A damped Newton algorithm for computing viscoplastic fluid flows

Pierre Saramito

▶ **To cite this version:**

Pierre Saramito. A damped Newton algorithm for computing viscoplastic fluid flows. 2015. hal-01228347v2

HAL Id: hal-01228347

https://hal.science/hal-01228347v2

Preprint submitted on 5 Feb 2016 (v2), last revised 11 May 2016 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A damped Newton algorithm for computing viscoplastic fluid flows

Pierre Saramito

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

mail: Pierre.Saramito@imag.fr

Abstract – For the first time, a Newton method is proposed for the unregularized viscoplastic fluid flow problem. It leads to a superlinear convergence for Herschel-Bulkley fluids when $0 < n < 1$, where n is the power law index. Performances are enhanced by using the inexact variant of the Newton method and, for solving the Jacobian system, by using an efficient preconditioner based on the regularized problem. A demonstration is provided by computing a viscoplastic flow in a pipe with a square cross section. Comparisons with the augmented Lagrangian algorithm show a dramatic reduction of the required computing time while this new algorithm provides an equivalent accuracy for the prediction of the yield surfaces.

Keywords – viscoplastic fluid ; Bingham model ; Herschel-Bulkley model ; Newton method

Introduction

The numerical resolution of viscoplastic fluid flows is still a challenging task. The augmented Lagrangian method has been introduced in 1969 by Hestenes [20] and Powell [29]. During the 1970s, this approach became popular for solving optimization problems (see e.g. Rockafellar [33]). In 1980, Glowinski [17] and then Fortin and Glowinski [15] proposed to apply it to the solution of the linear Stokes problem and also to others non-linear problems such as Bingham fluid flows. In 1980, Bercovier and Engelman [3] proposed a viscosity function for the regularization of Bingham flow problem. In 1987, another viscosity function was proposed by Papanastasiou [28]. During the 1980s and the 1990s, numerical computations for Bingham flow problems was dominated by the regularization method, perhaps due to its simplicity, while the augmented Lagrangian algorithm leaded not yet convincing results for viscoplastic flow applications. In 1989, Glowinski and le Tallec [18] revisited the augmented Lagrangian method, using new optimization and convex analysis tools, such as subdifferential, but no evidence of the efficiency of this approach to viscoplasticity was showed, while regularization approach becomes more popular in the 1990s with the work of Mitsoulis *et al.* [21] and Wilson and Taylor [44]. In 2001, Saramito and Roquet [41, 35] showed for the first time the efficiency of the augmented Lagrangian algorithm when combined with auto-adaptive mesh methods for capturing accurately the yield surface. In the 2000s, this approach became mature and a healthy competition developed between the regularization approach and the augmented Lagrangian one. Vola, Boscardin and Latché [43] obtained results for a driven cavity flow with the augmented Lagrangian algorithm while Mitsoulis *et al.* [22] presented computation for an expansion flow with regularization and Frigaard *et al.* [23, 16, 31] pointed out some drawbacks of the regularization approach. Finally, at the end of the 2000s decade, the augmented Lagrangian algorithm becomes the most popular way to solve viscoplastic flow problems [24, 12]

because of its accuracy, despite the regularization approach runs much more faster. The free software `Rheolef` library, developed by the author and supporting both the augmented Lagrangian algorithm and an auto-adaptive mesh technique is now widely used for various flow applications (see e.g. [30, 36, 37, 4]).

The main drawback of the augmented Lagrangian algorithm is its computing time for large applications, especially when the Bingham number becomes large. This paper is a contribution to an ongoing effort for the development of faster algorithms for the resolution of the unregularized viscoplastic model. One of the most efficient algorithm to solve nonlinear problems is the Newton method, due to its super-linear convergence properties (see e.g. [26]). This approach has already been investigated for the regularized approach of the viscoplastic problem (see e.g. [5] and most recently [9, 10, 11] for the biviscous regularization). Applying the Newton method to the unregularized viscoplastic problem leads to a singular Jacobian matrix. This difficulty has been recently addressed by using the trusted region algorithm [42], that regularizes the Jacobian matrix but loses the superlinear convergence of the method. In this paper, our contribution is to address directly the singularity of the Jacobian matrix in the Newton method in order to preserve the superlinear convergence. The proposed reformulation of the viscoplastic flow problem is inspired by the work of Alart and Curnier [1] on another non-smooth problem, the frictional contact one, that was successfully addressed by a Newton method.

Section 1 presents the viscoplastic flow problem and its mathematical statement. This problem is reformulated in section 2 in terms of a projection operator and section 3 presents its variational formulation and discretization. Section 4 develops the Newton algorithm and the resolution of the Jacobian matrix while section 5 shows preliminary results for this approach. The paper included two appendices, dealing respectively with the proof of equivalence for the present reformulation with a projection and with the spectral study of a preconditioner.

1 Problem statement

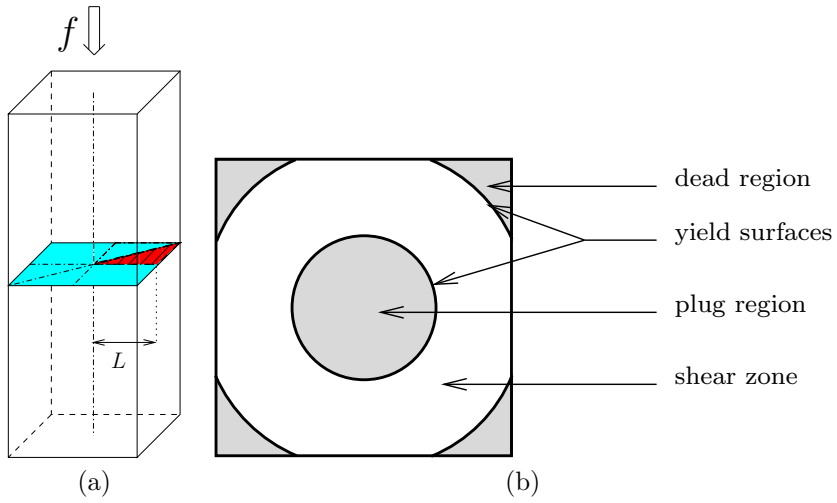

Figure 1: Square tube cross-section: (a) tridimensional view; (b) schematic view of the cross-section.

The fully developed flow of a Herschel-Bulkley fluid [19] in a prismatic tube, as shown on Fig. 1.a (see also [41]). Let (Oz) be the axis of the tube and (Oxy) the plane of the bounded section $\Omega \subset I\!R^2$. The pressure gradient is written as $\nabla p = (0,\, 0,\, -f)$ in Ω, where $f > 0$ is the constant applied force density. The velocity is written as $\boldsymbol{u} = (0,\, 0,\, u)$, where the third component u along

the (Oz) axis depends only upon x and y, and is independent of t and z. The problem can be considered as a two-dimensional one, and the stress tensor σ is equivalent to a two shear stress component vector: $\boldsymbol{\sigma} = (\sigma_{xz}, \sigma_{yz})$. We also use the following notations:

$$\begin{aligned} \nabla u &= \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right) \\ \operatorname{div} \boldsymbol{\sigma} &= \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} \\ |\boldsymbol{\sigma}| &= \sqrt{\sigma_{xz}^2 + \sigma_{yz}^2} \end{aligned}$$

In the case of a square cross-section using three symmetries with respect to the (Ox), (Oy) and the first bisector, the domain of computation Ω can be reduced to a triangular shaped domain (see Fig. 1.a). The problem can be summarized as:

(P): *find* $\boldsymbol{\sigma}$ *and* u *defined in* Ω *such that*:

$$\boldsymbol{\sigma} = K|\nabla u|^{n-1}\nabla u + \sigma_0 \frac{\nabla u}{|\nabla u|} \text{ when } \nabla u \neq 0 \tag{1a}$$

$$|\boldsymbol{\sigma}| \leqslant \sigma_0 \quad \text{when } \nabla u = 0 \tag{1b}$$

$$\operatorname{div} \boldsymbol{\sigma} = -f \text{ in } \Omega \tag{1c}$$

$$u = 0 \text{ on } \partial\Omega \tag{1d}$$

where $\sigma_0 \geqslant 0$ is the yield stress, $K > 0$ is the consistency and $n > 0$ is the power index. Notice that when $\sigma_0 = 0$ and $n = 1$, one is led to the classical viscous incompressible fluid. When $\sigma_0 > 0$, rigid zones in the interior of the fluid can be observed. As σ_0 becomes larger, these rigid zones develop and may completely block the flow when σ_0 is sufficiently large. When $n = 1$ and $\sigma_0 > 0$ the model is called the Bingham one [6, 25] while for a general power law index $n > 0$ this is the Herschel-Bulkley model. Here, (1a)-(1b) express the constitutive equation, (1c) the conservation of momentum and (1d) the no-slip boundary condition. In the case of a square cross-section, we reduce the domain of computation by using symmetries (see Fig. 1.a).

