residual-based stabilization for equal-order finite element approximations of incompressible flows, Int. J. Numer. Methods Eng. 122 (8) (2021) 2075–2094.

[42] S. Turek, A comparative study of time-stepping techniques for the incompressible Navier-Stokes equations: from fully implicit non-linear schemes to semi-implicit projection methods, Int. J. Numer. Methods Fluids 22 (1996) 987–1011.

[43] A. Smith, D. Silvester, Implicit algorithms and their linearization for the transient incompressible Navier-Stokes equations, IMA J. Numer. Anal. 17 (4) (1997) 527–545.

[44] J. Deteix, D. Yakoubi, Shear rate projection schemes for non-Newtonian fluids, Comput. Methods Appl. Mech. Eng. 354 (2019) 620–636.

[45] S. Turek, Efficient Solvers for Incompressible Flow Problems - an Algorithmic and Computational Approach, Lecture Notes in Computational Science and Engineering, vol. 6, Springer, Berlin Heidelberg, 1999.

[46] M. Benzi, Preconditioning techniques for large linear systems: a survey, J. Comput. Phys. 182 (2002) 418–477.

[47] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numer. 14 (2005) 1–137.

[48] H.C. Elman, D.J. Silvester, A.J. Wathen, Finite Elements and Fast Iterative Solvers, Oxford University Press, Oxford, 2014.

[49] S.V. Patankar, D.B. Spalding, A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows, Int. J. Heat Mass Transf. 15 (10) (1972) 1787–1806.

[50] C. Vuik, A. Saghir, G.P. Boerstoel, The Krylov accelerated SIMPLE(R) method for flow problems in industrial furnaces, Int. J. Numer. Methods Fluids 33 (7) (2000) 1027–1040.

[51] M. ur Rehman, C. Vuik, G. Segal, SIMPLE-type preconditioners for the Oseen problem, Int. J. Numer. Methods Fluids 61 (4) (2009) 432–452.

[52] D. Silvester, H. Elman, D. Kay, A. Wathen, Efficient preconditioning of the linearized Navier–Stokes equations for incompressible flow, J. Comput. Appl. Math. 128 (1–2) (2001) 261–279.

[53] D. Kay, D. Loghin, A. Wathen, A preconditioner for the steady-state Navier–Stokes equations, SIAM J. Sci. Comput. 24 (1) (2003) 237–256.

[54] H. Elman, V.E. Howle, J. Shadid, R. Shuttleworth, R. Tuminaro, Block preconditioners based on approximate commutators, SIAM J. Sci. Comput. 27 (5) (2006) 1651–1668.

[55] H. Elman, V.E. Howle, J. Shadid, D. Silvester, R. Tuminaro, Least squares preconditioners for stabilized discretizations of the Navier–Stokes equations, SIAM J. Sci. Comput. 30 (1) (2008) 290–311.

[56] M. Benzi, M.A. Olshanskii, An augmented Lagrangian-based approach to the Oseen problem, SIAM J. Sci. Comput. 28 (6) (2006) 2095–2113.

[57] M. Benzi, M.A. Olshanskii, Z. Wang, Modified augmented Lagrangian preconditioners for the incompressible Navier-Stokes equations, Int. J. Numer. Methods Fluids 66 (4) (2011) 486–508.

[58] T. Heister, G. Rapin, Efficient augmented Lagrangian-type preconditioning for the Oseen problem using Grad-Div stabilization, Int. J. Numer. Methods Fluids 71 (1) (2013) 118–134.

[59] H.C. Elman, Preconditioning strategies for models of incompressible flow, J. Sci. Comput. 25 (1) (2005) 347–366.

[60] A.C. de Niet, F.W. Wubs, Two preconditioners for saddle point problems in fluid flows, Int. J. Numer. Methods Fluids 54 (4) (2007) 355–377.

[61] M. ur Rehman, C. Vuik, G. Segal, A comparison of preconditioners for incompressible Navier-Stokes solvers, Int. J. Numer. Methods Fluids 57 (12) (2008) 1731–1751.

[62] J. Cahouet, J.-P. Chabard, Some fast 3D finite element solvers for the generalized Stokes problem, Int. J. Numer. Methods Fluids 8 (8) (1988) 869–895.

[63] X. He, M. Neytcheva, C. Vuik, On preconditioning of incompressible non-Newtonian flow problems, J. Comput. Math. 33 (1) (2015) 33–58.

[64] M. Kronbichler, A. Diagne, H. Holmgren, A fast massively parallel two-phase flow solver for microfluidic chip simulation, Int. J. High Perform. Comput. Appl. 32 (2) (2018) 266–287.

