Optimization Methods and Software
Vol. 00, No. 00, May 2011, 1–23

An interior-point algorithm for the minimization arising from 3D contact problems with friction

R. Kučera[a], J. Machalová[b], H. Netuka[b*], P. Ženčák[b]

[a] *Centre of Excellence IT4I, VŠB-TU Ostrava, 17. listopadu 15/2172, 70833 Ostrava, CZ*;
[b] *Department of Mathematical Analysis and Applications of Mathematics, Faculty of Science, Palacký University Olomouc, 17. listopadu 1192/12, 77146 Olomouc, CZ*

(*received May 2011*)

The paper deals with a variant of the interior-point method for the minimization of strictly quadratic objective function subject to simple bounds and separable quadratic inequality constraints. Such minimizations arise from the finite element approximation of contact problems of linear elasticity with friction in three space dimensions. The main goal of the paper is the convergence analysis of the algorithm and its implementation. The optimal preconditioners for solving ill-conditioned inner linear systems are proposed. Numerical experiments illustrate the computational efficiency for large scale problems.

Keywords: interior-point algorithm; convergence; preconditioners; contact problems; friction

AMS Subject Classification: 65K05; 90C51; 74M10

1. Introduction

We consider the problem

$$\min_{x\in\Omega} \; q(x) \tag{1}$$

with $q(x) = \frac{1}{2}\, x^\top Ax - x^\top b$ and $\Omega = \{x \in \mathbb{R}^{3m} :\; x_i \geq l_i,\; x_{i+m}^2 + x_{i+2m}^2 \leq g_i^2,\; i = 1, \ldots, m\}$, where $n = 3m$, $A \in \mathbb{R}^{n\times n}$ is symmetric, positive definite, $b \in \mathbb{R}^n$, and $l_i \in \mathbb{R}$, $g_i \in \mathbb{R}_+$, $i = 1, \ldots, m$. Such minimizations arise, e.g., from the finite element approximation of contact problems of linear elasticity with friction in three space dimensions (3D) [15, 17]. If the Coulomb friction law is considered, then the fixed-point approach leads to iterations given as a sequence of problems (1). There are at least three important reasons why to solve in this case also (1) by an iterative method. *(i)* The requirements on the accuracy of the computed solutions to (1) are different, they increase successively. *(ii)* The initial guess of the solution to (1) is known from the previous fixed-point step. *(iii)* The matrix A is not assembled explicitly, it is given as a product of other matrices so that only matrix-vector products with A may be performed. Moreover, one can combine both iterative processes in one (inexact) loop so that the resulting algorithm may be an efficient tool for solving large scale problems.

*Corresponding author. Email: horymir.netuka@upol.cz

ISSN: 1055-6788 print/ISSN 1029-4937 online

DOI: 10.1080/1055678xxxxxxxxxxxxx
http://www.informaworld.com

The problem (1) can be solved by a general method, e.g. [2, 5, 6, 28]. Nevertheless, one can expect that a specialized algorithm taking into account the particular structure of (1) will be more efficient. The first author proposed such algorithm called KPRGP (KKT-Proportioning with Reduced Gradient Projections [10, 20, 21]), in which conjugate gradient method is combined with gradient projections in an active set strategy. It extends the algorithm of Dostál and Schöberl [9, 11] originally developed for simple bound problems. The common feature of these algorithms is the theory comprising the same convergence rate that enables us to prove the scalability of methods for solving 3D contact problems without [8, 9] and with [7, 10] friction. However, the practical behavior may be different due to the difference in the finite termination property. After finding indices of simple bounds active in the solution, the simple bound problem reduces to a linear one, for which the conjugate gradient method completes the iterative process. Unfortunately, this property does not hold for the quadratic inequalities $x_{i+m}^2 + x_{i+2m}^2 \leq g_i^2$. Here, the algorithm seeks also positions of the pairs (x_{i+m}, x_{i+2m}) lying on the curved boundaries of the active circles. Therefore, "zig-zag iterations" may be generated so that short conjugate gradient sequences alternate with projection steps changing the active set. In order to overcome this drawback, one can recommend using a *strictly feasible* algorithm that does not work with any active set. Its typical representative is an *interior-point* method. We will deal with its *primal-dual* variant that is usually faster and more reliable than the pure primal or pure dual methods [29] and it is also efficient when the high accuracy of the solution is needed [5].

In the paper, we develop the *path-following* algorithm that was proposed in [13, 27] for solving linear programming problems. The main idea consists in applying the Newton iterations to solve equations in the system of the Karush-Kuhn-Tucker (KKT) conditions to (1). Since some unknowns in the KKT system are constrained by simple bounds, the Newton steps are damped. Another key ingredient is the centering that keeps iterations deeper in the feasible region so that longer steps may be performed. The most expensive part of each iteration is the solution of an indefinite linear system. To this end, we apply methods based on the Schur complement reduction so that reduced linear systems are solved by the conjugate gradient method. As the reduced matrices are typically ill-conditioned, preconditioners are needed. Our preconditioners are optimal in the sense that condition numbers of the preconditioned matrices are bounded by a constant multiple of the condition number of A. The total efficiency is increased by a precision control terminating adaptively the inner conjugate gradient iterations.

The rest of the paper is organized as follows. In Section 2 we introduce preliminary results comprising a suitable reformulation of the KKT conditions. In Section 3, we propose the algorithm. Its convergence is proved in Section 4. The implementation of the algorithm and the spectral analysis of the preconditioners are discussed in Section 5. Finally, Section 6 summarizes results of numerical experiments and Section 7 gives comments and concluding remarks.

Let us introduce some conventions that we use through the whole paper. The symbols $\mathcal{M}_1$ and $\mathcal{M}_2$ stand for the following sets of indices: $\mathcal{M}_1 = \{1, 2, \ldots, m\}$ and $\mathcal{M}_2 = \{1, 2, \ldots, 2m\}$. The Euclidean norm of $x \in \mathbb{R}^p$ is denoted by $\|x\| = (x_1^2 + \cdots + x_p^2)^{1/2}$. The condition number of a symmetric, positive definite matrix $M \in \mathbb{R}^{p\times p}$ is given by $\kappa(M) = \lambda_{\max}/\lambda_{\min}$, where $\lambda_{\max}$ and $\lambda_{\min}$ are the smallest and the largest eigenvalue of M, respectively. The symbol *diag* defines a diagonal matrix as follows: if $x \in \mathbb{R}^p$ or $X \in \mathbb{R}^{p\times p}$, the diagonal entries of $diag(x) \in \mathbb{R}^{p\times p}$ or $diag(X) \in \mathbb{R}^{p\times p}$ are given by x or by the diagonal entries of X, respectively. By I and 0 we denote the identity and the zero matrix, respectively. Let $L : \mathbb{R}^p \mapsto \mathbb{R}$, $L = L(x)$, $x \in \mathbb{R}^p$, be a differentiable function and let y be a subset of its variables x.

The symbol $\nabla_y L$ stands for the partial gradient of L whose components are given by y.

2. Notation and preliminaries

As (1) is the minimization of the strictly quadratic objective function q on the convex set Ω, the solution exists and it is necessarily unique [5]. We denote it by x^*. Let us introduce the Lagrangian $L : \mathbb{R}^n \times \mathbb{R}^{2m} \mapsto \mathbb{R}$ associated with (1) by

$$L(x,\nu) = q(x) + \sum_{i\in\mathcal{M}_1} \nu_i(l_i - x_i) + \sum_{i\in\mathcal{M}_1} \nu_{i+m}(x_{i+m}^2 + x_{i+2m}^2 - g_i^2),$$

where $\nu = (\nu_1, \ldots, \nu_{2m})^\top \in \mathbb{R}^{2m}$ is the Lagrange multiplier vector to the constraints in Ω. The solution x^* is the first component of the saddle-point $(x^*, \nu^*) \in \mathbb{R}^n \times \mathbb{R}^{2m}$ to L that is fully determined by the KKT conditions [5]:

$$\nabla_x L(x,\nu) = 0,\ \nabla_\nu L(x,\nu) \le 0,\ \nu \ge 0,\ \nu^\top \nabla_\nu L(x,\nu) = 0. \tag{2}$$

Introducing the new variable $z = -\nabla_\nu L(x,\nu)$, $z \in \mathbb{R}^{2m}$, (2) is equivalent to:

$$\nabla_x L(x,\nu) = 0,\ \nabla_\nu L(x,\nu) + z = 0,\ \nu^\top z = 0,\ \nu \ge 0,\ z \ge 0. \tag{3}$$

Let us define the function $F : \mathbb{R}^{n+4m} \mapsto \mathbb{R}^{n+4m}$ by

$$F(v) = (\nabla_x L(x,\nu)^\top, (\nabla_\nu L(x,\nu) + z)^\top, e^\top NZ)^\top,$$

where $v = (x^\top, \nu^\top, z^\top)^\top \in \mathbb{R}^{n+4m}$, $N = diag(\nu)$, $Z = diag(z)$, and $e \in \mathbb{R}^{2m}$ is the vector of all ones.