Let L be a characteristic length of the cross-section Ω, i.e. the half-length of an edge of a square section, or the radius of a circular section (also denoted by R for convenience in that case). A characteristic stress is given by $\Sigma = Lf/2$ and a characteristic velocity U is such that $\Sigma = K(U/L)^n$ i.e. $U = (Lf/(2K))^{1/n}L$. The Bingham dimensionless number is defined by the ratio of the yield stress σ_0 by the representative viscous stress Σ:

$$Bi = \frac{2\sigma_0}{L\,f}$$

The Bingham number Bi and the power law index n are the only two dimensionless numbers of the problem.

2 Reformulation of the problem

Problem (1a)-(1d) is equivalent to

$(HB)_0$: *find* $\boldsymbol{\sigma}$ *and* u *defined in* Ω *such that*:

$$\nabla u = P_0(\boldsymbol{\sigma}) \tag{2a}$$

$$\operatorname{div} \boldsymbol{\sigma} = -f \text{ in } \Omega \tag{2b}$$

$$u = 0 \text{ on } \partial\Omega \tag{2c}$$

where P_0 denotes the following projection operator, defined for all $\boldsymbol{\tau} \in \mathbb{R}^2$ by

$$P_0(\boldsymbol{\tau}) = \begin{cases} \dfrac{1}{K^{1/n}} \left(|\boldsymbol{\tau}| - \sigma_0\right)^{1/n} \dfrac{\boldsymbol{\tau}}{|\boldsymbol{\tau}|} & \text{when } |\boldsymbol{\tau}| > \sigma_0 \\ \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The proof of this equivalence is postponed in appendix A, property 1. Observe that, for all $\boldsymbol{\sigma}$ and $\boldsymbol{\gamma} \in \mathbb{R}^2$ and for all $\sigma_0 \geqslant 0$ and $n > 0$, the projection operator P_0 introduced in (3) satisfies, for all $r \geqslant 0$, the following property:

$$\boldsymbol{\gamma} = P_0(\boldsymbol{\sigma}) \Longleftrightarrow \boldsymbol{\gamma} = P_r(\boldsymbol{\sigma} + r\boldsymbol{\gamma}) \tag{4}$$

where P_r denotes the following extended projection operator, defined for all $\boldsymbol{\tau} \in \mathbb{R}^2$ by

$$P_r(\boldsymbol{\tau}) = \begin{cases} \varphi_r^{-1}(|\boldsymbol{\tau}|) \dfrac{\boldsymbol{\tau}}{|\boldsymbol{\tau}|} & \text{when } |\boldsymbol{\tau}| > \sigma_0 \\ \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

and where φ_r is defined for all $\dot{\gamma} \geqslant 0$ by

$$\varphi_r(\dot{\gamma}) = \sigma_0 + K\dot{\gamma}^n + r\dot{\gamma}$$

The proof of equivalence (4) is postponed in appendix A, property 2. Remark that the r parameter, involved in the extended projection operator P_r, interprets as an augmentation parameter, similar to those involved by the augmented Lagrangian formulation.

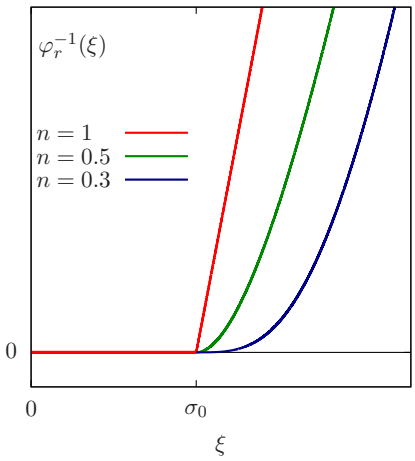

Figure 2: The φ_r^{-1} function for various values of n.

As the function φ_r is strictly increasing in $[0, +\infty[$, it is invertible from $[0, +\infty[$ to $[\sigma_0, +\infty[$ and its inverse denoted by φ_r^{-1} is well defined in $[\sigma_0, +\infty[$ (see Fig. 2). Notice that when $r = 0$ we have $\varphi_0^{-1}(\xi) = K^{-1/n}(\xi - \sigma_0)^{1/n}$ for all $\xi \geqslant \sigma_0$ while there is no such closed form for φ_r^{-1} when $r > 0$.

Let us perform a change of unknown by introducing

$$\boldsymbol{\beta} = \boldsymbol{\sigma} + r\boldsymbol{\nabla} u \Longleftrightarrow \boldsymbol{\sigma} = \boldsymbol{\beta} - r\boldsymbol{\nabla} u$$

Then, problem (2a)-(2c) writes equivalently as

$(HB)_r$: *find* u *and* $\boldsymbol{\beta}$ *such that*

$$\begin{aligned} r\Delta u - \operatorname{div}\boldsymbol{\beta} &= f \text{ in } \Omega & \text{(6a)}\\ \boldsymbol{\nabla} u - P_r(\boldsymbol{\beta}) &= 0 \text{ in } \Omega & \text{(6b)}\\ u &= 0 \text{ on } \partial\Omega & \text{(6c)} \end{aligned}$$

Observe that, from (4), the solution is independent upon $r \geqslant 0$ while, when $r = 0$, this problem reduces to (2a)-(2c). The proof of the equivalence between this problem and (2a)-(2c) is postponed in appendix A, property 3.

3 Variational formulation and discretization

Consider the following forms:

$$\begin{aligned} a(u,v) &= -r\int_\Omega \boldsymbol{\nabla} u\,.\boldsymbol{\nabla} v\,\mathrm{d}x, \quad \forall u,\, v \in H^1(\Omega)\\ b(v,\boldsymbol{\tau}) &= \int_\Omega \boldsymbol{\nabla} v\,.\boldsymbol{\tau}\,\mathrm{d}x, \quad \forall \boldsymbol{\tau} \in \left(L^2(\Omega)\right)^2, \quad \forall v \in H^1(\Omega)\\ c(\boldsymbol{\beta},\boldsymbol{\tau}) &= \int_\Omega P_r(\boldsymbol{\beta}).\boldsymbol{\tau}\,\mathrm{d}x, \quad \forall \boldsymbol{\beta},\boldsymbol{\tau} \in \left(L^2(\Omega)\right)^2\\ \ell(v) &= \int_\Omega f\,v\,\mathrm{d}x, \quad \forall v \in L^2(\Omega) \end{aligned}$$

The variational formulation writes:

$(FV)_r$: *find* $u \in H_0^1(\Omega)$ *and* $\boldsymbol{\beta} \in \left(L^2(\Omega)\right)^2$ *such that*

$$\begin{aligned} a(u,v) + b(v,\boldsymbol{\beta}) &= \ell(v), \quad \forall v \in H_0^1(\Omega)\\ b(u,\boldsymbol{\tau}) - c(\boldsymbol{\beta},\boldsymbol{\tau}) &= 0, \quad \forall \boldsymbol{\tau} \in \left(L^2(\Omega)\right)^2 \end{aligned}$$

This problem is then discretized by using the mixed finite element method introduced in [41]: the velocity u is approximated by continuous piecewise polynomials of order $k \geqslant 1$ while the shear stress vector $\boldsymbol{\beta}$ is approximated by piecewise discontinuous $k-1$ polynomials. As the corresponding approximated problem is similar to the continuous one, it is not developed here. For the purpose of simplicity, we also continue to work with the continuous problem in the next section, dedicated to the Newton method.

4 Newton method

The problem can be expressed in a compact form:

find $u \in H_0^1(\Omega)$ *and* $\boldsymbol{\beta} \in \left(L^2(\Omega)\right)^2$ *such that*

$$F(u,\boldsymbol{\beta}) = 0$$

where F is defined in variational form for all $v \in H^1(\Omega)$ and $\boldsymbol{\tau} \in \left(L^2(\Omega)\right)^2$ by

$$\langle F(u,\boldsymbol{\beta}),(v,\boldsymbol{\tau})\rangle = a(u,v) + b(\boldsymbol{\beta},v) + b(\boldsymbol{\tau},u) - c(\boldsymbol{\beta},\boldsymbol{\tau})$$

and where $\langle .,. \rangle$ denotes the duality product induced by the L^2 pivot space i.e. $\langle \varphi,\phi\rangle = \int_\Omega \varphi\phi\,\mathrm{d}x$ for all φ,ϕ defined in Ω.

