Semi-smooth Newton method for solving the Stokes problem with the stick-slip boundary condition

Kučera Radek, Pacholek Jan

VŠB-TU Ostrava, 17. listopadu 15/2172, 708 33 Ostrava-Poruba, CZ
E-mail: jan.pacholek.st@vsb.cz

Abstract: The paper deals with the Stokes flow with the monotonously increasing threshold slip boundary condition. Using the P1-bubble/P1 finite element approximation of the velocity-pressure formulation we arrive at an algebraic variational inequality. This inequality is equivalent to a saddle-point problem whose optimality conditions are the starting point for the algorithm. The semi-smooth Newton method implementation of the algorithm is based on active/inactive sets. We discuss two possible options on how to create them. The algorithm is then tested in MATLAB environment. Experiments are done on the "L-shaped" domain, where we study the effects of the adhesive coefficient and preconditioning on the efficiency of computations.

Keywords: semi-smooth Newton method, stick-slip condition, Stokes problem, preconditioning

1 Introduction

Let Ω be a bounded domain in $\mathbb{R}^2$ with a sufficiently smooth boundary $\partial\Omega$ that is split into three nonempty disjoint parts: $\partial\Omega = \overline{\gamma}_{\mathrm{D}} \cup \overline{\gamma}_{\mathrm{N}} \cup \overline{\gamma}_{\mathrm{C}}$. We consider the model of a viscous incompressible Newtonian fluid modelled by the Stokes system with the Dirichlet and Neumann boundary conditions on γ_{D} and γ_{N}, respectively, and with the impermeability and the stick-slip boundary condition of the Navier-Tresca type on γ_{C}:

$$\left.\begin{aligned} -\nu\Delta\mathbf{u} + \nabla p &= \mathbf{f} && \text{in } \Omega,\\ \nabla\cdot\mathbf{u} &= 0 && \text{in } \Omega,\\ \mathbf{u} &= \mathbf{u}_{\mathrm{D}} && \text{on } \gamma_{\mathrm{D}},\\ \boldsymbol{\sigma} &= \boldsymbol{\sigma}_{\mathrm{N}} && \text{on } \gamma_{\mathrm{N}},\\ u_n &= 0 && \text{on } \gamma_{\mathrm{C}},\\ u_t = 0 \Rightarrow |\sigma_t| &\leq \mathrm{g} && \text{on } \gamma_{\mathrm{C}},\\ \sigma_t u_t + \mathrm{g}|u_t| + \kappa u_t^2 &= 0 && \text{on } \gamma_{\mathrm{C}}. \end{aligned}\right\} \tag{1}$$

We are searching for a vector function representing the flow velocity field $\mathbf{u} : \bar{\Omega} \to \mathbb{R}^2$, $\mathbf{u} = (u_1, u_2)$ and a scalar function representing the pressure field $p : \bar{\Omega} \to \mathbb{R}$, where $\nu > 0$ is the dynamic viscosity, $\mathbf{f} : \bar{\Omega} \to \mathbb{R}^2$ describes the forces acting on the fluid, $\mathbf{u}_{\mathrm{D}} : \gamma_{\mathrm{D}} \to \mathbb{R}^2$ and $\boldsymbol{\sigma}_{\mathrm{N}} : \gamma_{\mathrm{N}} \to \mathbb{R}^2$ are the Dirichlet and Neumann boundary data, respectively. Further $\mathbf{n}$ and $\mathbf{t}$ are the unit outward normal and tangential vectors on $\partial\Omega$ and we define the normal and tangential component of the velocity and the stress as

$$u_n = \mathbf{u} \cdot \mathbf{n},\ u_t = \mathbf{u} \cdot \mathbf{t},\ \sigma_n = \boldsymbol{\sigma} \cdot \mathbf{n},\ \sigma_t = \boldsymbol{\sigma} \cdot \mathbf{t},$$

where

$$\boldsymbol{\sigma} = \nu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} - \mathrm{p}\mathbf{n},\ \frac{\partial \mathbf{u}}{\partial \mathbf{n}} = \left(\frac{\partial u_1}{\partial \mathbf{n}}, \frac{\partial u_2}{\partial \mathbf{n}}\right) \quad \text{on } \partial\Omega$$

is the stress vector on $\partial\Omega$. On γ_{C} we consider the given slip bound $g : \gamma_C \to \mathbb{R}_+$ and the adhesive coefficient $\kappa : \gamma_C \to \mathbb{R}_+$ defining the stick-slip boundary condition. We get the classical Navier law for $g = 0$, while $\kappa = 0$ leads to the Tresca law.