LEMMA 2.1 *The solution x^* to (1) is the first component of the solution $v^* = (x^{*\top}, \nu^{*\top}, z^{*\top})^\top$ to*

$$F(v) = 0,\ \nu \ge 0,\ z \ge 0. \tag{4}$$

Proof (4) is equivalent to (3). ■

The Jacobi matrix $J = J(v)$ to F at v reads as follows:

$$J = \begin{pmatrix} J_{11} & J_{12} & 0 \\ J_{21} & 0 & I \\ 0 & Z & N \end{pmatrix}, \tag{5}$$

where

$$J_{11} = A + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 2N_2 & 0 \\ 0 & 0 & 2N_2 \end{pmatrix}, \quad J_{12} = J_{21}^\top = \begin{pmatrix} -I & 0 \\ 0 & 2X_2 \\ 0 & 2X_3 \end{pmatrix}$$

with $N_2 = diag(\nu_{m+1}, \ldots, \nu_{2m})$ and $X_k = diag(x_{(k-1)m+1}, \ldots, x_{km})$, $k = 2, 3$.

LEMMA 2.2 *If $z > 0$, then $J = J(v)$ is non-singular for any x and $\nu \ge 0$.*

Proof The non-singularity of a matrix is guaranteed by the non-singularity of its diagonal block and the respective Schur complement [3]. The block $\begin{pmatrix} 0 & I \\ Z & N \end{pmatrix}$ in J is non-singular due to $z > 0$. The Schur complement to this block reads as $J_{11} + J_{12}Z^{-1}NJ_{21}$. It is the positive definite matrix, since A is positive definite and $\nu \geq 0$, $z > 0$. ■

Remark 1 The Jacobi matrix may be singular at the solution v^*. This situation arises, e.g., when $\nu_i^* = 0$ and $z_i^* = 0$ for at least one $i \in \mathcal{M}_2$, since the corresponding row in $J(v^*)$ vanishes.

Let $v^{(k)} = ({x^{(k)}}^\top, {\nu^{(k)}}^\top, {z^{(k)}}^\top)^\top$ with $\nu^{(k)} > 0$, $z^{(k)} > 0$ be a known approximation of v^*. The damped Newton method consists in computing the standard Newton direction $\Delta v^{(k+1)}$ from the linear system given by the Jacobi matrix and, then, in defining the new iteration $v^{(k+1)}$ in this direction using a steplength $\alpha_k \in (0, 1]$:

$$J(v^{(k)})\Delta v^{(k+1)} = -F(v^{(k)}), \quad v^{(k+1)} = v^{(k)} + \alpha_k \Delta v^{(k+1)}. \tag{6}$$

By the choice of α_k one can guarantee positiveness of the components $\nu^{(k+1)}$, $z^{(k+1)}$ of $v^{(k+1)}$. Unfortunately, the computations based directly on (6) can take short steps before violating conditions $\nu^{(k+1)} > 0$, $z^{(k+1)} > 0$ so that the convergence rate may be slow. In the next section we modify these iterations.

3. Algorithm

Let us replace (4) by

$$F(v) = (0^\top, 0^\top, \tau e^\top)^\top, \ \nu > 0, \ z > 0, \tag{7}$$

where $\tau > 0$. Solutions v^τ to (7) define in $\mathbb{R}^{n+4m}$ a curve $\mathcal{C}(\tau)$ called the *central path*. This curve leads to v^*, when τ tends to zero. In the algorithm, we combine the Newton method applied to the equation in (7) with changes of τ so that the iterations follow $\mathcal{C}(\tau)$. To this end, we define the *neighborhood* of the central path by

$$\begin{aligned} \mathcal{N}(\gamma, \beta) = \{v = (x^\top, \nu^\top, z^\top)^\top \in \mathbb{R}^{n+4m} : \ & \|\nabla_x L(x, \nu)\| \leq \beta\vartheta, \\ & \|\nabla_\nu L(x, \nu) + z\| \leq \beta\vartheta, \ \nu \geq 0, \ z \geq 0, \ \nu_i z_i \geq \gamma\vartheta, \ i \in \mathcal{M}_2\}, \end{aligned} \tag{8}$$

where $\beta \geq 0$, $\gamma \in (0, 1]$, and

$$\vartheta = \vartheta(v) = \frac{\nu^\top z}{2m}.$$

It is readily seen that $\mathcal{N}(\gamma, \beta) \supseteq \mathcal{C}(\tau)$ and $\mathcal{N}(1, 0) = \mathcal{C}(\tau)$.

LEMMA 3.1 *Let $v \in \mathcal{N}(\gamma, \beta)$ and let either $\nu_j = 0$ or $z_j = 0$ for at least one $j \in \mathcal{M}_2$. Then v is the solution to (4).*

Proof As $\nu \geq 0$ and $z \geq 0$, the inequality $0 = \nu_j z_j \geq \gamma\vartheta \geq 0$ yields $\vartheta = 0$. The remaining inequalities in the definition of $\mathcal{N}(\gamma, \beta)$ imply $\nabla_x L(x, \nu) = 0$ and $\nabla_\nu L(x, \nu) + z = 0$. Therefore, (3) holds that proves the lemma. ■

In the kth iteration of the path-following (PF) algorithm, we modify τ by the product of $\vartheta_k = \vartheta(v^{(k)})$ with the centering parameter $\sigma_k \in [\sigma_{\min}, \sigma_{\max}]$, $0 \leq \sigma_{\min} \leq \sigma_{\max} \leq 1$. The value $\sigma_k = 0$ leads to the standard Newton direction, while $\sigma_k = 1$ gives the fully centered direction. The algorithm uses also the *Armijo-type condition* (10) ensuring that the sequence $\{\vartheta_k\}$ is decreasing (see [29]). In the initialization section of the algorithm, we introduce restrictions on the values of β and σ that are required by the convergence analysis.

ALGORITHM PF: Given $\gamma \in (0, 1]$, $\beta \geq 1$, $0 < \sigma_{\min} \leq \sigma_{\max} \leq 1/2$, $\omega \in (0, 1)$, and $\epsilon \geq 0$. Let $v^{(0)} \in \mathcal{N}(\gamma, \beta)$ and set $k := 0$.

(1°) Choose $\sigma_k \in [\sigma_{\min}, \sigma_{\max}]$.

(2°) If $\nu^{(k)} > 0$ and $z^{(k)} > 0$, solve

$$J(v^{(k)})\Delta v^{(k+1)} = -F(v^{(k)}) + (0^\top, 0^\top, \sigma_k \vartheta_k e^\top)^\top, \tag{9}$$

else set $\Delta v^{(k+1)} = 0$.

(3°) Set

$$v^{(k+1)} = v^{(k)} + \alpha_k \Delta v^{(k+1)}$$

using the largest $\alpha_k \in (0, 1]$ satisfying $v^{(k+1)} \in \mathcal{N}(\gamma, \beta)$ and

$$\vartheta_{k+1} \leq (1 - \alpha_k \omega(1 - \sigma_k))\vartheta_k. \tag{10}$$

(4°) Return $\bar{v} = v^{(k+1)}$, when

$$err^{(k)} := \|v^{(k+1)} - v^{(k)}\| / \|v^{(k+1)}\| \leq \epsilon, \tag{11}$$

else set $k := k + 1$ and go to step (1°).

Let us note that the algorithm is well-defined. Since $z^{(k)} > 0$, Lemma 2.2 guarantees the non-singularity of the Jacobi matrix in (9). The existence of α_k satisfying all requirements of step (3°) will be proved in Lemma 4.2. The typical values of the parameters and the choice of the initial iteration will be discussed in Section 5. The algorithm extends, in some sense, ideas used in [29] for solving linear programming problems. The main change of our implementation consists in the necessity to solve inner linear systems (9) whose spectral properties depend on the matrix A. Moreover, the convergence analysis requires to add the norm terms in the definition of $\mathcal{N}(\gamma, \beta)$ and to use the Armijo-type condition (10).

4. Convergence analysis

To prove the convergence of ALGORITHM PF we modify the ideas of [26, 30], where analogous path-following algorithms are analyzed for the solution of discrete variational inequality or nonlinear complementarity problems, respectively. Our modification leads to different restrictions on the parameters of the algorithm. Moreover, we remove the lack of the analysis by adding the assumption in Theorem 4.3 on the boundedness of the sequence generated by ALGORITHM PF.