We will see at the end of this section that F is differentiable if and only if $n < 1$. In the general case $n > 0$, we are able to write a *non-smooth* version of the Newton method [32, p. 358] that involves the *subdifferential* ∂F as a generalized gradient [7] of F. This method defines the sequence $(u_m,\boldsymbol{\beta}_m)_{m\geqslant 0}$ by recurrence as:

- $m = 0$: let $(u_0, \boldsymbol{\beta}_0) \in H_0^1(\Omega) \times \left(L^2(\Omega)\right)^2$ being given.
- $m \geqslant 0$: let $\left(u_{m-1}, \boldsymbol{\beta}_{m-1}\right) \in H_0^1(\Omega) \times \left(L^2(\Omega)\right)^2$ being known.
  Find $(\delta u, \delta\boldsymbol{\beta}) \in H_0^1(\Omega) \times \left(L^2(\Omega)\right)^2$ such that

$$\mathscr{A}_0 \,.\, (\delta u, \delta\boldsymbol{\beta}) = -F\left(u_{m-1}, \boldsymbol{\beta}_{m-1}\right)$$

  where $\mathscr{A}_0 \in \partial F\left(u_{m-1}, \boldsymbol{\beta}_{m-1}\right)$ is an arbitrarily element of the subdifferential. Then defines

$$u_m = u_{m-1} + \delta u \quad \text{and} \quad \boldsymbol{\beta}_m = \boldsymbol{\beta}_{m-1} + \delta\boldsymbol{\beta}$$

Notice that when F is differentiable, then its subdifferential ∂F contains only one element F' and the method coincides with the usual Newton method for smooth functions. Otherwise, the choice of any element $\mathscr{A}_0$ of the subdifferential as a Jacobian is arbitrarily. At each step $m \geqslant 0$, this algorithm solves a linear subproblem involving a Jacobian $\mathscr{A}_0$. The Newton method has only local convergence properties, i.e. the initial value should be close enough to the solution. In order to circumvent this limitation, a globalized Newton variant is used here. It is based on a damping strategy, as described and implemented in the `Rheolef` free software FEM library [39]. This damping strategy is presented in details in the Rheolef user's manual, volume 1 [39], section 8.4, and the corresponding source code is fully available under the GNU Public License.

The subdifferential ∂F is defined, for all $\delta u \in H^1(\Omega)$ and $\delta\boldsymbol{\beta} \in \left(L^2(\Omega)\right)^2$ and $v \in H_0^1(\Omega)$ and $\boldsymbol{\tau} \in \left(L^2(\Omega)\right)^2$ by

$$\langle \partial F(u, \boldsymbol{\beta}).(\delta u, \delta\boldsymbol{\beta}), (v, \boldsymbol{\tau})\rangle = a(\delta u, v) + b(\delta\boldsymbol{\beta}, v) + b(\boldsymbol{\tau}, \delta u) - c_1(\boldsymbol{\beta};\, \delta\boldsymbol{\beta}, \boldsymbol{\tau})$$

where c_1 denotes the following form:

$$c_1(\boldsymbol{\beta};\, \delta\boldsymbol{\beta}, \boldsymbol{\tau}) \quad = \quad \int_\Omega \left(\partial P_r(\boldsymbol{\beta})\delta\boldsymbol{\beta}\right) .\boldsymbol{\tau} \,\mathrm{d}x$$

and, for all $\boldsymbol{\beta} \in \mathbb{R}^2$, we denote by $\partial P_r(\boldsymbol{\beta})$ the following subdifferential of P_r, which is 2×2 matrix valued:

$$\partial P_r(\boldsymbol{\beta}) = \begin{cases} \left\{ \dfrac{\varphi_r^{-1}(|\boldsymbol{\beta}|)}{|\boldsymbol{\beta}|}\boldsymbol{I} + \dfrac{\mu\,|\boldsymbol{\beta}| - \varphi_r^{-1}(|\boldsymbol{\beta}|)}{|\boldsymbol{\beta}|^3}\boldsymbol{\beta}\otimes\boldsymbol{\beta}, \quad \forall \mu \in \partial\left(\varphi_r^{-1}\right)(|\boldsymbol{\beta}|) \right\} & \text{when } |\boldsymbol{\beta}| \geqslant \sigma_0 \\ \{0\} & \text{otherwise} \end{cases}$$

Here, $\partial\left(\varphi_r^{-1}\right)$ denotes the subdifferential of the inverse of the function φ_r, defined for all $\xi \geqslant 0$ by

$$\partial\left(\varphi_r^{-1}\right)(\xi) \quad = \quad \begin{cases} \left\{ \dfrac{1}{\varphi_r'\left(\varphi_r^{-1}(\xi)\right)} = \dfrac{1}{r + nK\left(\varphi_r^{-1}(\xi)\right)^{-1+n}} \right\} & \text{when } \xi > \sigma_0 \\ \left[0,\ \dfrac{1}{\varphi_r'(0)}\right] & \text{when } \xi = \sigma_0 \\ \{0\} & \text{otherwise} \end{cases}$$

Remark that when $\varphi_r'(0) = +\infty$ i.e. $n < 1$ then φ_r^{-1} is differentiable in zero (see also Fig. 2) and then both P_r and F are differentiable. Otherwise, $\varphi_r'(0) = nK + r$ when $n = 1$ and $\varphi_r'(0) = r$ when $n > 1$: in these two cases $\partial\left(\varphi_r^{-1}\right)(\sigma_0)$ is a multi-valued set. For a practical implementation, as the element of the subdifferential is arbitrarily chosen for the non-smooth Newton method, it is sufficient to choose $0 \in \partial\left(\varphi_r^{-1}\right)(\xi)$ when $\xi = \sigma_0$. This choice is convenient for any $n > 0$.

The Jacobian matrix $\mathscr{A}_0 \in \partial F$ associated to the linear problem satisfied by $(\delta u, \delta\boldsymbol{\beta})$ expresses as:

$$\mathscr{A}_0 = \begin{pmatrix} A & B^T \\ B & -C_0 \end{pmatrix}$$

were A, B and C_0 denotes the operators associated to the bilinear forms $a(.,.)$, $b(.,.)$ and $c_1(\boldsymbol{\beta}_{m-1};.,.)$ respectively. Notice that A is linear symmetric definite negative while C_0 is symmetric semi-definite positive. Recall that we expect the existence of unyielded regions of Ω with nonzero measure where $\boldsymbol{\nabla} u = 0$ and $|\boldsymbol{\beta}| < \sigma_0$ (see Fig. 1). In these regions, the only 2×2 matrix that belongs to $\partial P_r(\boldsymbol{\beta})$ is zero. Thus, the Jacobian $\mathscr{A}_0 \in \partial F$ is not expected to be invertible in general. Moreover, as A is negative, the Jacobian $\mathscr{A}_0$ is indefinite (i.e. it has both positive and negative eigenvalues). As $\mathscr{A}_0$ is singular, there exists an infinity of solutions: it is sufficient to choose one of them for the damped Newton method to converge. To this purpose, we use Saad and Schultz' GMRES algorithm [38] for solving this indefinite and possibly singular linear system. It can be considered as a generalization of Paige and Saunders' MINRES algorithm [27] which applies more specifically to symmetric and definite systems, possibly singular. This algorithm is implemented in the `Rheolef` free software FEM library [39] together with the damped Newton method.

The convergence rate of the GMRES algorithm can be dramatically increased by supplying a preconditioner $\tilde{\mathscr{A}}$, i.e. another invertible matrix, easier to invert, and close to the Jacobian matrix $\mathscr{A}_0 \in \partial F(\chi)$ where $\chi = (u, \boldsymbol{\beta})$. Let the linear system writes $\mathscr{A}_0\ \delta\chi = -F(\chi)$ where $\delta\chi = (\delta u, \delta\boldsymbol{\beta})$ and r denotes the residual terms at the right-hand-side. It writes equivalently

$$\tilde{\mathscr{A}}^{-1}\mathscr{A}_0\ \delta\chi = -\tilde{\mathscr{A}}^{-1}F(\chi) \tag{7}$$

As soon as $\tilde{\mathscr{A}}$ is invertible, applying $\tilde{\mathscr{A}}^{-1}$ to both the left and right hand sides of the linear system do not change its solution. Here are some extreme choices for the preconditioner:

- When $\tilde{\mathscr{A}} = \mathscr{A}_0$ then we have the perfect preconditioner: the linear system is solved in one iteration. The drawback is that it is not easiest to apply $\tilde{\mathscr{A}}^{-1}$ to a vector than to solve the initial linear system.
- Conversely, when $\tilde{\mathscr{A}}$ is the identity, there is no preconditioning at all.

The idea is to find some $\tilde{\mathscr{A}}$ between $\mathscr{A}_0$ and the identity, between the perfect and the do-nothing preconditioner. A good preconditioner is closest as possible to $\mathscr{A}_0$, and in such a way that applying $\tilde{\mathscr{A}}^{-1}$ to a vector is easier than solving the initial linear system.