The solution of (1) for $\kappa = 0$ has been discussed and solved in [1, 3, 10]. In texts [2] and [9] we can find solutions for $\kappa \neq 0$. Furthermore, in [2] and [7] the interior point method is introduced as one of the methods that may be used in either cases. The other method used to find solution of (1) for $\kappa \neq 0$ is the semi-smooth Newton method [5]. It has been described in [9] and in this text, we will introduce the given algorithm and modify it so that it accepts even $\kappa = 0$.

2 Weak formulation and algebraic problems

The weak formulation of (1) leads to the following variational inequality problem:

$$\left.\begin{array}{l} \text{Find } (\mathbf{u}, p) \in V_{\mathbf{u}_D} \times L^2(\Omega) \text{ so that} \\ a(\mathbf{u}, \mathbf{v}-\mathbf{u}) + b(p, \mathbf{v}-\mathbf{u}) + j(\mathbf{v}, \mathbf{u}) - j(\mathbf{u}, \mathbf{u}) \geq l(\mathbf{v}-\mathbf{u}) \quad \forall \mathbf{v} \in V_{\mathbf{u}_D} \\ b(q, \mathbf{u}) = 0 \quad \forall q \in L^2(\Omega), \end{array}\right\} \tag{2}$$

where $V_{\mathbf{u}_D} = \{\mathbf{v} \in (H^1(\Omega))^2 : \mathbf{v} = \mathbf{u}_D \text{ on } \gamma_D,\ \nu_n = 0 \text{ on } \gamma_C\}$ is the velocity set and

$$a(\mathbf{w}, \mathbf{v}) = \nu \int_\Omega \nabla \mathbf{w} : \nabla \mathbf{v} d\mathbf{x}, \quad b(q, \mathbf{v}) = -\int_\Omega q(\nabla \cdot \mathbf{v}) d\mathbf{x},$$
$$l(\mathbf{v}) = \int_\Omega \mathbf{f} \cdot \mathbf{v} d\mathbf{x} + \int_{\gamma_N} \boldsymbol{\sigma}_N \cdot \mathbf{v} ds, \quad j(\mathbf{v}, \mathbf{w}) = \int_{\gamma_C} g|v_t| + \kappa w_t v_t ds$$

for $\mathbf{v}, \mathbf{w} \in (H^1(\Omega))^2$ and $q \in L^2(\Omega)$. The solvability of (2) can be proven in a similar way as in [3] for $\kappa = 0$. The symbol $\mathbf{u}$ denoting here the vector function will be used also in the subsequent parts for the algebraic vector representing its finite element counterpart.

We approximate (2) by the mixed finite element method based on the P1-bubble/P1 finite elements [8]. The resulting algebraic problem leads to the saddle-point formulation which is

equivalent to the following optimality conditions:

$$\left.\begin{array}{l} \text{Find } (\mathbf{u},\mathbf{p},\mathbf{s}_t,\boldsymbol{\lambda}_n)\in\mathbb{R}^{n_u}\times\mathbb{R}^{n}\times\mathbb{R}^{n_c}\times\mathbb{R}^{n_c} \text{ such that} \\ \mathbf{Au}-\mathbf{l}+\mathbf{T}^T\mathbf{s}_t+\mathbf{N}^T\boldsymbol{\lambda}_n+\mathbf{B}^T\mathbf{p}=\mathbf{0}, \\ \mathbf{Bu}-\mathbf{Ep}-\mathbf{c}=\mathbf{0}, \\ \mathbf{Nu}=\mathbf{0} \\ \left.\begin{array}{l} (\mathbf{Tu})_i=0\Rightarrow|s_{ti}|\le g_i, \\ (\mathbf{Tu})_i>0\Rightarrow s_{ti}=g_i+\kappa_i(\mathbf{Tu})_i, \\ (\mathbf{Tu})_i<0\Rightarrow s_{ti}=-g_i+\kappa_i(\mathbf{Tu})_i, \end{array}\right\} \ i\in\mathcal{N}, \end{array}\right\} \tag{3}$$