LEMMA 4.1 *Let $v^{(k)} \in \mathcal{N}(\gamma,\beta)$ with $z^{(k)} > 0$ and let $\Delta v^{(k+1)}$ satisfy (9). It holds:*

$$z_i^{(k)}\Delta\nu_i^{(k+1)} + \nu_i^{(k)}\Delta z_i^{(k+1)} = -\nu_i^{(k)}z_i^{(k)} + \sigma_k\vartheta_k, \qquad i \in \mathcal{M}_2.$$

Proof Using notation introduced in (5), the third block equation in (9) reads as $Z^{(k)}\Delta\nu^{(k+1)} + N^{(k)}\Delta z^{(k+1)} = -N^{(k)}Z^{(k)}e + \sigma_k\vartheta_k e$ that proves the lemma. ■

In the next lemma, we denote by $\mathcal{B}(\hat{v},\hat{\delta})$ the closed ball in $\mathbb{R}^{n+4m}$ with the center at $\hat{v}$ and the radius $\hat{\delta} > 0$.

LEMMA 4.2 *Let $\gamma \in (0,1]$, $\beta \geq 1$, $0 < \sigma_{\min} \leq \sigma_{\max} \leq 1/2$, $\omega \in (0,1)$, and $\hat{v} \in \mathcal{N}(\gamma,\beta)$, $\hat{v} \neq v^*$. There are $\hat{\delta} > 0$ and $\hat{\alpha} \in (0,1]$ such that the steplength α_k in step (3°) of* ALGORITHM PF *satisfies $\alpha_k \geq \hat{\alpha}$ for any $v^{(k)} \in \mathcal{N}(\gamma,\beta) \cap \mathcal{B}(\hat{v},\hat{\delta})$.*

Proof Let us define

$$\hat{\delta} = \frac{1}{2}\min_{i\in\mathcal{M}_2}\{\min\{\hat{\nu}_i,\hat{z}_i\}\}. \tag{12}$$

Note that $\hat{\delta} > 0$ by Lemma 2.2. For the sake of simplicity, we omit the iterative index k in the proof and we denote $v(\alpha) = v + \alpha\Delta v$ that induces also notation $x(\alpha)$, $\nu(\alpha)$, $z(\alpha)$, and $\vartheta(\alpha) = \vartheta(v(\alpha))$. The value of $\hat{\alpha}$ will be defined successively so that we will examine lower bounds for $\alpha = \alpha_k$ given by the requirements of step (3°), i.e.:

$$\nu(\alpha), z(\alpha) \geq 0, \tag{13}$$

$$\nu_i(\alpha)z_i(\alpha) \geq \gamma\vartheta(\alpha), \ i \in \mathcal{M}_2, \tag{14}$$

$$\vartheta(\alpha) \leq (1 - \alpha\omega(1-\sigma))\vartheta, \tag{15}$$

$$\|\nabla_x L(x(\alpha),\nu(\alpha))\| \leq \beta\vartheta(\alpha), \tag{16}$$

$$\|\nabla_\nu L(x(\alpha),\nu(\alpha)) + z(\alpha)\| \leq \beta\vartheta(\alpha). \tag{17}$$

First of all we introduce some auxiliary results. The definition (12) implies $\hat{\nu}_i - \hat{\delta} \geq \hat{\delta}$ and $\hat{z}_i - \hat{\delta} \geq \hat{\delta}$ and, since $v \in \mathcal{B}(\hat{v},\hat{\delta})$, we get also $\hat{\nu}_i - \nu_i \leq \hat{\delta}$, $\hat{z}_i - z_i \leq \hat{\delta}$. Therefore,

$$\nu_i z_i \geq (\hat{\nu}_i - \hat{\delta})(\hat{z}_i - \hat{\delta}) \geq \hat{\delta}^2, \quad i \in \mathcal{M}_2, \tag{18}$$

and, consequently,

$$\vartheta \geq \hat{\delta}^2. \tag{19}$$

As $z > 0$ by (18), the Jacobi matrix $J(v)$ is non-singular by Lemma 2.2 and its inversion as well as the right-hand side in (9) are the continuous functions on $\mathcal{B}(\hat{v},\hat{\delta})$. Thus, there is a constant $C > 0$ such that

$$\|\Delta v\| \leq C \tag{20}$$

for all $v \in \mathcal{B}(\hat{v},\hat{\delta})$ and $\sigma \in [\sigma_{\min},\sigma_{\max}]$. Recall that due to definition (8), it holds

$$\nu_i z_i \geq \gamma\vartheta, \qquad i \in \mathcal{M}_2. \tag{21}$$

Using Lemma 4.1, we obtain

$$\begin{aligned}\nu_i(\alpha)z_i(\alpha) &= \nu_i z_i + \alpha(z_i\Delta\nu_i + \nu_i\Delta z_i) + \alpha^2\Delta\nu_i\Delta z_i \\ &= (1-\alpha)\nu_i z_i + \alpha\sigma\vartheta + \alpha^2\Delta\nu_i\Delta z_i \end{aligned} \tag{22}$$

that gives $\nu(\alpha)^\top z(\alpha) = (1-\alpha)\nu^\top z + \alpha\sigma\nu^\top z + \alpha^2\Delta\nu^\top\Delta z$. By the Cauchy–Schwarz inequality and (20), we arrive at

$$(1-\alpha)\nu^\top z + \alpha\sigma\nu^\top z - \alpha^2C^2 \le \nu(\alpha)^\top z(\alpha) \le (1-\alpha)\nu^\top z + \alpha\sigma\nu^\top z + \alpha^2C^2 \tag{23}$$

and

$$(1-\alpha)\vartheta + \alpha\sigma\vartheta - \alpha^2\frac{C^2}{2m} \le \vartheta(\alpha) \le (1-\alpha)\vartheta + \alpha\sigma\vartheta + \alpha^2\frac{C^2}{2m}. \tag{24}$$

Note that (20) yields also $-C \le \Delta\nu_i \le C$ and $-C \le \Delta z_i \le C$. Therefore, (22) and (21) imply

$$\nu_i(\alpha)z_i(\alpha) \ge (1-\alpha)\gamma\vartheta + \alpha\sigma\vartheta - \alpha^2C^2. \tag{25}$$

Let us define $\hat\alpha_1 = \min\{1, \hat\delta/(2C)\}$ and let $\alpha \in (0, \hat\alpha_1]$. Using $\nu_i \ge \hat\nu_i - \hat\delta \ge \hat\delta$ and (20), we get

$$\nu_i(\alpha) = \nu_i + \alpha\Delta\nu_i \ge \hat\delta - \frac{\hat\delta}{2C}|\Delta\nu_i| \ge \frac{\hat\delta}{2} > 0, \qquad i \in \mathcal{M}_2.$$

Since the same bound holds for $z_i(\alpha)$, the inequalities (13) are satisfied.

Further, let us define $\hat\alpha_2 = \min\{\hat\alpha_1, \sigma_{\min}(1-\gamma)\hat\delta^2/(2C^2)\}$ and let $\alpha \in (0, \hat\alpha_2]$. Then $\alpha^2 \le \alpha\sigma(1-\gamma)\vartheta/(2C^2)$ due to (19) so that

$$\alpha\sigma\vartheta - \alpha^2C^2 \ge \alpha\sigma\gamma\vartheta + \alpha^2C^2 \ge \alpha\sigma\gamma\vartheta + \alpha^2\gamma\frac{C^2}{2m}.$$

Using this result and the right inequality (24) in (25), we get

$$\nu_i(\alpha)z_i(\alpha) \ge (1-\alpha)\gamma\vartheta + \alpha\sigma\gamma\vartheta + \alpha^2\gamma\frac{C^2}{2m} \ge \gamma\vartheta(\alpha), \qquad i \in \mathcal{M}_2.$$

We have proved (14).

Let us consider $\alpha \in (0, \hat\alpha_3]$ with $\hat\alpha_3 = \min\{\hat\alpha_2, m(1-\omega)\hat\delta^2/C^2\}$. Then $\alpha^2C^2 = \alpha(\alpha C^2) \le \alpha m(1-\omega)\hat\delta^2$ together with (19) and $\sigma \le 1/2$ give

$$\alpha^2C^2 \le \alpha m(1-\omega)\vartheta = \alpha(1-\omega)\frac{\nu^\top z}{2} \le \alpha(1-\omega)(1-\sigma)\nu^\top z.$$

Applying this result in (23), we arrive at

$$\nu(\alpha)^\top z(\alpha) \le \nu^\top z - \alpha(1-\sigma)\nu^\top z + \alpha^2C^2 \le [1-\alpha\omega(1-\sigma)]\nu^\top z \tag{26}$$

that proves (15).

It remains to show when the conditions (16) and (17) are satisfied. For convenience, we denote $r_x(x,\nu) = \nabla_x L(x,\nu)$ and $r_\nu(x,z) = \nabla_\nu L(x,\nu) + z$.