In this paper, we consider a preconditioner which is based on the Jacobian of the regularized problem. A similar idea was suggested by Aposporidis *et al.* [2] in the context of a fixed point algorithm and with a different reformulation of the viscoplastic flow problem. The Jacobian of the regularized problem is invertible, thus easier to invert than the unregularized one. Also, as shown below, when the regularization parameter tends to zero, the Jacobian of the regularized problem becomes close to those of the unregularized one, thus increasing the convergence rate of the GMRES algorithm for solving the unregularized problem. For all $\varepsilon > 0$, let us introduce the regularized function $\varphi_{r,\varepsilon}$ is defined for all $\dot{\gamma} \in \mathbb{R}^+$ by

$$\varphi_{r,\varepsilon}(\dot{\gamma}) = r\dot{\gamma} + K\dot{\gamma}^n + \frac{\sigma_0\,\dot{\gamma}}{(\dot{\gamma}^2 + \varepsilon^2)^{\frac{1}{2}}}$$

This is a Bercovier and Engelman [3] style of regularization for the last term of the right-hand-side. The regularized projection operator is then defined for all $\boldsymbol{\tau} \in \mathbb{R}^2$ by

$$P_{r,\varepsilon}(\boldsymbol{\tau}) = \begin{cases} \varphi_{r,\varepsilon}^{-1}(|\boldsymbol{\tau}|)\dfrac{\boldsymbol{\tau}}{|\boldsymbol{\tau}|} & \text{when } \boldsymbol{\tau} \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

As $\varphi_{r,\varepsilon}$ is strictly increasing from $]0,+\infty[$ to $]0,+\infty[$, its inverse $\varphi_{r,\varepsilon}^{-1}$ is well defined. The regularized problem writes:

$(HB)_{r,\varepsilon}$: find u and $\boldsymbol{\beta}$, defined in Ω, such that

$$
\begin{aligned}
r\Delta u - \operatorname{div}\boldsymbol{\beta} &= f \quad \text{in } \Omega && \text{(8a)}\\
\boldsymbol{\nabla} u - P_{r,\varepsilon}(\boldsymbol{\beta}) &= 0 \quad \text{in } \Omega && \text{(8b)}\\
u &= 0 \quad \text{on } \partial\Omega && \text{(8c)}
\end{aligned}
$$

This problem is differentiable and its Jacobian $\mathscr{A}_\varepsilon = F'_\varepsilon$ has an expression which is similar to those of the unregularized one, just replacing φ_r by $\varphi_{r,\varepsilon}$. For practical computations, the evaluations of φ_r^{-1} and $\varphi_{r,\varepsilon}^{-1}$ are also performed by a Newton method with a stopping criterion at the machine precision (about 10^{-15} in double precision). This criterion is reached in very few iterations, as both φ_r and $\varphi_{r,\varepsilon}$ are regular. The Jacobian matrix $\mathscr{A}_\varepsilon = F'_\varepsilon$

$$
\mathscr{A}_\varepsilon = \begin{pmatrix} A & B^T \\ B & -C_\varepsilon \end{pmatrix}
$$

Notice that C_ε is symmetric definite positive. It is also block diagonal, since the stress components are approximated by piecewise discontinuous $k-1$ degree polynomials. For instance, when $k=1$, C_ε is diagonal and when $k=2$, it presents a 3×3 block diagonal structure. Thus, it is easy to compute the inverse C_ε^{-1} and $\boldsymbol{\beta}$ can be eliminated from the Jacobian of the regularized system. After this elimination, it remains only one scalar unknown u and the corresponding reduced matrix is the Schur complement $S_\varepsilon = A + B^T C_\varepsilon B$. This reduced matrix S_ε can be factored one time for all before to start the GMRES iterations.

Finally, the Jacobian matrix $\mathscr{A}_\varepsilon$ is expected to be close to the unregularized one $\mathscr{A}_0 \in \partial F$. Also, as $\mathscr{A}_\varepsilon$ is non-singular and much more easier to invert than $\mathscr{A}_0$. Thus, it is a good candidate to be a preconditioner: the next section will confirm its efficiency on numerical experiments while appendix B presents a study of the spectral convergence of $\mathscr{A}_\varepsilon$ to $\mathscr{A}_0$ when $\varepsilon \to 0$.

5 Numerical results and performances

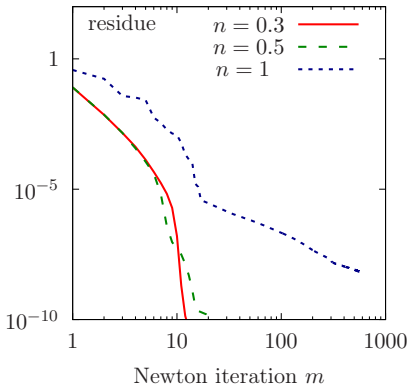

Figure 3: Damped Newton method for the Herschel-Bulkley problem: residue vs Newton iterations for various n, with $Bi = 0.1$, $r = 0.5$, $\varepsilon = 10^{-5}$, $k = 1$ and $h = 1/160$.

Fig. 3 shows the convergence of the Newton method for various values of the power law index n and an uniform mesh with $h = 1/160$. The residue at iteration m is computed as the L^2 norm of $F(u_m, \boldsymbol{\beta}_m)$. Recall that the unregularized problem is characterized by $F(u, \boldsymbol{\beta}) = 0$. Thus, the residue $F(u_m, \boldsymbol{\beta}_m)$ is associated to the unregularized problem: it is independent upon $\varepsilon > 0$ that

denotes the parameter of the preconditioner, associated to the Jacobian of a regularized problem. When the residue tends to zero while iteration $m \to +\infty$, then $(u_m, \boldsymbol{\beta}_m)$ tends to a solution of the unregularized problem. The stopping criteria on the residue is 10^{-12} and the preconditioner uses $\varepsilon = 10^{-5}$ on Fig. 3. Observe that, for both $n = 0.3$ and $n = 0.5$, the convergence is very fast, less than 20 iterations: this is the expected behavior of the Newton method when F is sufficiently regular. Changing the value of the parameter r has few influence on performances and all computations presented in this paper are performed with $r = 0.5$. When $n = 1$ (Bingham model), the convergence is much more slower: it is asymptotically linear in log-log scale, which means that the residue decreases as $1/m^{\alpha}$. This slow down of the convergence rate is probably due to the lower regularity of F when $n = 1$ (see also Fig. 2).

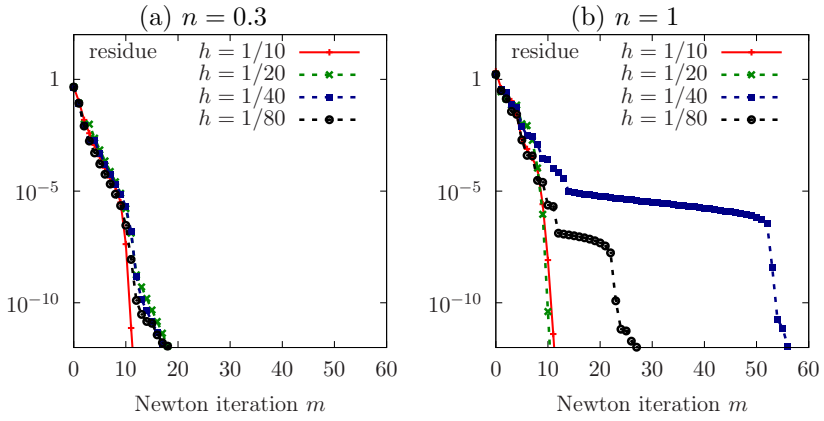

Figure 4: Damped Newton method for the Herschel-Bulkley problem: residue vs Newton iterations for various mesh refinement h with $Bi = 0.5$, $r = 0.5$, $k = 1$ and $\varepsilon = 10^{-5}$. (a) $n = 0.3$; (b) $n = 1$.

Observe on Fig. 4.a that when $n = 0.3$, the convergence is asymptotically *mesh-invariant* (for the mesh-invariance property of nonlinear algorithms, see [39, chap. 8]). Conversely, when $n = 1$ (Fig. 4.b), the convergence rate depends mesh refinement and there are long plateau where the residue decreases slowly.

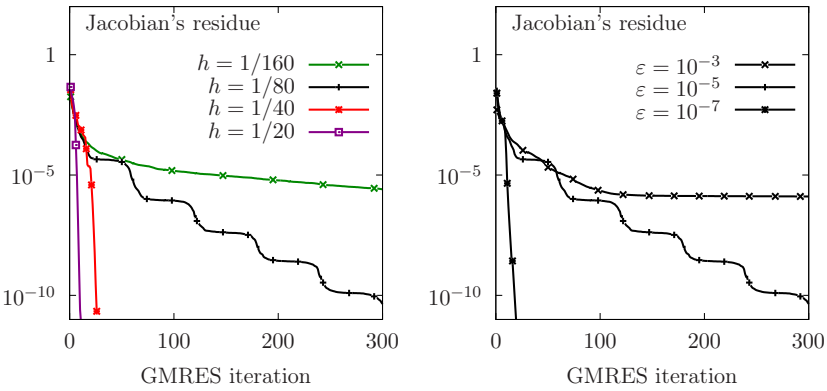

Figure 5: Preconditioning the Jacobian by using the Jacobian of the regularized problem: $Bi = 0.1$, $n = 0.5$, $k = 1$ and $r = 0.5$. (a) $\varepsilon = 10^{-5}$ and varying h; (b) $h = 1/80$ and varying ε.