where $\mathbf{s}_t=\boldsymbol{\lambda}_t+\mathbf{D}(\boldsymbol{\kappa})\mathbf{Tu}$ and $\mathcal{N}=\{1,\dots,n_c\}$. Here, $\mathbf{A}\in\mathbb{R}^{n_u\times n_u}$ is the symmetric, positive definite stiffness matrix for the Laplace operator, $\mathbf{l}\in\mathbb{R}^{n_u}$, $\mathbf{B}\in\mathbb{R}^{n\times n_u}$ is the full row rank stiffness matrix for the divergence operator. $\mathbf{T},\mathbf{N}\in\mathbb{R}^{n_c\times n_u}$ are full row rank matrices given by the normal and tangential vectors at the nodes $\mathbf{x}_i\in\bar{\gamma}_C\backslash\bar{\gamma}_D$, respectively. Further, $\mathbf{D}(\boldsymbol{\kappa})=\operatorname{diag}(\boldsymbol{\kappa})\in\mathbb{R}^{n_c\times n_c}$, where $\boldsymbol{\kappa}=(\kappa_1,\dots,\kappa_{n_c})^T\in\mathbb{R}^{n_c}$, $\kappa_i=h_i\kappa(\mathbf{x}_i)$, $g_i=h_ig(\mathbf{x}_i)$, and h_i is the length of the segment connecting $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$, $i\in\mathcal{N}$; n_u is the number of velocity components, n is the number of the finite elements nodes, and n_c is the number of the stick-slip nodes. The symmetric, positive semidefinite matrix $\mathbf{E}\in\mathbb{R}^{n_c\times n_c}$ and the vector $\mathbf{c}\in\mathbb{R}^{n_c}$ arise from the elimination of the bubble components. The unknowns $\mathbf{u}$, $\mathbf{p}$ are the vectors of velocity and pressure components, respectively. Finally, $\boldsymbol{\lambda}_t$, $\boldsymbol{\lambda}_n$ are Lagrange multipliers regularizing the non-differentiability of j from (2) and releasing the impermeability condition, respectively. Note that $\mathbf{s}_t,\boldsymbol{\lambda}_n$ approximate $-\sigma_t,-\sigma_n$ on γ_C, respectively.

3 Semi-smooth Newton method for $\kappa>0$

The stick-slip law in (3) is given by the relation between $(\mathbf{Tu})_i$ and s_{ti}; see Figure 1. It can be represented by the piecewise linear function with the independent variable s_{ti} and the dependent variable $(\mathbf{Tu})_i$ as follows:

$$(\mathbf{Tu})_i=\max\{0,\kappa_i^{-1}(s_{ti}-g_i)\}-\max\{0,-\kappa_i^{-1}(s_{ti}+g_i)\}. \tag{4}$$

Problem (3), with the stick-slip condition expressed by (4), can be written as one equation:

$$\mathbf{G}(\mathbf{y})=\mathbf{0} \tag{5}$$

for $\mathbf{G}(\mathbf{y})=(\mathbf{G}_1^T(\mathbf{y}),\mathbf{G}_2^T(\mathbf{y}),\mathbf{G}_3^T(\mathbf{y}),\mathbf{G}_4^T(\mathbf{y}))^T$ and $\mathbf{y}=(\mathbf{u}^T,\mathbf{s}_t^T,\boldsymbol{\lambda}_n^T,\mathbf{p}^T)^T$, where $\mathbf{G}_1(\mathbf{y})=\mathbf{Au}-\mathbf{l}+\mathbf{T}^T\mathbf{s}_t+\mathbf{N}^T\boldsymbol{\lambda}_n+\mathbf{B}^T\mathbf{p}$, $\mathbf{G}_2(\mathbf{y})=\mathbf{Tu}-\boldsymbol{\phi}(\mathbf{D}(\boldsymbol{\kappa})^{-1}(\mathbf{s}_t-\mathbf{g}))+\boldsymbol{\phi}(-\mathbf{D}(\boldsymbol{\kappa})^{-1}(\mathbf{s}_t+\mathbf{g}))$, $\mathbf{G}_3(\mathbf{y})=\mathbf{Nu}$, $\mathbf{G}_4(\mathbf{y})=\mathbf{Bu}-\mathbf{Ep}-\mathbf{c}$, and $\boldsymbol{\phi}(\mathbf{z})=(\phi(z_1),\dots,\phi(z_{n_c}))^T$, $\mathbf{z}\in\mathbb{R}^{n_c}$, $\phi(z)=\max\{0,z\}$, $z\in\mathbb{R}$. Equation (5) can be solved by a Newton-type method. Due to the presence of the max-functions, the Jacobian matrix to $\mathbf{G}$ does not exist at all points so that the classical Newton method can not be used. On the other hand $\mathbf{G}$ is semi-smooth in the sense of [5] and, therefore, the semi-smooth Newton method can be used as a solver.