Let us consider $\alpha \in (0, \hat{\alpha}_4]$ with $\hat{\alpha}_4 = \min\{\hat{\alpha}_3, \sigma_{\min}\hat{\delta}^2/(3C^2)\}$. Using (19), we get

$$\sigma\vartheta - \alpha\frac{C^2}{2m} - 2\alpha C^2 \geq \sigma\vartheta - 3\alpha C^2 \geq \sigma\vartheta - \sigma_{\min}\hat{\delta}^2 \geq 0. \tag{27}$$

We split r_x on three components of the same length m, i.e., $r_x = (r_{x_1}^\top, r_{x_2}^\top, r_{x_3}^\top,)^\top$. It is easily seen from the definition of L that r_{x_1} is the linear function while r_{x_2} and r_{x_3} are the non-linear functions. Therefore, the Taylor expansion of r_x reads as follows:

$$r_x(x(\alpha), \nu(\alpha)) = r_x(x,\nu) + \alpha(\nabla_x r_x(x,\nu)\Delta x + \nabla_\nu r_x(x,\nu)\Delta\nu) + \frac{\alpha^2}{2}s_x(\Delta x, \Delta\nu) \tag{28}$$

with $s_x = (s_{x_1}^\top, s_{x_2}^\top, s_{x_3}^\top)^\top$, where $s_{x_1} = 0$ and $s_{x_k,j}(\Delta x, \Delta\nu) = 4\Delta x_{j+(k-1)m}\Delta\nu_{j+m}$, $k = 2, 3$, $j \in \mathcal{M}_1$. Since (20) implies the boundness of Δx and the components of $\Delta\nu$ by C, it holds:

$$\begin{aligned}
\|s_x(\Delta x, \Delta\nu)\|^2 &= \sum_{j\in\mathcal{M}_1} \left((4\Delta x_{j+m}\Delta\nu_{j+m})^2 + (4\Delta x_{j+2m}\Delta\nu_{j+m})^2\right) \\
&\leq 16 \max_{i\in\mathcal{M}_1}\left\{(\Delta\nu_{i+m})^2\right\} \sum_{j\in\mathcal{M}_1}\left((\Delta x_{j+m})^2 + (\Delta x_{j+2m})^2\right) \\
&\leq 16C^2\|\Delta x\|^2 \leq 16C^4.
\end{aligned} \tag{29}$$

From the linear system (9), one can see that $\nabla_x r_x(x,\nu)\Delta x + \nabla_\nu r_x(x,\nu)\Delta\nu = -r_x(x,\nu)$. Substituting in (28) and using the norm, we get

$$\|r_x(x(\alpha),\nu(\alpha))\| \leq (1-\alpha)\|r_x(x,\nu)\| + \frac{\alpha^2}{2}\|s_x(\Delta x, \Delta\nu)\|.$$

The definition (8) and (29) yield

$$\|r_x(x(\alpha),\nu(\alpha))\| \leq (1-\alpha)\beta\vartheta + 2\alpha^2C^2. \tag{30}$$

The left inequality (24) implies

$$(1-\alpha)\vartheta \leq \vartheta(\alpha) - \alpha\sigma\vartheta + \alpha^2\frac{C^2}{2m}. \tag{31}$$

Using this result and the assumption $\beta \geq 1$ in (30), we arrive at

$$\|r_x(x(\alpha),\nu(\alpha))\| \leq \beta\vartheta(\alpha) - \beta\alpha(\sigma\vartheta - \alpha\frac{C^2}{2m} - 2\alpha C^2).$$

Finally, (27) implies $\|r_x(x(\alpha),\nu(\alpha))\| \leq \beta\vartheta(\alpha)$ that proves (16).

The proof of (17) is analogous. We split r_ν on two components of the same length m, i.e., $r_\nu = (r_{\nu_1}^\top, r_{\nu_2}^\top)^\top$. As for $r_x(x,\nu)$, we derive

$$\|r_\nu(x(\alpha), z(\alpha))\| \leq (1-\alpha)\beta\vartheta + \frac{\alpha^2}{2}\|s_\nu(\Delta x, \Delta z)\|, \tag{32}$$

where $s_\nu = (s_{\nu_1}^\top, s_{\nu_2}^\top)^\top$ is the quadratic term from the Taylor expansion of r_ν with the components $s_{\nu_1} = 0$ and $s_{\nu_2,j}(\Delta x, \Delta z) = 2(\Delta x_{j+m})^2 + 2(\Delta x_{j+2m})^2$, $j \in \mathcal{M}_1$. Using (20), we get

$$\begin{aligned}
\|s_\nu(\Delta x, \Delta z)\|^2 &= \sum_{j\in\mathcal{M}_1} \left(2(\Delta x_{j+m})^2 + 2(\Delta x_{j+2m})^2\right)^2 \\
&\le 4 \max_{i\in\mathcal{M}_1} \left\{(\Delta x_{i+1})^2, (\Delta x_{i+2m})^2\right\} \sum_{j\in\mathcal{M}_1} \left(|\Delta x_{j+m}| + |\Delta x_{j+2m}|\right)^2 \\
&\le 4C^2\Big(\sum_{j\in\mathcal{M}_1} (\Delta x_{j+m})^2 + 2 \sum_{j\in\mathcal{M}_1} |\Delta x_{j+m}||\Delta x_{j+2m}| + \sum_{j\in\mathcal{M}_1} (\Delta x_{j+2m})^2\Big) \\
&\le 4C^2 \left(\|\Delta x\|^2 + 2\|\Delta x\|\|\Delta x\| + \|\Delta x\|^2\right) \le 16C^4. \qquad (33)
\end{aligned}$$

The rest of the proof uses (31), (32), and (33) analogously as in the case (16). We obtain that (17) holds for $\alpha \in (0, \hat{\alpha}_4]$.

We have shown that all requirements (13)–(17) are satisfied for any $\alpha \in (0, \hat{\alpha}_4]$, $v \in \mathcal{B}(\hat{v}, \hat{\delta})$, and $\sigma \in [\sigma_{\min}, \sigma_{\max}]$. The steplength $\alpha_k = \alpha$ determined in step (3°) of Algorithm PF satisfies $\alpha_k \ge \hat{\alpha}$, where $\hat{\alpha} = \hat{\alpha}_4$. The proof is complete. ■

Theorem 4.3 *Let $\gamma \in (0,1]$, $\beta \ge 1$, $0 < \sigma_{\min} \le \sigma_{\max} \le 1/2$, $\omega \in (0,1)$, and $\epsilon = 0$. Let the sequence $\{v^{(k)}\}$ generated by* Algorithm PF *be bounded. Then, either $\{v^{(k)}\}$ is finite and its last element $\bar{v}$ is the solution to (4), or the limit point of $\{v^{(k)}\}$ is the solution to (4).*

Proof The algorithm terminates finitely, if at least one entry of $z^{(k)}$ vanishes for any $k \ge 0$. Then, Lemma 3.1 implies $\bar{v} = v^*$.

Let us assume that the sequence $\{v^{(k)}\}$ is infinite and consider its accumulation point $\hat{v} \in \mathcal{N}(\gamma, \beta)$. Let us suppose for contradiction that $\hat{v} \ne v^*$. Then Lemma 3.1 yields $\hat{\nu} > 0$, $\hat{z} > 0$ so that $\hat{\vartheta} = \hat{\nu}^\top \hat{z}/(2m) > 0$. Recall that the $(k+1)$th iteration satisfies

$$\vartheta_{k+1} \le \vartheta_k - \alpha_k \omega (1 - \sigma_k) \vartheta_k. \qquad (34)$$

Since $\{\vartheta_k\}$ is monotonically decreasing, we obtain

$$\vartheta_k \ge \hat{\vartheta} > 0, \qquad k \in \mathbb{N}. \qquad (35)$$

Denote by $\mathcal{I}$ an infinite subsequence of $\mathbb{N}$ such that $v^{(k)} \to \hat{v}$ for $k \to +\infty$ and $k \in \mathcal{I}$. By Lemma 4.2, there is $\hat{\alpha} \in (0,1]$ such that $\alpha_k \ge \hat{\alpha}$ for $k \in \mathcal{I}$ sufficiently large. Using this bound in (34) together with $\sigma_k \le 1/2$ and (35), we get $\vartheta_{k+1} \le \vartheta_k - \hat{\alpha}\omega\hat{\vartheta}/2$. Here, $\hat{\alpha}\omega\hat{\vartheta}/2$ is the positive constant independent of k. Since $\mathcal{I}$ is infinite and $\{\vartheta_k\}$ is monotonically decreasing, we arrive at $\vartheta_k \searrow -\infty$. This result contradicts (35) that completes the proof. ■

5. Implementation

In this section we describe the implementation of Algorithm PF for solving large scale problems.