There are two loops: one outer loop, with index m, for the Newton iterations, and one inner loop for the iterative resolution of the Jacobian. As we have now studied the outer loop, let us look at the inner one. Fig. 5.a shows the convergence properties of the Jacobian solver with the GMRES algorithm for various meshes and the preconditioner based on the regularized problem with

$\varepsilon = 10^{-5}$. For the smallest meshes, the Jacobian system is solved with few iterations while the convergence rate becomes slower with mesh refinement and the largest meshes are still the most difficult to solve. Fig. 5.b shows that the preconditioner efficiency increases when ε decreases, as the regularized Jacobian approaches better the exact one. Using too small ε, lower than 10^{-7}, could interfere with the finite machine precision, about 10^{-15} for double precision. Increasing the machine precision, e.g. quadruple precision could be useful here, in order to continue to decrease ε and increase the solver efficiency.

An *inexact* variant of the Newton [13] method is possible and very efficient here. The idea is to stop the inner GMRES iteration when the residue of the Jacobian system reaches a ratio, e.g. 10%, of the residue of the current Newton iteration. For simplicity, let us denote $\chi = (u, \boldsymbol{\beta})$ and consider the Jacobian system with $\mathscr{A}_0 \in \partial F(\chi)$:

$$\mathscr{A}_0 \ \delta\chi = -F(\chi)$$

In that case, the iterative GMRES solver stops when the residue is less than $0.1 \times \|F(\chi)\|$. This modification maintains the superlinear convergence property of the Newton method and each linear solver call requires only very few iterations, thanks to the efficient preconditioner.

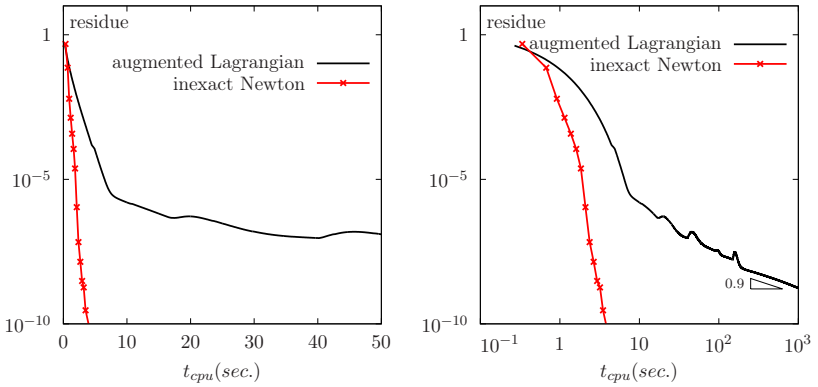

Figure 6: Comparison between the inexact damped Newton method and the augmented Lagrangian algorithm (AL) for the Herschel-Bulkley problem: residual term vs CPU time, in seconds when $Bi = 0.1$, $n = 0.5$, $h = 1/80$, $k = 1$ ($r = 0.5$, $\varepsilon = 10^{-7}$ for Newton and $r = 7$ for AL) (a) in semi-log scale ; (b) in log-log scale.

Fig. 6 plots a comparison of the present inexact preconditioned damped Newton algorithm with the classical Uzawa/augmented Lagrangian method, as proposed in [41] for the Bingham model and extended in [40, ch. 3] to the Herschel-Bulkley model. The augmentation parameter r for the augmented Lagrangian algorithm (AL) has been specially optimized ($r = 7$) for the present mesh (a pipe sector with $h = 1/80$ and 5781 elements) in order to present the best possible convergence rate. Both algorithms are implemented in the `Rheolef` free software FEM library [39]. The Poisson matrix with no-slip boundary condition used by the AL is factored in sparse format one time for all, thanks to the SUITESPARSE library [8]. At each iteration of the AL, a linear system is solved, based on this factorization. For the present inexact Newton algorithm, the Schur complement S_ε of the Jacobian matrix of the regularized problem, used as preconditioner, is also factored by the same way. This factorization has too be performed at each iteration of the Newton method, so each iteration of the Newton method is expected to be slower than its AL counterpart. For this reason, Fig. 6 compares these two methods in term of the CPU time. Observe the dramatic efficiency of the Newton algorithm, which converges in less than 5 seconds to a residue less than 10^{-10} while the AL becomes slower and slower in semi-log scale and adopts an asymptotic behavior, as shown in log-log scale: after a rapid decrease until 10^{-5}, it slow down and asymptotically behaves as $1/t^\alpha$, with α between 0.9 and 1. It reaches 10^{-10} after about three hours. By extrapolation, 10^{-11} will be reached after one day, 10^{-12} after 12 days and 10^{-15} after 31 years.

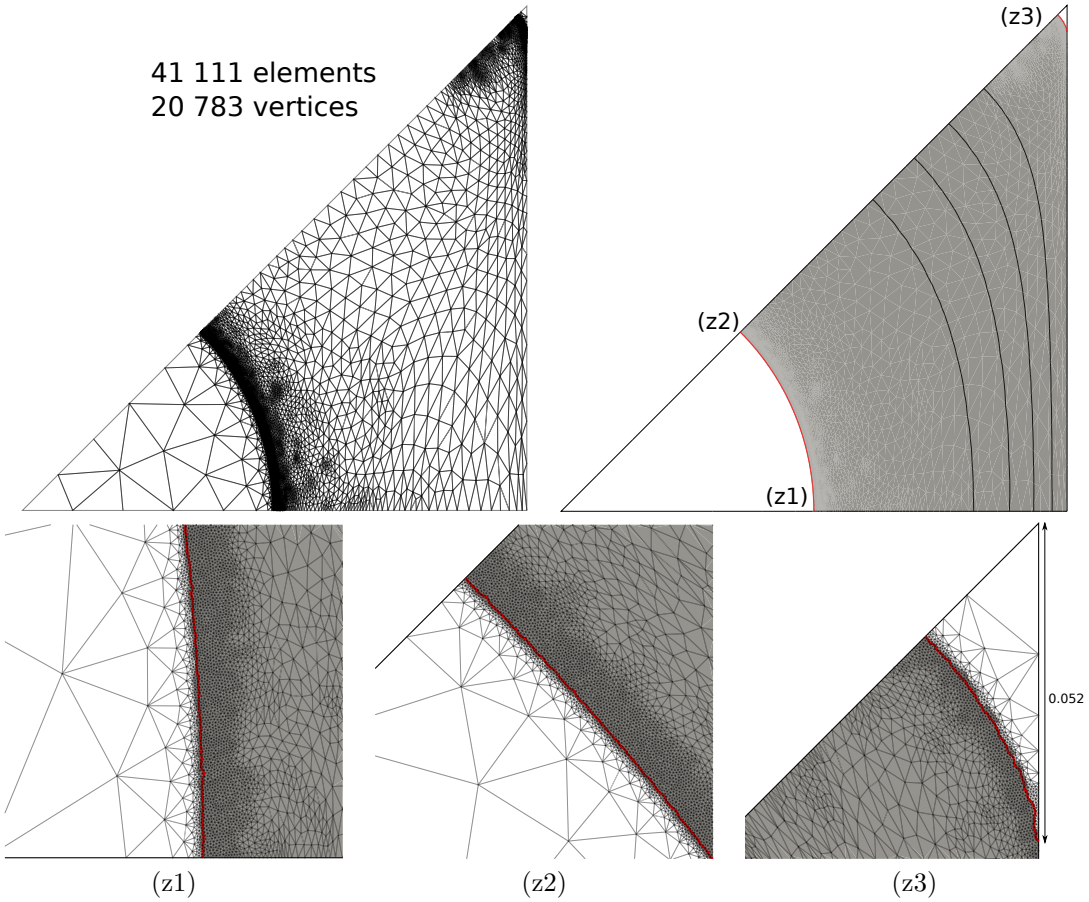

Figure 7: Representation of the auto-adaptive mesh and the solution obtained by the Newton method ($Bi = 0.5$, $n = 0.5$ and $k = 2$). From top to bottom and left to right: the adaptive mesh, global view, zooms along the x axis (z1), along the first bisector (z2) and near the dead zone (z3). In gray, the shear region $|\boldsymbol{\sigma}| > Bi$. Black lines are isocontours of the velocity. Comparison with the augmented Lagrangian solution: in red, isocontour $|\boldsymbol{\sigma}| = Bi$.

Fig. 7 shows the solution obtained with $k = 2$ (quadratic approximation for the velocity and discontinuous piecewise linear approximation of the stresses). The automatic adaptive mesh procedure, as presented in [41, 35], is used here, combined here with the Newton solver instead of the augmented Lagrangian one. Observe that the mesh is refined along the yield surface: as shown in [34], this procedure is required in order to recover an optimal convergence rate with respect to the polynomial degree k. The shear region is represented in gray while black lines are isocontours of the velocity. The yield surfaces is compared with those as predicted by the augmented Lagrangian algorithm (red lines) on the same adapted mesh. Both the augmented Lagrangian and the Newton algorithms were stopped when the residual term becomes lower than 10^{-10}. The maximum velocity, reached in the central plug, is 6.602×10^{-2} with less than 0.03 % of relative error between booth algorithms. The minimum edge length is 3×10^{-4}. Observe the good correspondence between both algorithms for the prediction of the yield surface: the difference is not perceptible on the global view. Zooms close to the intersection of the yield surface with the boundaries show some tiny differences that are of the order of magnitude of the smallest edge length of the mesh. The augmented Lagrangian algorithm stops after 252 193 iterations and uses 41 CPU hours while the Newton one stops after 27 iterations and uses 404 CPU seconds. The speedup is of about 350 in this case.