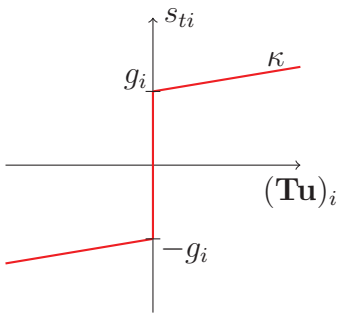

Figure 1: Stick-slip law in the node $\mathbf{x}_i$, $i \in \mathcal{N}$.

The algorithm generates the sequence $\{\mathbf{y}^k\}$. For each $\mathbf{y}^k$ we define the inactive sets

$$\mathcal{I}_t^+ = \{i \in \mathcal{N} : s_{ti}^k \geq g_i\}, \quad \mathcal{I}_t^- = \{i \in \mathcal{N} : s_{ti}^k \leq -g_i\}$$

and the indicator matrices $\mathbf{D}(\mathcal{I}_t^-)$, $\mathbf{D}(\mathcal{I}_t^+)$, respectively. The indicator matrix to $\mathscr{S} \subset \mathcal{N}$ is given by $\mathbf{D}(\mathscr{S}) = \operatorname{diag}(s_1, \ldots, s_{n_c}) \in \mathbb{R}^{n_c \times n_c}$ with $s_i = 1$ for $i \in \mathscr{S}$ and $s_i = 0$ if $\notin \mathscr{S}$. The new iterate $\mathbf{y}^{k+1}$ is computed by solving the following linear system:

$$\left(\begin{array}{c|cc|c} \mathbf{A} & \mathbf{T}^T & \mathbf{N}^T & \mathbf{B}^T \\ \hline \mathbf{T} & -\mathbf{D}(\boldsymbol{\kappa})^{-1}\mathbf{D}(\mathcal{I}_t^+ \cup \mathcal{I}_t^-) & \mathbf{0} & \mathbf{0} \\ \mathbf{N} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} & -\mathbf{E} \end{array}\right) \begin{pmatrix} \mathbf{u}^{k+1} \\ \mathbf{s}_t^{k+1} \\ \boldsymbol{\lambda}_n^{k+1} \\ \mathbf{p}^{k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{l} \\ \mathbf{D}(\boldsymbol{\kappa})^{-1}(\mathbf{D}(\mathcal{I}_t^-) - \mathbf{D}(\mathcal{I}_t^+))\mathbf{g} \\ \mathbf{0} \\ \mathbf{c} \end{pmatrix}. \quad (6)$$

As it is unrealistic to find the exact solution of large linear algebraic systems, we use the Schur complement system to solve (6):

$$\mathbf{S}^k \mathbf{r}^{k+1} = \mathbf{C}\mathbf{A}^{-1}\mathbf{l} - \mathbf{h}^k,$$

with $\mathbf{S}^k = \mathbf{C}\mathbf{A}^{-1}\mathbf{C}^T + \bar{\mathbf{E}}^k$, where

$$\mathbf{C} = \begin{pmatrix} \mathbf{T} \\ \mathbf{N} \\ \mathbf{B} \end{pmatrix}, \quad \bar{\mathbf{E}}^k = \begin{pmatrix} \mathbf{D}(\boldsymbol{\kappa})^{-1}\mathbf{D}(\mathcal{I}_t^+ \cup \mathcal{I}_t^-) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{E} \end{pmatrix}$$

and

$$\mathbf{r}^{k+1} = \begin{pmatrix} \mathbf{s}_t^{k+1} \\ \boldsymbol{\lambda}_n^{k+1} \\ \mathbf{p}^{k+1} \end{pmatrix}, \quad \mathbf{h}^k = \begin{pmatrix} \mathbf{D}(\boldsymbol{\kappa})^{-1}(\mathbf{D}(\mathcal{I}_t^-) - \mathbf{D}(\mathcal{I}_t^+))\mathbf{g} \\ \mathbf{0} \\ \mathbf{c} \end{pmatrix}.$$