[65] P.M. Gresho, D.F. Griffiths, D.J. Silvester, Adaptive time-stepping for incompressible flow part I: scalar advection-diffusion, SIAM J. Sci. Comput. 30 (4) (2008) 2018–2054.

[66] D.A. Kay, P.M. Gresho, D.F. Griffiths, D.J. Silvester, Adaptive time-stepping for incompressible flow part II: Navier–Stokes equations, SIAM J. Sci. Comput. 32 (1) (2010) 111–128.

[67] H. Elman, M. Mihajlović, D. Silvester, Fast iterative solvers for buoyancy driven flow problems, J. Comput. Phys. 230 (10) (2011) 3900–3914.

[68] Y.I. Cho, K.R. Kensey, Effects of the non-Newtonian viscosity of blood on flows in a diseased arterial vessel. Part 1: steady flows, Biorheology 28 (3–4) (1991) 241–262.

[69] D.R.Q. Pacheco, T.S. Müller, O. Steinbach, G. Brenn, On outflow boundary conditions in finite element simulations of non-Newtonian internal flows, Int. J. Comput. Vis. Sci. Eng. (2021), https://doi.org/10.51375/IJCVSE.2021.1.6.

[70] J. Donea, A. Huerta, Finite Element Methods for Flow Problems, John Wiley & Sons, Chichester, 2003.

[71] V. John, Finite Element Methods for Incompressible Flow Problems, Springer International Publishing, Basel, 2016.

[72] J.-J. Droux, T.J.R. Hughes, A boundary integral modification of the Galerkin least squares formulation for the Stokes problem, Comput. Methods Appl. Mech. Eng. 113 (1–2) (1994) 173–182.

[73] K.E. Jansen, S.S. Collis, C. Whiting, F. Shaki, A better consistency for low-order stabilized finite element methods, Comput. Methods Appl. Mech. Eng. 174 (1–2) (1999) 153–170.

[74] P. Bochev, M. Gunzburger, An absolutely stable pressure-Poisson stabilized finite element method for the Stokes equations, SIAM J. Numer. Anal. 42 (3) (2004) 1189–1207.

[75] J. Hinz, J. Helmig, M. Möller, S. Elgeti, Boundary-conforming finite element methods for twin-screw extruders using spline-based parameterization techniques, Comput. Methods Appl. Mech. Eng. 361 (2020) 112740.

[76] J.G. Heywood, R. Rannacher, S. Turek, Artificial boundaries and flux and pressure conditions for the incompressible Navier-Stokes equations, Int. J. Numer. Methods Fluids 22 (5) (1996) 325–352.

[77] G. Arbia, I.E. Vignon-Clementel, T.-Y. Hsia, J.-F. Gerbeau, Modified Navier–Stokes equations for the outflow boundary conditions in hemodynamics, Eur. J. Mech. B, Fluids 60 (2016) 175–188.

[78] D.R.Q. Pacheco, O. Steinbach, A continuous finite element framework for the pressure Poisson equation allowing non-Newtonian and compressible flow behavior, Int. J. Numer. Methods Fluids 93 (2021) 1435–1445.

[79] J. Rang, Pressure corrected implicit θ-schemes for the incompressible Navier-Stokes equations, Appl. Math. Comput. 201 (1–2) (2008) 747–761.

[80] D. Meidner, T. Richter, A posteriori error estimation for the fractional step theta discretization of the incompressible Navier-Stokes equations, Comput. Methods Appl. Mech. Eng. 288 (2015) 45–59.

[81] L. Failer, T. Wick, Adaptive time-step control for nonlinear fluid–structure interaction, J. Comput. Phys. 366 (2018) 448–477.

[82] J.C. Simo, F. Armero, Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier-Stokes and Euler equations, Comput. Methods Appl. Mech. Eng. 111 (1–2) (1994) 111–154.

[83] R. Ingram, A new linearly extrapolated Crank-Nicolson time-stepping scheme for the Navier-Stokes equations, Math. Comput. 82 (284) (2013) 1953–1973.

[84] M. Olshanskii, G. Lube, T. Heister, J. Löwe, Grad-div stabilization and subgrid pressure models for the incompressible Navier-Stokes equations, Comput. Methods Appl. Mech. Eng. 198 (49–52) (2009) 3975–3988.

[85] J.G. Heywood, R. Rannacher, Finite-element approximation of the nonstationary Navier–Stokes problem. Part IV: error analysis for second-order time discretization, SIAM J. Numer. Anal. 27 (2) (1990) 353–384.