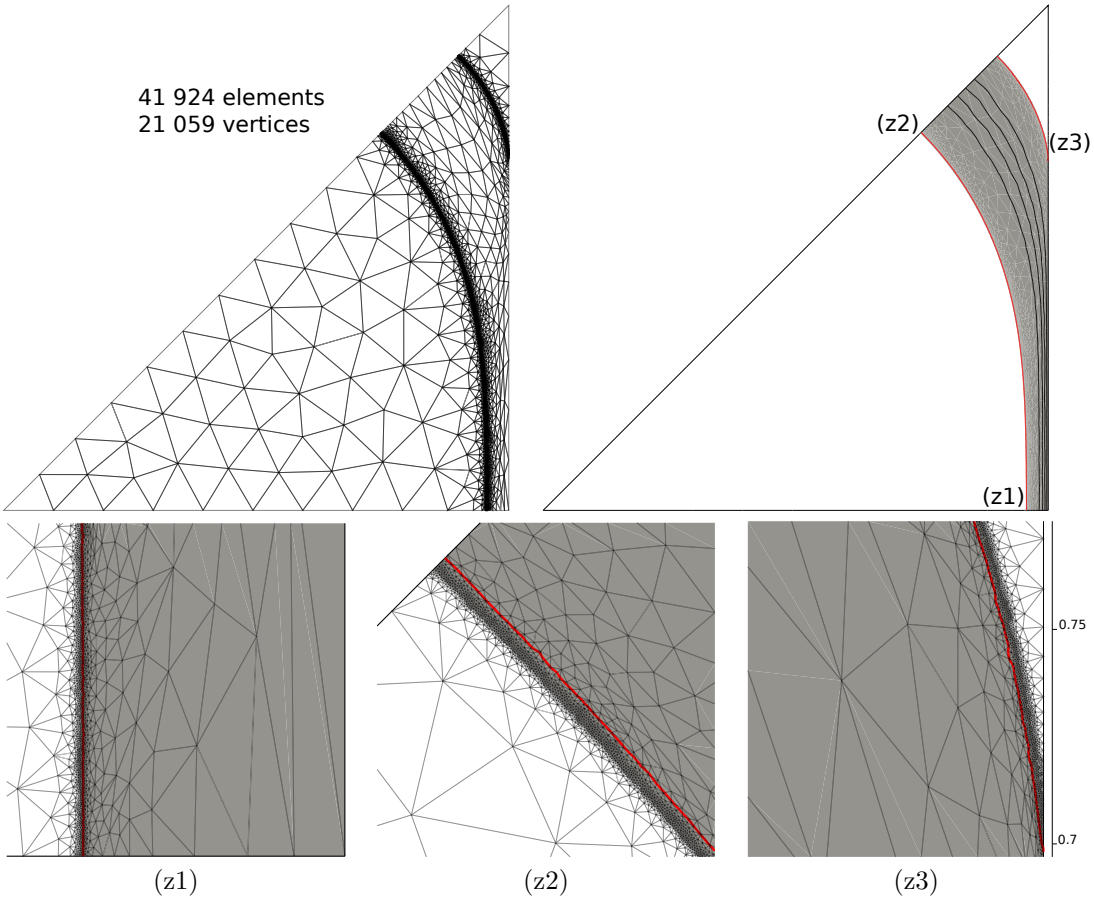

Figure 8: Representation of the auto-adaptive mesh and the solution obtained by the Newton method ($Bi = 1$, $n = 0.5$ and $k = 2$). Legend is the same as for Fig. 7.

Fig. 8 presents a similar comparison when $Bi = 1$: this situation is close to the arrested state $u = 0$, which occurs at a critical Bingham number $Bi_c = 4/(2 + \sqrt{\pi}) \approx 1.0603178\ldots$ as shown in [41]. Notice that this critical Bingham number Bi_c do not depend upon the power law index n. It is defined as the solution of the following limit load analysis problem: find the smallest Bi such that there exists $\boldsymbol{\sigma}$ satisfying $|\boldsymbol{\sigma}| \leqslant Bi$ and $\operatorname{div} \boldsymbol{\sigma} = -f$ in Ω. This is a challenging computation: it is difficult to get accurate solutions at the vicinity of the arrested state, and especially accurate predictions of the yield surface near the arrested state. When $Bi = 1$, the velocity is small: both the Newton and the augmented Lagrangian algorithms obtain that the maximum velocity is 1.04×10^{-4} with about 1 % of relative error. Both the augmented Lagrangian and the Newton algorithms were stopped when the residual term becomes lower than 10^{-10}. Fig. 8 confirms the good agreement between the computations obtained by these algorithms. Zooms show that the small differences observed for the prediction of the yield surface near the dead zone remains of the order of magnitude of the smallest edge length of the mesh, which is 2×10^{-4}. The augmented Lagrangian algorithm stops after 315 131 iterations and uses 52 CPU hours while the Newton one stops after 20 iterations and uses 110 CPU seconds. The speedup is of about 1700 in this case.

Conclusion

For the first time, a Newton method is proposed for the unregularized viscoplastic fluid flow problem. This method bases on a reformulation of the problem in terms of a projection operator. It leads to a superlinear convergence for Herschel-Bulkley fluids when $0 < n < 1$. At each iteration, the singular Jacobian system is solved by an iterative method and an efficient preconditioner based on the regularized problem. An inexact approach permits to increase the performances of the algorithm. A demonstration is provided by computing a viscoplastic flow in a pipe with a square cross section. Comparisons with the augmented Lagrangian algorithm show a dramatic reduction of the required computing time while this algorithm provides an equivalent accuracy for the prediction of the yield surfaces. Future work will extend this approach to larger flow problems such as flows around obstacles and tridimensional geometries.

Acknowledgment

The author would like to thank the anonymous reviewers for constructive comments that help improving the redaction of this paper.

References

[1] P. Alart and A. Curnier. A mixed formulation for frictional contact problems prone to Newton like solution methods. *Comput. Methods Appl. Mech. Engrg.*, 92(3):353–375, 1991.

[2] A. Aposporidis, P. S. Vassilevski, and A. Veneziani. Multigrid preconditioning of the non-regularized augmented bingham fluid problem. *Elect. Trans. Numer. Anal. (ETNA)*, 41:42–61, 2014.

[3] M. Bercovier and M. Engelman. A finite-element method for incompressible non-Newtonian flows. *J. Comp. Phys.*, 36:313–326, 1980.

[4] N. Bernabeu, P. Saramito, and C. Smutek. Numerical modeling of shallow non-newtonian flows: Part II. viscoplastic fluids and general tridimensional topographies. *Int. J. Numer. Anal. Model.*, 11(1):213–228, 2014.

[5] C. R. Beverly and R. I. Tanner. Numerical analysis of three-dimensional bingham plastic flow. *J. Non-Newt. Fluid Mech.*, 42(1):85–115, 1992.

[6] E. C. Bingham. *Fluidity and plasticity.* Mc Graw-Hill, New-York, USA, 1922. http://www.archive.org/download/fluidityandplast007721mbp/fluidityandplast007721mbp.pdf.

[7] F. H. Clarke. *Optimization and nonsmooth analysis.* SIAM, Philadelphia, USA, 1990.

[8] T. A. Davis. *UMFPACK version 5.6 user guide.* University of Florida, USA, 2012.

[9] J. C. de los Reyes and S. A. González Andrade. Numerical simulation of two-dimensional Bingham fluid flow by semismooth Newton methods. *J. Comput. Appl. Math.*, 235:11–32, 2010.

[10] J. C. de los Reyes and S. A. González Andrade. A combined BDF-semismooth Newton approach for time-dependent Bingham flow. *Numer. Meth. Part. Diff. Eqn.*, 28(3):834–860, 2012.

[11] J. C. de los Reyes and S. A. González Andrade. Numerical simulation of thermally convective viscoplastic fluids by semismooth second order type methods. *J. Non-Newt. Fluid Mech.*, 193:43–48, 2013.

[12] Y. Dimakopoulos, M. Pavlidis, and J. Tsamopoulos. Steady bubble rise in Herschel–Bulkley fluids and comparison of predictions via the augmented Lagrangian method with those via the Papanastasiou model. *J. Non-Newt. Fluid Mech.*, 200:34–51, 2013.

[13] S. C. Eisenstat and H. F. Walker. Globally convergent inexact Newton methods. *SIAM J. Optim.*, 4(2):393–422, 1994.

[14] H. C. Elman, D. J. Silvester, and A. J. Wathen. *Finite elements and fast iterative solvers with applications in incompressible fluid dynamics*. Oxford University Press, UK, 2005.

[15] M. Fortin and R. Glowinski. *Augmented Lagrangian methods*. Elsevier, 1983.