5.1. *Inner solvers*

The computational efficiency consists in the way how the inner linear systems (9) are solved. Here, we omit the iterative index k for simplicity. Then, (9) reads as follows:

$$\begin{pmatrix} J_{11} & J_{12} & 0 \\ J_{21} & 0 & I \\ 0 & Z & N \end{pmatrix} \begin{pmatrix} \Delta x \\ \Delta \nu \\ \Delta z \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}, \tag{36}$$

where $r_1 \in \mathbb{R}^n$, $r_2, r_3 \in \mathbb{R}^{2m}$ stand for the components of the right-hand side vector in (9). We present two solution methods both based on the Schur complement reduction [3]. The first one uses the symmetric, positive definite *normal matrix* $J_{SC} = J_{SC}(v)$ given by

$$J_{SC} = J_{11} + J_{12} D_{22}^{-1} J_{21},$$

where $D_{22} = N^{-1}Z$. It arises from (36) when the unknowns $\Delta\nu$ and Δz are eliminated by

$$\begin{pmatrix} \Delta \nu \\ \Delta z \end{pmatrix} = \begin{pmatrix} -D_{22}^{-1} & Z^{-1} \\ I & 0 \end{pmatrix} \left(\begin{pmatrix} r_2 \\ r_3 \end{pmatrix} - \begin{pmatrix} J_{21} \\ 0 \end{pmatrix} \Delta x \right). \tag{37}$$

The reduced system for the unknown Δx reads as follows:

$$J_{SC} \Delta x = r_1 - J_{12} \left(Z^{-1} r_3 - D_{22}^{-1} r_2 \right). \tag{38}$$

The second solution method is based on the *augmented matrix* $J_{AM} = J_{AM}(v)$ given by

$$J_{AM} = \begin{pmatrix} J_{11} & J_{12} \\ J_{21} & -D_{22} \end{pmatrix}.$$

It arises from (36) by eliminating the unknown Δz using

$$\Delta z = N^{-1} r_3 - D_{22} \Delta \nu. \tag{39}$$

The reduced system in terms of Δx and $\Delta \nu$ reads as follows:

$$J_{AM} \begin{pmatrix} \Delta x \\ \Delta \nu \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 - N^{-1} r_3 \end{pmatrix}. \tag{40}$$

Note that J_{AM} is symmetric but indefinite and that J_{SC} is the Schur complement to the (2,2)-block in J_{AM}.

To get the solution to (36), we proceed in two steps: first we solve (38) or (40) after that we compute remaining components of the solution using (37) or (39), respectively. For solving (38) and (40), we apply the conjugate gradient method. It is known that this method requires a symmetric, positive definite matrix and that its convergence rate depends on its condition number [12]. In order to guarantee convergence for the augmented matrix, we will use the indefinite preconditioner

$$P_{AM} = \begin{pmatrix} D_{11} & J_{12} \\ J_{21} & -D_{22} \end{pmatrix},$$

where $D_{11} = diag(J_{11})$. The Schur complement P_{SC} to the (2,2)-block in P_{AM}, i.e.,

$$P_{SC} = D_{11} + J_{12}D_{22}^{-1}J_{21},$$

will be used as the preconditioner for J_{SC}. Actions of P_{AM}^{-1} and P_{SC}^{-1} on vectors are easily computable, since the preconditioners are defined by diagonal blocks.

THEOREM 5.1 *The eigenvalues λ of $P_{AM}^{-1}J_{AM}$ are positive. The eigenvalue $\lambda = 1$ is of the multiplicity $2m$ and the remaining n eigenvalues are the same as the eigenvalues of $P_{SC}^{-1}J_{SC}$.*

Proof The eigenvalues of $P_{AM}^{-1}J_{AM}$ can be found trough the generalized eigenvalue problem:

$$\begin{pmatrix} J_{11} & J_{12} \\ J_{21} & -D_{22} \end{pmatrix} \begin{pmatrix} x \\ \nu \end{pmatrix} = \lambda \begin{pmatrix} D_{11} & J_{12} \\ J_{21} & -D_{22} \end{pmatrix} \begin{pmatrix} x \\ \nu \end{pmatrix}. \tag{41}$$

The second block equation in (41) is satisfied, when either $\lambda = 1$ or $J_{21}x - D_{22}\nu = 0$. *(i)* If $\lambda = 1$, then the first block equation in (41) reduces to $J_{11}x = D_{11}x$ that holds for $x = 0$ so that $(0^\top, \nu^\top)^\top$ is the eigenvector for any $\nu \neq 0$. Let e_i denote the ith column of the identity matrix $I \in \mathbb{R}^{2m\times 2m}$. Then $(0^\top, e_i^\top)^\top$, $i \in \mathcal{M}_2$, are linearly independent eigenvectors associated with $\lambda = 1$. *(ii)* Substituting $\nu = D_{22}^{-1}J_{21}x$ in the first block equation in (41), we get

$$J_{SC}x = \lambda P_{SC}x \tag{42}$$

so that λ is the eigenvalue of $P_{SC}^{-1}J_{SC}$. Since P_{SC} is symmetric, positive definite as well as J_{SC}, the eigenvalue $\lambda = (x^\top J_{SC}x)/(x^\top P_{SC}x)$, $x \neq 0$, is positive. The theorem is proved. ■

COROLLARY 5.2 *It follows immediately from the last theorem that*

$$\kappa(P_{SC}^{-1}J_{SC}) \leq \kappa(P_{AM}^{-1}J_{AM}).$$

Note that $J_{SC}(v)$ and $J_{AM}(v)$ are typically ill-conditioned, when $v = v^{(k)}$ is sufficiently close to v^*. The reason is that the diagonal entry of D_{22} or D_{22}^{-1} tends to infinity, when the corresponding $\nu_i^* = 0$, $z_i^* \neq 0$ or $\nu_i^* \neq 0$, $z_i^* = 0$, respectively. Moreover, if $\nu_i^* = 0$, $z_i^* = 0$, then the matrices in (36) converge to the singular matrix $J(v^*)$ due to Remark 1. We prove that the condition numbers of the preconditioned matrices are bounded independently on $v = v^{(k)}$. To this end, we denote $B = J_{11} - A + J_{12}D_{22}^{-1}J_{21}$ and $D = diag(A)$. The generalized eigenvalue problem (42) reads as $(A + B)x = \lambda(D + B)x$ and, therefore, λ is given by the generalized Rayleigh quotient as

$$\lambda = \frac{x^\top Ax + x^\top Bx}{x^\top Dx + x^\top Bx}, \qquad \|x\| = 1. \tag{43}$$

To get bounds on λ, we will analyze the right-hand side of λ in (43). Let $a_{\min}$ and $a_{\max}$, $0 < a_{\min} \leq a_{\max}$, be the smallest and the largest eigenvalues of A, respectively, so that $a_{\min} \leq x^\top Ax \leq a_{\max}$. Further, let $d_{\min} = \min_i\{a_{ii}\}$ and $d_{\max} = \max_i\{a_{ii}\}$, $0 < d_{\min} \leq d_{\max}$, be the smallest and the largest eigenvalues of D, respectively, so that $d_{\min} \leq x^\top Dx \leq d_{\max}$. Here, a_{ii} denote the diagonal entries of A. Finally, note that B is symmetric, positive semidefinite so that $0 \leq x^\top Bx < +\infty$.

THEOREM 5.3 *All eigenvalues λ of the preconditioned normal matrix $P_{SC}^{-1}J_{SC}$ lay in the interval $[a_{\min}d_{\max}^{-1}, a_{\max}d_{\min}^{-1}]$.*

Proof Denote in (43) $a = x^\top Ax$, $b = x^\top Bx$, and $d = x^\top Dx$ so that $\lambda = \lambda(b\,;a,d)$, where

$$\lambda(b\,;a,d) = \frac{a+b}{d+b} = 1 + \frac{a-d}{d+b}.$$

Here, $\lambda(b\,;a,d)$ can be considered as the hyperbola for the variable $b \in \mathbb{R}_+$ given by the parameters a and d satisfying $a_{\min} \le a \le a_{\max}$ and $d_{\min} \le d \le d_{\max}$, respectively. If $ad^{-1} > 1$ or $ad^{-1} < 1$, then the hyperbola is decreasing or increasing on $\mathbb{R}_+$, respectively, and, if $ad^{-1} = 1$, then it reduces to the constant "1". There are two extremal hyperbolas $\lambda_{\min}(b) = \lambda(b\,;a_{\min}, d_{\max})$ and $\lambda_{\max}(b) = \lambda(b\,;a_{\max}, d_{\min})$, for which

$$\inf_{b\in\mathbb{R}_+} \lambda_{\min}(b) \le \lambda \le \sup_{b\in\mathbb{R}_+} \lambda_{\max}(b). \tag{44}$$

Since $a_{\min} \le d_{\min}$, we obtain $1 \ge a_{\min}d_{\min}^{-1} \ge a_{\min}d_{\max}^{-1}$. Therefore, $\lambda_{\min}(b)$ is increasing or constant on $\mathbb{R}_+$ so that

$$\inf_{b\in\mathbb{R}_+} \lambda_{\min}(b) = \lambda_{\min}(0) = a_{\min}d_{\max}^{-1}.$$

Since $a_{\max} \ge d_{\max}$, we get $1 \le a_{\max}d_{\max}^{-1} \le a_{\max}d_{\min}^{-1}$. Therefore, $\lambda_{\max}(b)$ is decreasing or constant on $\mathbb{R}_+$ so that

$$\sup_{b\in\mathbb{R}_+} \lambda_{\max}(b) = \lambda_{\max}(0) = a_{\max}d_{\min}^{-1}.$$

The theorem is proved. ∎

COROLLARY 5.4 *The theorem implies*

$$\kappa(P_{SC}^{-1}J_{SC}) \le \frac{a_{\max}d_{\max}}{a_{\min}d_{\min}} = \kappa(A)\kappa(D).$$

Using Theorem 5.1 and the fact that $a_{\min}d_{\max}^{-1} \le 1 \le a_{\max}d_{\min}^{-1}$, we get the same result also for the preconditioned augmented matrix, i.e.,

$$\kappa(P_{AM}^{-1}J_{AM}) \le \kappa(A)\kappa(D).$$

Note that $\kappa(A)\kappa(D) \le \kappa(A)^2$ gives simpler but usually more pessimistic bounds.