We use the conjugate gradient method with an adaptive precision control and the diagonal preconditioner

$$\mathbf{P}^k = \operatorname{diag} \mathbf{S}^k.$$

One can prove [6, 9] that the spectral condition number of $(\mathbf{P}^k)^{-1}\mathbf{S}^k$ does not depend on the block $\mathbf{D}(\boldsymbol{\kappa})^{-1}$, which is ill-conditioned if κ or h_i are too small.

4 Semi-smooth Newton method for $\kappa \geq 0$

Considering the same stick-slip law in (3), we can represent it by another piecewise linear function that will allow us to work even if $\kappa = 0$, which was not possible in (4). Using the projection on the interval $[a, b]$

$$P_{[a,b]}(x) = x - \max\{0, x - b\} + \max\{0, a - x\}, \quad x \in \mathbb{R}$$

the stick-slip law in (3) can be represented by

$$\left.\begin{aligned} (\mathbf{Tu})_i &= \max\{0, \kappa_i^{-1}(s_{ti} - g_i)\} - \max\{0, -\kappa_i^{-1}(s_{ti} + g_i)\} && \text{for } \rho_i = \kappa_i > 0, \\ \rho(\mathbf{Tu})_i &= \max\{0, s_{ti} - g_i + \rho(\mathbf{Tu})_i\} - \max\{0, -s_{ti} - g_i - \rho(\mathbf{Tu})_i\} && \text{for } \kappa_i = 0, \rho > 0. \end{aligned}\right\} \quad (7)$$

The problem (3), with the stick-slip condition represented by (7), can again be written as one equation (5), but with modified $\mathbf{G}(\mathbf{y})$. Now $\mathbf{G}(\mathbf{y}) = (\mathbf{G}_1^T(\mathbf{y}), \mathbf{G}_2^T(\mathbf{y}), \mathbf{G}_3^T(\mathbf{y}), \mathbf{G}_4^T(\mathbf{y}), \mathbf{G}_5^T(\mathbf{y}))^T$ and $\mathbf{y} = (\mathbf{u}^T, \mathbf{s}_t^T, \boldsymbol{\lambda}_n^T, \mathbf{p}^T)^T$, where $\mathbf{G}_1(\mathbf{y}) = \mathbf{Au} - \mathbf{l} + \mathbf{T}_+^T\mathbf{s}_{t+} + \mathbf{T}_0^T\mathbf{s}_{t0} + \mathbf{N}^T\boldsymbol{\lambda}_n + \mathbf{B}^T\mathbf{p}$, $\mathbf{G}_2(\mathbf{y}) = \mathbf{T}_+\mathbf{u} - \max\{\mathbf{0}, \mathbf{D}_{\kappa+}^{-1}(\mathbf{s}_{t+} - \mathbf{g}_+)\} + \max\{\mathbf{0}, -\mathbf{D}_{\kappa+}^{-1}(\mathbf{s}_{t+} + \mathbf{g}_+)\}$, $\mathbf{G}_3(\mathbf{y}) = \rho\mathbf{T}_0\mathbf{u} - \max\{\mathbf{0}, \mathbf{s}_{t0} - \mathbf{g}_0 + \rho\mathbf{T}_0\mathbf{u}\} + \max\{\mathbf{0}, -\mathbf{s}_{t0} - \mathbf{g}_0 - \rho\mathbf{T}_0\mathbf{u}\}$, $\mathbf{G}_4(\mathbf{y}) = \mathbf{Nu}$, $\mathbf{G}_5(\mathbf{y}) = \mathbf{Bu} - \mathbf{Ep} - \mathbf{c}$, and $\mathcal{N}_+ = \{i \in \mathcal{N} : \kappa_i > 0\}$, $\mathcal{N}_0 = \{i \in \mathcal{N} : \kappa_i = 0\}$, $\mathbf{T}_+ := \mathbf{T}_{\mathcal{N}_+}$, $\mathbf{T}_0 := \mathbf{T}_{\mathcal{N}_0}$, $\mathbf{s}_{t+} := \mathbf{s}_{t_{\mathcal{N}_+}}$, $\mathbf{s}_{t0} := \mathbf{s}_{t_{\mathcal{N}_0}}$, $\mathbf{g}_+ := \mathbf{g}_{t_{\mathcal{N}_+}}$, $\mathbf{g}_0 := \mathbf{g}_{t_{\mathcal{N}_0}}$. Again we obtain equation that can be solved by a Newton-type method. Since max-functions appear, the Jacobian matrix to $\mathbf{G}$ does not exist. We use the semi-smooth Newton method as a solver again. The algorithm generates the sequence $\{\mathbf{y}^k\}$. For each $\mathbf{y}^k$ we define the active/inactive sets:

$$\begin{aligned} &\mathcal{I}_{t+}^+ = \{i \in \mathcal{N}_+ : s_{ti}^k - g_i \geq 0\}, && \mathcal{I}_{t+}^- = \{i \in \mathcal{N}_+ : s_{ti}^k + g_i \leq 0\}, \\ &\mathcal{I}_{t0}^+ = \{i \in \mathcal{N}_0 : s_{ti}^k - g_i + \rho(\mathbf{Tu})_i \geq 0\}, && \mathcal{I}_{t0}^- = \{i \in \mathcal{N}_+ : s_{ti}^k + g_i + \rho(\mathbf{Tu})_i \leq 0\}, \\ &\mathcal{A}_0 = \mathcal{N}_0 \backslash (\mathcal{I}_{t0}^+ \cup \mathcal{I}_{t0}^-), \end{aligned}$$

and the indicator matrices in the same way as in Section 3. The new iterate $\mathbf{y}^{k+1}$ is computed by solving the following linear system:

$$\left(\begin{array}{c|ccccc} \mathbf{A} & \mathbf{T}_+^T & \mathbf{T}_{\mathcal{A}_0}^T & \mathbf{N}^T & \mathbf{B}^T \\ \hline \mathbf{T}_0 & -\mathbf{D}_{\kappa+}^{-1}\mathbf{D}(\mathcal{I}_{t+}^+ \cup \mathcal{I}_{t+}^-) & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{T}_+ & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{N} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{B} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{E} \end{array}\right) \begin{pmatrix} \mathbf{u}^{k+1} \\ \mathbf{s}_{t+}^{k+1} \\ \mathbf{s}_{t\mathcal{A}_0}^{k+1} \\ \boldsymbol{\lambda}_n^{k+1} \\ \mathbf{p}^{k+1} \end{pmatrix} = \begin{pmatrix} \mathbf{b} - \mathbf{T}_{\mathcal{I}_{t0}^+}^T\mathbf{g}_{\mathcal{I}_{t0}^+} + \mathbf{T}_{\mathcal{I}_{t0}^+}^T\mathbf{g}_{\mathcal{I}_{t0}^-} \\ \mathbf{D}_{\kappa+}^{-1}(\mathbf{D}(\mathcal{I}_{t+}^-) - \mathbf{D}(\mathcal{I}_{t+}^+))\mathbf{g}_+ \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{c} \end{pmatrix}. \quad (8)$$

To solve (8), we use again the Schur complement $\mathbf{S}^k = \mathbf{CA}^{-1}\mathbf{C}^T + \mathbf{D}^k$ with

$$\mathbf{C} = (\mathbf{T}_+^T, \mathbf{T}_{\mathcal{A}_0}^T, \mathbf{N}^T, \mathbf{B}^T)^T, \quad \mathbf{D}^k = \operatorname{diag}(\mathbf{D}_{\kappa+}^{-1}\mathbf{D}(\mathcal{I}_{t+}^+ \cup \mathcal{I}_{t+}^-), \mathbf{0}, \mathbf{0}, \mathbf{E}).$$

The conjugate gradient method with the adaptive precision control is used again in computations as well as the preconditioner described in Section 3.