[16] I. A. Frigaard and C. Nouar. On the usage of viscosity regularisation methods for visco-plastic fluid flow computation. *J. Non-Newt. Fluid Mech.*, 127(1):1–26, 2005.

[17] R. Glowinski. *Lecture on numerical methods for nonlinear variational problems*. Springer, 1980.

[18] R. Glowinski and P. Le Tallec. *Augmented Lagrangian and operator splitting methods in nonlinear mechanics*. SIAM, Philadelphia, USA, 1989.

[19] W. H. Herschel and T. Bulkley. Measurement of consistency as applied to rubber-benzene solutions. *Proceedings of the American Society for Testing and Material*, 26(2):621–633, 1926.

[20] M. R. Hestenes. Multiplier and gradient methods. *J. Optim. Theory Appl.*, 4(5):303–320, 1969.

[21] E. Mitsoulis, S. S. Abdali, and N. C. Markatos. Flow simulation of Herschel-Bulkley fluids through extrusion dies. *Can. J. Chem. Eng.*, 71:147–160, 1993.

[22] E. Mitsoulis and R. R. Huilgol. Entry flows of Bingham plastics in expansions. *J. Non-Newtonian Fluid Mech.*, 122:45–54, 2004.

[23] M. A. Moyers-Gonzalez and I. A. Frigaard. Numerical solution of duct flows of multiple visco-plastic fluids. *J. Non-Newtonian Fluid Mech.*, 127:227–241, 2004.

[24] L. Muravleva, E. Muravleva, G. C. Georgiou, and E. Mitsoulis. Numerical simulations of cessation flows of a Bingham plastic with the augmented Lagrangian method. *J. Non-Newtonian Fluid Mech.*, 165:544–550, 2010.

[25] J. G. Oldroyd. A rational formulation of the equations of plastic flow for a Bingham fluid. *Proc. Cambridge Philos. Soc.*, 43:100–105, 1947.

[26] J. M. Ortega and W. C. Rheinboldt. *Iterative solution of nonlinear equations in several variables*. SIAM, Philadelphia, PA, USA, 1970.

[27] C. C. Paige and M. A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM J. Numer. Anal.*, 12(4):617–629, 1975.

[28] T. C. Papanastasiou. Flow of materials with yield. *J. Rheol.*, 31:385–404, 1987.

[29] M. J. D. Powell. *A method for nonlinear constraints in minimization problems*, pages 283–298. Academic Press, London, 1969.

[30] A. Putz and I. A. Frigaard. Creeping flow around particle in a Bingham fluid. *J. Non-Newt. Fluid Mech.*, 165(5–6):263–280, 2010.

[31] A. Putz, I. A. Frigaard, and D. M. Martinez. On the lubrication paradox and the use of regularisation methods for lubrication flows. *J. Non-Newt. Fluid Mech.*, 163:62–77, 2009.

[32] L. Qi and J. Sun. A nonsmooth version of Newton's method. *Math. Prog.*, 58(1-3):353–367, 1993.

[33] R. T. Rockafellar. Augmented Lagrangians and applications of the proximal point algorithm in convex programming. *Math. Oper. Res.*, 1(2):97–116, 1976.

[34] N. Roquet, R. Michel, and P. Saramito. Errors estimate for a viscoplastic fluid by using Pk finite elements and adaptive meshes. *C. R. Acad. Sci. Paris, ser. I*, 331(7):563–568, 2000.

[35] N. Roquet and P. Saramito. An adaptive finite element method for Bingham fluid flows around a cylinder. *Comput. Appl. Meth. Mech. Engrg.*, 192(31-32):3317–3341, 2003.

[36] A. Roustaei and I. A. Frigaard. The occurrence of fouling layers in the flow of a yield stress fluid along a wavy-walled channel. *J. Non Newt. Fluid Mech.*, 198:109–124, 2013.

[37] A. Roustaei, A. Gosselin, and I. A. Frigaard. Residual drilling mud during conditioning of uneven boreholes in primary cementing. Part 1: rheology and geometry effects in non-inertial flows. *J. Non-Newt. Fluid Mech.*, to appear, 2014.

[38] Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, 1986.

[39] P. Saramito. *Efficient C++ finite element computing with Rheolef*. CNRS and LJK, 2013. http://cel.archives-ouvertes.fr/cel-00573970.

[40] P. Saramito. *Méthodes numériques en fluides complexes : théorie et algorithmes*. CNRS-CCSD, 2013. http://cel.archives-ouvertes.fr/cel-00673816.

[41] P. Saramito and N. Roquet. An adaptive finite element method for viscoplastic fluid flows in pipes. *Comput. Meth. Appl. Mech. Eng.*, 190(40-41):5391–5412, 2001.

[42] T. Treskatis, M. A. Moyers-Gonzalez, and C. J. Price. A trust-region SQP method for the numerical approximation of viscoplastic fluid flow. *submitted*, 2015.

[43] D. Vola, L. Boscardin, and J. C. Latché. Laminar unsteady flows of Bingham fluids: a numerical strategy and some benchmark results. *J. Comput. Phys.*, 187:441–456, 2003.

[44] S. D. R. Wilson and A. J. Taylor. The channel entry problem for a yield stress fluid. *J. Non-Newt. Fluid Mech.*, 65:165–176, 1996.

A Equivalence proof for the reformulations with projections

PROPERTY **1** *(equivalence of the formulation with projection)*
Problems (1a)-(1d) *and* (2a)-(2c) *are equivalent.*

Proof: As (1c)-(1d) and (2b)-(2c) are identical, it is sufficient to prove that (1a)-(1b) is equivalent to (2a). Assuming $\nabla u \neq 0$, we take the norm of (1a) and obtain $|\boldsymbol{\sigma}| = K|\nabla u|^n + \sigma_0$ Then, as $|\boldsymbol{\sigma}| \geqslant \sigma_0$, we get $|\nabla u| = K^{-1/n}\left(|\boldsymbol{\sigma}| - \sigma_0\right)^{1/n}$. Remark that (1a) expresses that the vectors $\boldsymbol{\sigma}$ and ∇u are co-linear, i.e. $\dfrac{\nabla u}{|\nabla u|} = \dfrac{\boldsymbol{\sigma}}{|\boldsymbol{\sigma}|}$. Substituting the previous expression of $|\nabla u|$ in terms of $|\boldsymbol{\sigma}|$, we get $\nabla u = |\nabla u|\dfrac{\boldsymbol{\sigma}}{|\boldsymbol{\sigma}|} = K^{-1/n}\left(|\boldsymbol{\sigma}| - \sigma_0\right)^{1/n}\dfrac{\boldsymbol{\sigma}}{|\boldsymbol{\sigma}|}$. From the definition (3) of the projection P_0, this is equivalent (2a) when $|\boldsymbol{\sigma}| \geqslant \sigma_0$. Otherwise, when $|\boldsymbol{\sigma}| \leqslant \sigma_0$, we have $\nabla u = 0$ and the proof is complete. □

PROPERTY **2** *(equivalence for the augmented projection)*
Relation (4) *is satisfied for all* $r \geqslant 0$.

Proof: Suppose first that $|\boldsymbol{\sigma} + r\boldsymbol{\gamma}| > \sigma_0$. Then, tacking the norm of $\boldsymbol{\gamma} = P_r(\boldsymbol{\sigma} + r\boldsymbol{\gamma})$ leads to $|\boldsymbol{\gamma}| = \varphi_r^{-1}(|\boldsymbol{\sigma} + r\boldsymbol{\gamma}|)$ or equivalently $|\boldsymbol{\sigma} + r\boldsymbol{\gamma}| = \varphi_r(|\boldsymbol{\gamma}|) = \sigma_0 + K|\boldsymbol{\gamma}|^n + r|\boldsymbol{\gamma}|$. Remark that $\boldsymbol{\gamma} = P_r(\boldsymbol{\sigma} + r\boldsymbol{\gamma})$ implies that vectors $\boldsymbol{\gamma}$ and $\boldsymbol{\sigma} + r\boldsymbol{\gamma}$ are co-linear, and then that $\boldsymbol{\gamma}$ and $\boldsymbol{\sigma}$ are also co-linear. Then $|\boldsymbol{\sigma} + r\boldsymbol{\gamma}| = |\boldsymbol{\sigma}| + r|\boldsymbol{\gamma}|$ and the previous relation gives $|\boldsymbol{\sigma}| = \sigma_0 + K|\boldsymbol{\gamma}|^n > \sigma_0$. As $\boldsymbol{\gamma}$ and $\boldsymbol{\sigma}$ are co-linear, we have $\boldsymbol{\sigma} = |\boldsymbol{\sigma}|\dfrac{\boldsymbol{\gamma}}{|\boldsymbol{\gamma}|} = \sigma_0\dfrac{\boldsymbol{\gamma}}{|\boldsymbol{\gamma}|} + K|\boldsymbol{\gamma}|^{n-1}\boldsymbol{\gamma}$ or equivalently $\boldsymbol{\gamma} = K^{-1/n}\left(|\boldsymbol{\sigma}| - \sigma_0\right)^{1/n}\dfrac{\boldsymbol{\sigma}}{|\boldsymbol{\sigma}|}$. This is exactly $\boldsymbol{\gamma} = P_0(\boldsymbol{\sigma} + r\boldsymbol{\gamma})$ when $\boldsymbol{\sigma} > \sigma_0$. Otherwise, when $|\boldsymbol{\sigma} + r\boldsymbol{\gamma}| \geqslant \sigma_0$ then $\boldsymbol{\gamma} = 0$ and thus $|\boldsymbol{\sigma}| \geqslant \sigma_0$ which completes the proof. □