For an alternative analysis of preconditioners in the interior-point methods we refer to [4, 24].

5.2. *Implementation details*

In this section we discuss, among others, the choice of the parameters in ALGORITHM PF. Their optimal values was found by large numerical tests that are not included in the paper. For that, we used the benchmarks of Section 6 arising from finite element approximations and, as we observed comparable behaviors of the

computations for various meshes, it seems that the optimal values of the parameters do not depend on the size of the problem.

5.2.1. Adaptive inner precision control

To increase the efficiency, we initialize the conjugate gradient method in the kth iteration of ALGORITHM PF by the result obtained in the previous iteration and we stop it by the adaptive precision control $\epsilon_{cgm} = \epsilon_{cgm}^{(k)} \times \|r^{(k)}\|$, where $r^{(k)}$ stands for the right-hand side vector in (38) or (40). The value $\epsilon_{cgm}^{(k)}$ is proportional to the current outer precision $err^{(k-1)}$ defined in (11) or, if the progress is not sufficient, to the improved inner tolerance $\epsilon_{cgm}^{(k-1)}$ from the previous step:

$$\epsilon_{cgm}^{(k)} = \min\{r_{tol} \times err^{(k-1)}, c_{fact} \times \epsilon_{cgm}^{(k-1)}\}, \tag{45}$$

where $0 < r_{tol} < 1$, $0 < c_{fact} < 1$, $err^{(-1)} = 1$, and $\epsilon_{cgm}^{(-1)} = r_{tol}/c_{fact}$. Note that the computational experience showed that the initialization of the conjugate gradient method from the zero vectors is less beneficial.

5.2.2. Adaptive centering

The value $\sigma_k \in [\sigma_{\min}, \sigma_{\max}]$ is chosen by the following adaptive strategy:

$$\sigma_k = \min\left\{\sigma_{\max}, \max\left\{\sigma_{\min}, c_\sigma \times ((1-\xi_k)/\xi_k)^3\right\}\right\}, \tag{46}$$

where $\xi_k = \min_{i\in\mathcal{M}_2}\{\nu_i^{(k)} z_i^{(k)}\}/\vartheta_k$ and $c_\sigma > 0$. Notice that $\xi_k \in [\gamma, 1]$, since $\nu_i^{(k)} z_i^{(k)} \geq \gamma\vartheta_k$. The value $\xi_k = 1$ implies $\nu_i^{(k)} z_i^{(k)} = \vartheta_k$ for all $i \in \mathcal{M}_2$ indicating that the current iteration $v^{(k)}$ lies near to the central path. In this case, (46) gives $\sigma_k = \sigma_{\min}$ so that the computed direction is centered modestly. Oppositely, if $\xi_k = \gamma$, then $v^{(k)}$ lies on the boundary of $\mathcal{N}(\gamma, \beta)$ and $\sigma_k > \sigma_{\min}$ so that the centering is more distinctive. The critical value of ξ_k, for which (46) gives $\sigma_k = \sigma_{\max}$, is $\xi_k^{crit} = 1/(1 + \sqrt[3]{\sigma_{\max}/c_\sigma})$. In our computations, we use $\sigma_{\min} = 10^{-30}$, $\sigma_{\max} = 0.5$, and $c_\sigma = 1.25 \times 10^{-5}$ (resulting in $\xi_k^{crit} \doteq 0.0284$). See [25] for more details.

5.2.3. Steplength

The steplength $\alpha = \alpha_k$ in step (3°) of ALGORITHM PF must satisfy five inequalities (13)-(17) for $v = v^{(k)}$. To determine α_k, we use the backtracking procedure:

(a) Initialize $\rho \in (0,1)$, $\alpha_k^0 \in (0,1]$, and $j := 0$.
(b) While $ineq(\alpha_k^j)$ does not hold, set $\alpha_k^{j+1} = \rho\alpha_k^j$, $j := j+1$ and repeat.
(c) Return $\alpha_k = \alpha_k^j$.

The initial value α_k^0 is determined by

$$\alpha_k^0 = \min_{\Delta\nu_i^{(k+1)}, \Delta z_i^{(k+1)} < 0} \{1, -\delta\nu_i^{(k)}/\Delta\nu_i^{(k+1)}, -\delta z_i^{(k)}/\Delta z_i^{(k+1)}\} \tag{47}$$

with $\delta \in (0,1)$. It guarantees $\nu(\alpha_k^j) > 0$, $z(\alpha_k^j) > 0$ for $j \geq 0$ so that (13) holds. The satisfaction of (14)-(17) is achieved by inserting the respective inequality in the backtracking loop (b) as $ineq(\alpha_k^j)$. We consider different values of ρ: first we check (14) with ρ_1, then (15) with ρ_2, and finally (16) and (17) with ρ_3. The finite termination of the backtracking loop follows from Lemma 4.2. In the numerical experiments, we take $\delta = 0.999$, $\rho_1 = \rho_2 = 0.9$, and $\rho_3 = 0.5$.

5.2.4. Parameter ω

The parameter $\omega \in (0,1)$ in the Armijo-type condition (10) realizes certain compromise between the steplength α_k and the decrease in $\{\vartheta_k\}$. The value of ω close to one may result in short α_k while ω close to zero may lead to slow decrease between ϑ_k and ϑ_{k+1}. The numerical experiments showed that $\omega = 0.1$ gives the satisfactory behavior of ALGORITHM PF for our problems.

5.2.5. Initial iteration

We describe how to ensure that the initial iteration $v^{(0)} = ({x^{(0)}}^\top, {\nu^{(0)}}^\top, {z^{(0)}}^\top)^\top$ belongs to $\mathcal{N}(\gamma,\beta)$. We choose $x^{(0)} \in \mathbb{R}^n$ arbitrarily, e.g., $x^{(0)} = 0$. The vectors $\nu^{(0)} \in \mathbb{R}^{2m}$ and $z^{(0)} \in \mathbb{R}^{2m}$ are defined by $\nu^{(0)} = c_\nu e$ and $z^{(0)} = c_z e$, where c_ν and c_z are given positive constants, respectively. It is easily seen that $\nu_i^{(0)} z_i^{(0)} = c_\nu c_z \geq \gamma c_\nu c_z = \gamma\vartheta_0$ for all $i \in \mathcal{M}_2$ and for any $\gamma \in (0,1]$. Therefore, no restriction on γ is necessary. The value of β is adapted with respect to the initial iteration. First we compute

$$\bar{\beta} = \beta_0 \times \max\{\|\nabla_x L(x^{(0)},\nu^{(0)})\|/\vartheta_0, \|\nabla_\nu L(x^{(0)},\nu^{(0)}) + z^{(0)}\|/\vartheta_0\}$$

for given $\beta_0 \geq 1$. As Theorem 4.3 requires $\beta \geq 1$, we take $\beta = \max\{1,\bar{\beta}\}$. Note that large values of β lead to saving in computations, since the backtracking steps for the time consuming conditions (16) and (17) may be avoided. We use $c_\nu = c_z = 1$, $\gamma = 0.001$, and $\beta_0 = 10^9$.

6. Numerical experiments

Numerical experiments will illustrate behavior of ALGORITHM PF for two model problems. The first one was used in [21] for testing the active set KPRGP algorithm. Its difficultness consists in the fact that the solution is almost degenerate. The second one is the contact problem of linear elasticity with Tresca friction whose finite element approximation leads to (1) [15, 17]. Moreover, we extend ALGORITHM PF for solving contact problems with Coulomb friction. All computations are performed in MATLAB on PC Core i7(2.8 GHz) with 4GB RAM.