5 Numerical experiments

We consider the L-shaped domain $\Omega = (0,5) \times (0,2) \backslash \bar{\mathcal{S}}$, $\mathcal{S} = (0,1) \times (0,1)$ with $\nu = 1$ and $\mathbf{f} = \mathbf{0}$; $\gamma_D = \gamma_{top} \cup \gamma_{left}$ with $\gamma_{top} = (0,5) \times \{2\}$, $\gamma_{left} = \{0\} \times (1,2)$, $\mathbf{u}_{D|\gamma_{top}} = \mathbf{0}$, and $\mathbf{u}_{D|\gamma_{left}} = (4(x_2 - 2)(1 - x_2), 0)$; $\gamma_N = \{5\} \times (0,2)$ with $\boldsymbol{\sigma}_N = \mathbf{0}$; $\gamma_C = \partial\Omega \backslash (\gamma_D \cup \gamma_N)$ with $g = 1$. In tables below we present $iter/n_S$, where $iter$ is the number of the Newton iterations and n_S is the total number of the matrix-vector multiplications by the Schur complements. In other words, n_S determines the computational efficiency. We illustrate the efficiency of both algorithms introduced in Sections 3 and 4. Table 1 and Table 4 show the computational efficiency without preconditioning, while Table 2 and Table 5 with preconditioning for each algorithm, respectively. It is easy to observe that n_S increases considerably for finer meshes and smaller κ in both cases and that this effect is eliminated by preconditioning. Table 3 and Table 6 display the computational efficiency of the interior point method. In comparison with the semi-smooth Newton method we see that the interior point method requires more matrix-vector multiplications.

$n_u/n_p/n_c$	$\kappa = 1$	$\kappa = 0.5$	$\kappa = 0.1$	$\kappa = 0.01$	$\kappa = 0.001$
344/206/32	8/1313	8/1197	9/1755	10/2375	*
1352/744/64	8/2727	9/3776	9/4684	10/6559	11/8858
5366/2819/128	9/6539	9/7314	10/11801	11/19484	11/17000
21386/10965/256	9/13064	10/16384	10/23394	11/28977	*
85394/43241/512	10/23324	10/29940	10/36387	13/62109	*

Table 1: The computational complexity for different $\kappa > 0$ without preconditioning.

$n_u/n_p/n_c$	$\kappa = 1$	$\kappa = 0.5$	$\kappa = 0.1$	$\kappa = 0.01$	$\kappa = 0.001$
344/206/32	7/108	8/140	8/113	8/113	10/143
1352/744/64	7/139	8/182	8/163	9/171	10/189
5366/2819/128	8/249	8/262	9/231	10/268	12/340
21386/10965/256	8/286	8/258	9/271	11/339	10/262
85394/43241/512	8/378	8/322	8/248	11/444	11/434

Table 2: The computational complexity for different $\kappa > 0$ with preconditioning

$n_u/n_p/n_c$	$\kappa = 1$	$\kappa = 0.5$	$\kappa = 0.1$	$\kappa = 0.01$	$\kappa = 0.001$
344/206/32	18/347	18/351	19/397	18/381	18/387
1352/744/64	20/494	20/486	20/503	20/491	20/483
5366/2819/128	18/500	18/445	19/521	19/494	19/476
21386/10965/256	19/605	19/565	19/608	19/557	19/608
85394/43241/512	19/687	19/690	19/742	19/705	19/671

Table 3: The computational complexity for different $\kappa > 0$ for interior point method

$n_u/n_p/n_c$	$\kappa = 1$	$\kappa = 0.5$	$\kappa = 0.1$	$\kappa = 0.01$	$\kappa = 0.001$	$\kappa = 0$
344/206/32	9/1377	9/1467	10/2001	11/2770	11/2837	12/1054
1352/744/64	10/3698	10/4210	11/6850	11/7429	12/9523	12/1686
5366/2819/128	9/6305	10/9021	10/1161	11/16596	13/27768	15/3942
21386/10965/256	10/14848	10/16680	11/29878	12/38700	*	10/5386
85394/43241/512	10/21789	10/27256	11/41416	*	*	*

Table 4: The computational complexity for different $\kappa \geq 0$ without preconditioning.