PROPERTY **3** *(equivalence of the augmented formulation with projection)*
Problems (1a)-(1d) *and* (6a)-(2c) *are equivalent.*

Proof: From property 1, it is sufficient to prove that problems (1a)-(1d) and (6a)-(2c) are equivalent. Replacing $\boldsymbol{\beta}$ by $\boldsymbol{\sigma} + r\nabla u$ in (6a) leads to (1c). The equivalence between (6a) and (1c) is a direct consequence of property 2. □

B Spectral study of the preconditioner

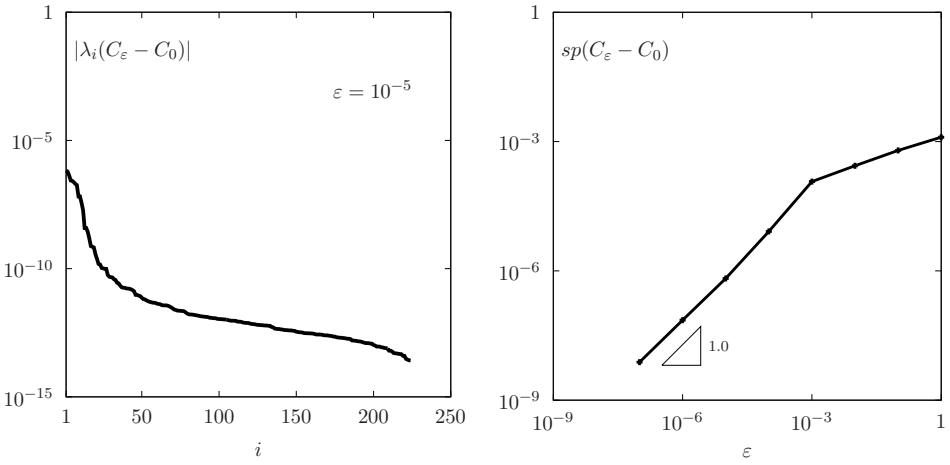

Figure 9: Spectral study of the preconditioner: (left) absolute values of the eigenvalues of $C_\varepsilon - C_0$; (right) spectral radius $\mathrm{sp}(C_\varepsilon - C_0)$ vs ε. With $Bi = 0.1$, $n = 0.5$, $h = 1/10$ and $k = 1$.

A spectral investigation of the matrix $\mathscr{A}_\varepsilon^{-1}\mathscr{A}_0$ of the preconditioned system (7) is performed in this appendix. Here, $\mathscr{A}_\varepsilon$ denotes the Jacobian of the regularized system and $\mathscr{A}_0$ those of the unregularized one. Fig. 9 (left) plots the eigenvalues of $C_\varepsilon - C_0$ for $\varepsilon = 10^{-5}$ for a small-sized problem ($h = 1/10$). Observe that all eigenvalues are lower than 10^{-5}. Fig. 9 (right) represents the spectral radius $\mathrm{sp}(C_\varepsilon - C_0) = \lambda_{\max}(C_\varepsilon - C_0) - \lambda_{\min}(C_\varepsilon - C_0)$ versus ε. where $\lambda_{\max}$and $\lambda_{\min}$ denotes the two extremal eigenvalues. Observe that $\mathrm{sp}(C_\varepsilon - C_0) = \mathcal{O}(\varepsilon)$.

PROPERTY **4** *(convergence of the preconditioner)*
The matrix of preconditioned linear system $\mathscr{A}_\varepsilon^{-1}\mathscr{A}_0$ *converge to the identity when* $\varepsilon \to 0$. *Moreover, the distance of its eigenvalues from* 1 *scales as* $\mathcal{O}(\varepsilon)$.

Proof: The proof is inspired by Aposporidis *et al.* [2, p. 46]: these authors studied a preconditioner with a similar structure, in a different context, for a fixed point algorithm and a different reformulation of the problem. The matrix $\mathscr{A}_\varepsilon$ admits the following block factorization as a product of block triangular matrix [14, ch. 6]:

$$\mathscr{A}_\varepsilon = \begin{pmatrix} A & B^T \\ B & -C_\varepsilon \end{pmatrix} = \begin{pmatrix} A & 0 \\ B & -S_\varepsilon \end{pmatrix} \begin{pmatrix} I & A^{-1}B^T \\ 0 & I \end{pmatrix}$$

where $S_\varepsilon = C_\varepsilon + BA^{-1}B^T$ denotes the Schur complement of $\mathscr{A}_\varepsilon$. Then

$$\mathscr{A}_\varepsilon^{-1} = \begin{pmatrix} I & -A^{-1}B^T \\ 0 & I \end{pmatrix} \begin{pmatrix} A^{-1} & 0 \\ S_\varepsilon^{-1}BA^{-1} & -S_\varepsilon^{-1} \end{pmatrix}$$

and

$$\mathscr{A}_\varepsilon^{-1}\mathscr{A}_0 = \begin{pmatrix} I & A^{-1}B^T(I - S_\varepsilon^{-1}S_0) \\ 0 & S_\varepsilon^{-1}S_0 \end{pmatrix}$$

where $S_0 = C_0 + BA^{-1}B^T$. Remark that $\mathscr{A}_\varepsilon^{-1}\mathscr{A}_0$ tends to the identity when $S_\varepsilon^{-1}S_0$ tends also to I. Then, the convergence study of $\mathscr{A}_\varepsilon^{-1}\mathscr{A}_0$ reduces to those of $S_\varepsilon^{-1}S_0$.

$$\begin{aligned} S_\varepsilon^{-1}S_0 &= \left(C_\varepsilon + BA^{-1}B^T\right)^{-1}\left(C_0 + BA^{-1}B^T\right) \\ &= \left(C_\varepsilon + BA^{-1}B^T\right)^{-1}\left(C_\varepsilon + BA^{-1}B^T - (C_\varepsilon - C_0)\right) \\ &= I - \left(C_\varepsilon + BA^{-1}B^T\right)^{-1}(C_\varepsilon - C_0) \end{aligned}$$

Let λ_ε be an eigenvalue of $\left(C_\varepsilon + BA^{-1}B^T\right)^{-1}(C_\varepsilon - C_0)$. There exists an associated eigenvector $\boldsymbol{\tau} \neq 0$ such that

$$\begin{aligned} & \left(C_\varepsilon + BA^{-1}B^T\right)^{-1}(C_\varepsilon - C_0)\,\boldsymbol{\tau} = \lambda_\varepsilon \boldsymbol{\tau} \\ \iff \quad & (1-\lambda_\varepsilon)C_\varepsilon\boldsymbol{\tau} = \left(C_0 + \lambda_\varepsilon BA^{-1}B^T\right)\boldsymbol{\tau} \end{aligned}$$

after some rearrangements. Next, let μ_ε be an eigenvalue of $\left(C_0 + BA^{-1}B^T\right)^{-1}(C_\varepsilon - C_0)$. There exists an associated eigenvector $\boldsymbol{\zeta} \neq 0$ such that

$$\begin{aligned} & \left(C_0 + BA^{-1}B^T\right)^{-1}(C_\varepsilon - C_0)\,\boldsymbol{\zeta} = \mu_\varepsilon \boldsymbol{\zeta} \\ \iff \quad & (1-\tilde{\lambda}_\varepsilon)C_\varepsilon\boldsymbol{\zeta} = \left(C_0 + \tilde{\lambda}_\varepsilon BA^{-1}B^T\right)\boldsymbol{\zeta} \end{aligned}$$

with $\tilde{\lambda}_\varepsilon = \mu_\varepsilon/(1+\mu_\varepsilon)$. Then $\tilde{\lambda}_\varepsilon$ is an eigenvalue of $\left(C_\varepsilon + BA^{-1}B^T\right)^{-1}(C_\varepsilon - C_0)$. As $\mathrm{sp}(C_\varepsilon - C_0)$ tends to zero as $\mathcal{O}(\varepsilon)$ (see Fig. 9), we have $\mu_\varepsilon = \mathcal{O}(\varepsilon)$ and so is $\tilde{\lambda}_\varepsilon$. Then $S_\varepsilon^{-1}S_0$ and $\mathscr{A}_\varepsilon^{-1}\mathscr{A}_0$ tend to identity when $\varepsilon \to 0$ and the distance of their eigenvalues from 1 scales as $\mathcal{O}(\varepsilon)$. □