In tables below we report the number *iter* of outer iterations, the number n_A of matrix-vector multiplications by A, and the solution time in seconds. As the multiplication by A is usually the most expensive operation, the value n_A determines the total complexity of computations. In labeling of columns PF(r_{tol}, c_{fact}), we introduce the values of r_{tol} and c_{fact}. In order to compare the efficiency with KPRGP, we report its solution characteristics in respective columns.

6.1. *Chord problem*

Let us consider the following problem:

$$\min \frac{1}{2}\int_0^1 \|u'(t)\|^2\,dt - \int_0^1 u(t)^\top f(t)\,dt \qquad \text{s.t. } u = (u_1,u_2)^\top \in \mathcal{K},$$

where $\mathcal{K} = \{u \in (H_0^1(0,1))^2 : \ u_2(t) \geq 0 \text{ on } (0,0.5),\ \|u(t)\| \leq 1.4 \text{ on } (0.5,1)\}$ and $f(t) = (36\pi^2 \sin 6\pi t, -4\pi^2 \sin 2\pi t)^\top$. It describes the loaded chord fixed at the endpoints that is partially above the plan, partially inside the cylindrical tube; see Figure 1. The finite element approximation based on the continuous piecewise linear functions over a regular grid with $n = 4m$ degrees of freedom leads to

the constrained minimization problem that, contrary to (1), contains the fourth unconstrained component $x_4 \in \mathbb{R}^m$. Obviously, ALGORITHM PF may be easily adapted for this new problem so that Theorem 4.3 remains valid.

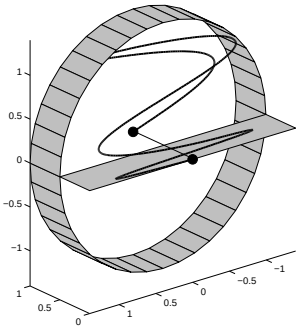

Figure 1. Deformation of the chord.

Tables 1 and 2 show how ALGORITHM PF behaves for the inner solvers based on J_{SC} and J_{AM}, respectively. One can see that J_{AM} leads usually to a higher performance of the algorithm in terms of n_A. However, the progress measured by the computational time is not so big, since the actions of P_{AM}^{-1} are costlier than of P_{SC}^{-1}. In order to obtain the same final precision of the solutions computed from ALGORITHM PF and KPRGP, we adopt in ALGORITHM PF the stopping criterion from KPRGP (instead of (11)), i.e., the iterations are stopped, when the relative norm of the reduced gradient [10, 21] for the projection of $x^{(k)}$ on Ω is less or equal to 10^{-4}. The efficiency of ALGORITHM PF exceeds KPRGP, if n is large. The automatic choice $r_{tol} = \kappa(A)^{-1/2}$ used in the first columns of the tables is motivated by practical experience. When we replace the adaptive inner precision control (45) by the non-adaptive one $\epsilon_{cgm}^{(k)} = r_{tol} \times \epsilon$, then ALGORITHM PF generates the same number of outer iterations *iter* for J_{SC} and J_{AM} (with the same r_{tol}).

The iteration history of the centering parameter σ_k and the steplength α_k are depicted in Figures 2 and 3 for $n = 1024$. One can see alternations of modestly and strongly centered iterations so that the longer steps are generated after the stronger centering. Figures 4 and 5 show the preconditioning effect. Although the condition numbers $\kappa(J_{SC})$ and $\kappa(J_{AM})$ increase without any stint, the condition numbers of the preconditioned matrices $\kappa(P_{SC}^{-1}J_{SC})$ and $\kappa(P_{AM}^{-1}J_{AM})$ are bounded by the value $\kappa(A) \doteq 1.07 \times 10^5$, as it is predicted by Corollary 5.4 (since $\kappa(D) = 1$).

Table 1. Chord problem, inner solver based on J_{SC}.

	PF$(\kappa(A)^{-1/2}, 0.99)$		PF$(0.001, 0.99)$		PF$(0.01, 0.9)$		KPRGP	
n	$iter/n_A$	*time*	$iter/n_A$	*time*	$iter/n_A$	*time*	n_A	*time*
64	13/357	0.05	13/564	0.08	14/474	0.06	163	0.03
128	17/913	0.14	16/1247	0.20	16/941	0.14	522	0.12
256	19/2179	0.39	19/2869	0.53	19/2209	0.39	1209	0.33
512	24/5876	1.42	22/6434	1.61	24/5525	1.31	3163	1.08
1024	27/13721	4.80	28/16214	5.87	27/12396	4.32	8983	4.37
2048	31/33238	19.30	31/35187	20.34	30/28949	16.68	26061	18.61
4096	35/80841	83.30	35/77892	80.73	32/58091	60.08	91439	107.20
8192	33/160084	311.05	35/154755	300.02	35/130412	254.86	351528	744.19

Table 2. Chord problem, inner solver based on J_{AM}.

	PF$(\kappa(A)^{-1/2}, 0.99)$		PF$(0.001, 0.99)$		PF$(0.01, 0.9)$		KPRGP	
n	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$	n_A	$time$
64	16/292	0.06	13/415	0.08	15/340	0.06	163	0.03
128	16/472	0.11	15/807	0.19	17/632	0.16	522	0.12
256	24/1222	0.37	20/1617	0.50	22/1129	0.34	1209	0.33
512	26/2625	1.17	24/3297	1.45	27/2280	1.00	3163	1.08
1024	29/5867	4.10	27/6509	4.59	34/4416	3.07	8983	4.37
2048	35/10965	14.01	35/13850	17.78	42/8961	11.23	26061	18.61
4096	39/21778	53.73	44/23828	58.44	54/18875	46.02	91439	107.20
8192	51/47542	218.59	62/41673	191.82	69/51497	236.92	351528	744.19

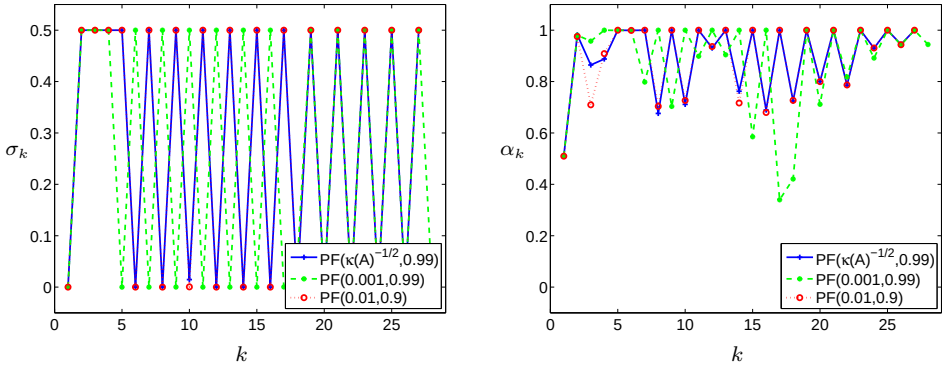

Figure 2. Chord problem, σ_k and α_k for J_{SC}.

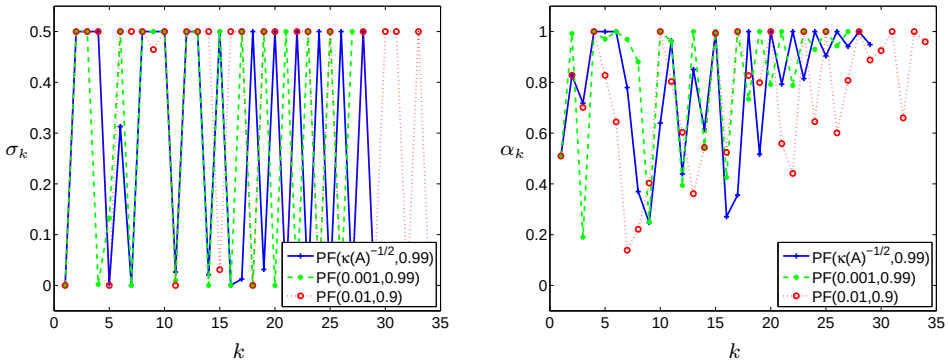

Figure 3. Chord problem, σ_k and α_k for J_{AM}.