[86] W. Dettmer, D. Perić, An analysis of the time integration algorithms for the finite element solutions of incompressible Navier-Stokes equations based on a stabilised formulation, Comput. Methods Appl. Mech. Eng. 192 (9–10) (2003) 1177–1226.

[87] M. Luskin, R. Rannacher, W. Wendland, On the smoothing property of the Crank-Nicolson scheme, Appl. Anal. 14 (2) (1982) 117–135.

[88] R. Rannacher, Finite element solution of diffusion problems with irregular data, Numer. Math. 43 (2) (1984) 309–327.

[89] B. Lindberg, On smoothing and extrapolation for the trapezoidal rule, BIT Numer. Math. 11 (1) (1971) 29–52.

[90] O. Østerby, Five ways of reducing the Crank–Nicolson oscillations, BIT Numer. Math. 43 (4) (2003) 811–822.

[91] W.A. Wall, Fluid-Struktur-Interaktion mit stabilisierten Finiten Elementen, PhD thesis, 1999.

[92] V. John, J. Rang, Adaptive time step control for the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 199 (9) (2010) 514–524.

[93] K. Gustafsson, M. Lundh, G. Söderlind, A PI stepsize control for the numerical solution of ordinary differential equations, BIT Numer. Math. 28 (2) (1988) 270–287.

[94] V. John, On the efficiency of linearization schemes and coupled multigrid methods in the simulation of a 3D flow around a cylinder, Int. J. Numer. Methods Fluids 50 (7) (2006) 845–862.

[95] B.M. Irons, R.C. Tuck, A version of the Aitken accelerator for computer iteration, Int. J. Numer. Methods Eng. 1 (3) (1969) 275–277.

[96] V.M. Verzhbitskiĭ, I.F. Yumanova, On the quadratic convergence of the Aitken Δ^2 process, Comput. Math. Math. Phys. 51 (10) (2011) 1659–1663.

[97] U. Küttler, W.A. Wall, Fixed-point fluid-structure interaction solvers with dynamic relaxation, Comput. Mech. 43 (1) (2008) 61–72.

[98] D. Braess, Finite Elements: Theory, Fast Solvers, and Applications in Solid Mechanics, Cambridge University Press, Cambridge, 2007.

[99] V.E. Henson, U.M. Yang, BoomerAMG: a parallel algebraic multigrid solver and preconditioner, Appl. Numer. Math. 41 (1) (2002) 155–177.

[100] Y. Saad, M.H. Schultz GMRES, A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. 7 (3) (1986) 856–869.

[101] H.C. Elman, R.S. Tuminaro, Boundary conditions in approximate commutator preconditioners for the Navier-Stokes equations, Electron. Trans. Numer. Anal. 35 (2009) 257–280.

[102] The deal.II library, version 9.0, J. Numer. Math. 26 (4) (2018) 173–183.

[103] P.R. Amestoy, I.S. Duff, J.Y. L'Excellent, J. Koster, A fully asynchronous multifrontal solver using distributed dynamic scheduling, SIAM J. Matrix Anal. Appl. 23 (1) (2001) 15–41.

[104] E. Burman, P. Hansbo, Edge stabilization for the generalized Stokes problem: a continuous interior penalty method, Comput. Methods Appl. Mech. Eng. 195 (19–22) (2006) 2393–2410.

[105] H.W. Choi, A.I. Barakat, Numerical study of the impact of non-Newtonian blood behavior on flow over a two-dimensional backward facing step, Biorheology 42 (6) (2005) 493–509.

[106] V. John, G. Matthies, J. Rang, A comparison of time-discretization/linearization approaches for the incompressible Navier-Stokes equations, Comput. Methods Appl. Mech. Eng. 195 (44–47) (2006) 5995–6010.

[107] L. Jay, Convergence of a class of Runge-Kutta methods for differential-algebraic systems of index 2, BIT Numer. Math. 33 (1) (1993) 137–150.

[108] A. Quarteroni, A. Manzoni, C. Vergara, The cardiovascular system: mathematical modelling, numerical algorithms and clinical applications, Acta Numer. 26 (2017) 365–590.

[109] M. Thiriet, Biology and Mechanics of Blood Flows, Springer, New York, 2008.

[110] A.M. Gambaruto, J. Janela, A. Moura, A. Sequeira, Sensitivity of hemodynamics in a patient specific cerebral aneurysm to vascular geometry and blood rheology, Math. Biosci. Eng. 8 (2) (2011) 409–423.