$n_u/n_p/n_c$	$\kappa = 1$	$\kappa = 0.5$	$\kappa = 0.1$	$\kappa = 0.01$	$\kappa = 0.001$	$\kappa = 0$
344/206/32	8/146	8/153	9/159	10/164	10/149	8/150
1352/744/64	8/199	9/222	10/256	10/205	11/223	8/168
5366/2819/128	9/269	10/319	10/298	11/304	12/326	8/207
21386/10965/256	9/342	8/247	10/340	11/398	12/404	8/290
85394/43241/512	9/353	9/346	9/349	11/522	12/509	9/477

Table 5: The computational complexity for different $\kappa \geq 0$ with preconditioning

$n_u/n_p/n_c$	$\kappa = 1$	$\kappa = 0.5$	$\kappa = 0.1$	$\kappa = 0.01$	$\kappa = 0.001$	$\kappa = 0$
344/206/32	18/344	18/357	19/367	18/382	19/381	19/410
1352/744/64	20/491	20/455	20/480	20/492	20/471	20/483
5366/2819/128	18/454	19/505	19/473	19/487	19/483	19/507
21386/10965/256	19/581	20/625	19/561	19/579	19/560	19/561
85394/43241/512	19/733	19/733	19/769	19/743	19/691	19/677

Table 6: The computational complexity for different $\kappa \geq 0$ for interior point method

Conclusion

Summarizing and using the results from [9], we succesfully modified the algorithm to accept $\kappa = 0$ as well, increasing its potential use. Furthermore, we have shown that the preconditioning has an important impact on the computational efficiency and that without it, it is impossible to solve larger linear systems.

References

[1] M.Ayadi, L. Baffico, M. K. Gdoura, T. Sassi, *Error estimates for Stokes problem with Tresca friction conditions,* ESAIM: Mathematical Modelling and Numerical Analysis, **48**, 1413 - 1429 (2014).

[2] J. K. Djoko, M. Mbehou, *Finite element analysis for Stokes and Navier Stokes equations driven by threshold slip boundary conditions,* Int. J. Numer. Anal. Model., Series B, **4**, 235-255 (2013).

[3] H. Fujita, *A mathematical analysis of motions of viscous incompressible fluid under leak and slip boundary conditions,* RIMS Kokyuroku, **888**, 199-216 (1994).

[4] J. Haslinger, R. Kučera, and V. Šátek, *Stokes system with selection-dependant threshold slip boundary conditions: Analysis, approximation and implementation,* accepted in Mathematics and Mechanics of Solids (2017).

[5] H. Hintermüller, K. Ito, K. Kunisch, *The primal-dual active set strategy as a semismooth Newton method,* SIAM J. Optim., **13**, 865-888 (2003).

[6] R. Kučera, J. Machalová, H. Netuka, P. Ženčák, *An interior point algorithm for the minimization arising from 3d contact problems with fricion,* Optimization Methods and Software **28**, 1195-1217 (2013).

[7] R. Kučera, J. Haslinger, V. Šátek, M. Jarošová, *Efficient methods for solving the Stokes problem with slip boundary conditions,* accepted in Mathematics and Computers in Simulation (2017), `https://doi.org/10.1016/j.matcom.2016.05.012`.

[8] J. Koko, *Vectorized Matlab codes for the Stokes problem with P1-bubble/P1 finite element,* at: `https://www.isima.fr/~jkoko/Codes/StokesP1BubbleP1.pdf`.

[9] J. Pacholek, *Semi-smooth Newton method for solving the Stokes equations with monotonously increasing slip condition,* Diploma thesis, VŠB-TU Ostrava (2017).

[10] I. Rao, K. Rajagopal, *The effect of the slip boundary conditions on the flow of fluids in a channel,* Acta Mechanica **135**, 113-126 (1999).

Nehladká Newtonova metoda pro řešení Stokesových rovnic se skluzovou okrajovou podmínkou

Abstrakt (Streszczenie): Tento text se zabývá řešením Stokesových rovnic s monotónně rostoucí skluzovou podmínkou. Použitím P1-bubble/P1 aproximace konečných prvků dostaneme algebraickou variační nerovnici, která je ekvivalentní jisté minimalizační úloze, jejíž podmínky optimality jsou výchozím bodem pro návrh algoritmu. Použitým algoritmem je implementace nehladké Newtonovy metody, založená na použití aktivních a neaktivních množin, kde zmíníme 2 možné postupy konstrukce. Algoritmus je poté testován v prostředí MATLAB. Experimenty jsou provedeny na čtvercové a „L-shaped“ oblasti, přičemž studujeme vlivy koeficientu přilnavosti a předpodmínění na efektivitu výpočtů.

Klíčová slova (Słowa kluczowe): nehladká Newtonova metoda, skluzová podmínka, Stokesův problém, předpodmínění