6.2. *Tresca friction*

Let us consider a steel brick $\mathcal{S}$ in $\mathbb{R}^3$ lying on a rigid obstacle. The brick occupies the domain $\mathcal{S} = (0,3) \times (0,1) \times (0,1)$ whose boundary $\partial\mathcal{S}$ split into three parts $\Gamma_u = \{0\} \times (0,1) \times (0,1)$, $\Gamma_c = (0,3) \times (0,1) \times \{0\}$, and $\Gamma_p = \partial\mathcal{S} \setminus (\bar{\Gamma}_u \cup \bar{\Gamma}_c)$ with different boundary conditions; see Figure 6. The zero displacements are prescribed on Γ_u while the surface tractions act on Γ_p. On Γ_c we consider the contact conditions, i.e., the non-penetration and the effect of Tresca friction. The elastic behavior of the brick is described by the Lamé equations that lead after finite element approximation to the symmetric, positive definite stiffness matrix $K \in \mathbb{R}^{3n_n \times 3n_n}$ and to a load vector $f \in \mathbb{R}^{3n_n}$, where n_n is the number of finite element nodes in $\bar{\mathcal{S}} \setminus \bar{\Gamma}_u$. By m we denote the number of contact nodes, i.e., the nodes lying on $\bar{\Gamma}_c \setminus \bar{\Gamma}_u$. To describe the contact conditions, we introduce the full-rank matrices $N, T_1, T_2 \in \mathbb{R}^{m \times 3n_n}$ projecting displacements in the contact nodes to the normal and tangential directions, respectively. Finally, $d \in \mathbb{R}^m_+$ and $g \in \mathbb{R}^m_+$ collect distances to the obstacle and à-priori given slip bound values at the contact nodes, respectively.

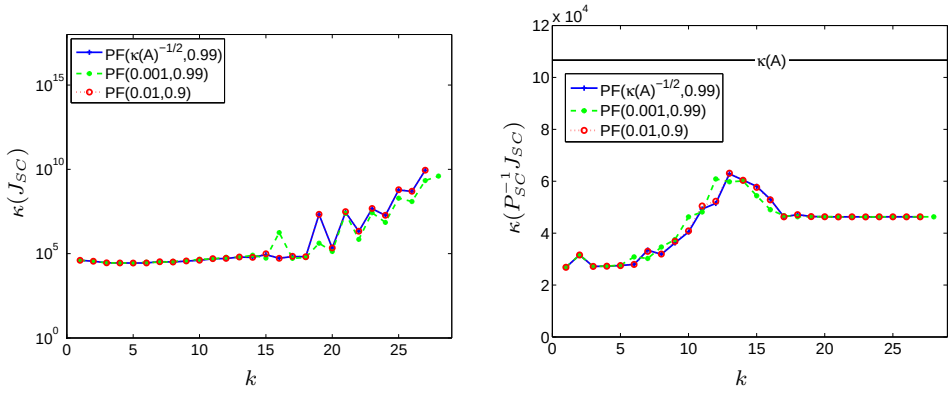

Figure 4. Chord problem, the preconditioning effect for J_{SC}.

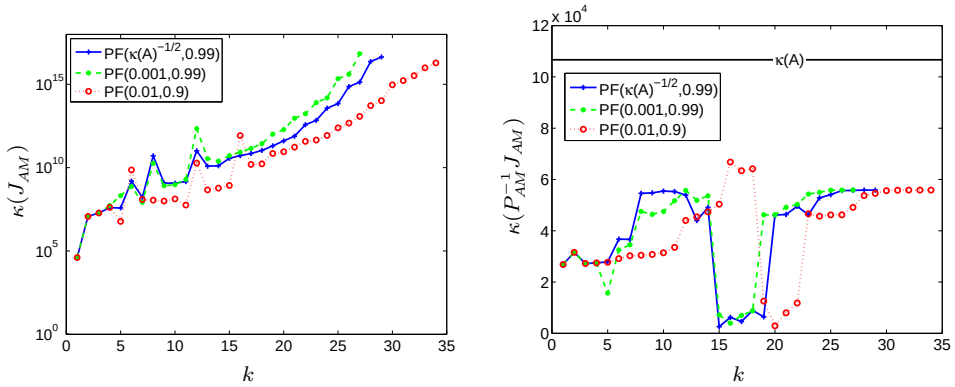

Figure 5. Chord problem, the preconditioning effect for J_{AM}.

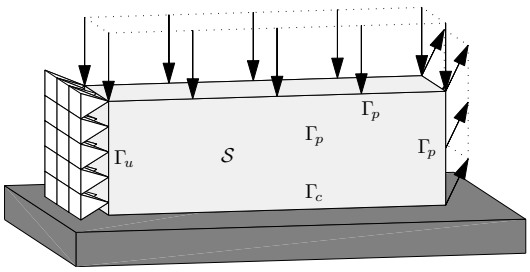

Figure 6. Steel brick.

The discrete contact problem with Tresca friction consists in seeking $(u^*, \lambda_\nu^*, \lambda_{\tau_1}^*, \lambda_{\tau_2}^*) \in \mathbb{R}^{n_n} \times \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^m$ satisfying the following conditions [17]:

$$Ku - f + N^\top \lambda_\nu + T_1^\top \lambda_{\tau_1} + T_2^\top \lambda_{\tau_2} = 0, \tag{48}$$

$$Nu - d \leq 0,\ \lambda_\nu \geq 0,\ \lambda_\nu^\top (Nu - d) = 0, \tag{49}$$

$$\left.\begin{array}{l} \|\lambda_{\tau,i}\| \leq g_i, \\ \|\lambda_{\tau,i}\| < g_i \Rightarrow u_{\tau,i} = 0, \\ \|\lambda_{\tau,i}\| = g_i \Rightarrow \exists c_i \geq 0 :\ u_{\tau,i} = c_i \lambda_{\tau,i}, \end{array}\right\} \quad i \in \mathcal{M}_1, \tag{50}$$

where $\lambda_{\tau,i} = (\lambda_{\tau_1,i}, \lambda_{\tau_2,i})^\top \in \mathbb{R}^2$, $u_{\tau,i} = ((T_1 u)_i, (T_2 u)_i)^\top \in \mathbb{R}^2$, and g_i are the

entries of g. Let us denote

$$\lambda = \left(\lambda_\nu^\top, \lambda_{\tau_1}^\top, \lambda_{\tau_2}^\top\right)^\top, \quad B = \left(N^\top, T_1^\top, T_2^\top\right)^\top, \quad h = \left(d^\top, 0^\top, 0^\top\right)^\top.$$

Since (48) gives $u = K^{-1}(f - B^\top\lambda)$, we can eliminate u from (48)-(50) that results in problem (1) for $x = \lambda$ with

$$A = BK^{-1}B^\top, \quad b = BK^{-1}f - h, \quad l_i = 0, \quad i \in \mathcal{M}_1.$$

Recall that the preconditioners in ALGORITHM PF require the knowledge of the diagonal of A. As its assemblage is now expensive, we use the following approximation:

$$diag(A) \approx diag(B\, diag(K)^{-1}B^\top).$$

In Tables 3, 4 and in Figures 7, 8, 9, 10 for $n = 18252$, we present analogous tests as for the chord problem. We use in ALGORITHM PF the stopping criterion (11) with $\epsilon = 10^{-2}$. The comparability of results computed by ALGORITHM PF and KPRGP is checked by the accuracy in satisfying (48), (49), and (50). Note that our choice of ϵ is adequate to the precision control 10^{-4} in the stopping criterion of KPRGP that is sufficient for many engineering problems. One can see that only few strongly centered iterations are generated. The comparisons with KPRGP show higher efficiency of ALGORITHM PF. In the agreement with Corollary 5.4, the condition numbers of the preconditioned matrices are bounded by $\kappa(A)\kappa(D)$, where $\kappa(A) \doteq 6.7 \times 10^3$ and $\kappa(D) \doteq 4$.

Table 3. Tresca friction, inner solver based on J_{SC}.

	PF(0.3, 0.99)		PF(0.1, 0.9)		PF(0.01, 0.99)		KPRGP	
n/m	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$	n_A	$time$
900/180	15/87	0.28	14/92	0.28	14/119	0.36	187	0.48
2646/378	13/85	1.61	13/96	1.78	12/105	1.89	181	2.70
5832/648	15/91	6.43	14/104	7.11	15/130	8.47	187	10.14
10890/990	17/120	23.20	15/116	22.17	17/166	29.72	203	30.64
18252/1404	16/106	48.22	15/116	51.51	15/147	62.04	230	78.13
28350/1890	17/121	111.18	16/125	113.40	16/151	131.90	254	177.31
41616/2448	18/130	226.19	17/130	224.11	17/175	285.51	259	344.97

Table 4. Tresca friction, inner solver based on J_{AM}.

	PF(0.3, 0.99)		PF(0.1, 0.9)		PF(0.01, 0.99)		KPRGP	
n/m	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$	n_A	$time$
900/180	24/155	0.47	18/110	0.34	14/91	0.28	187	0.48
2646/378	17/116	2.12	17/118	2.17	13/106	2.01	181	2.70
5832/648	15/88	6.30	16/108	7.47	15/125	8.24	187	10.14
10890/990	19/132	24.91	17/134	24.98	16/136	25.24	203	30.64
18252/1404	16/95	44.43	13/75	36.49	16/141	60.47	230	78.13
28350/1890	19/128	117.44	16/120	109.72	16/133	118.97	254	177.31
41616/2448	16/102	184.47	18/135	232.27	17/157	260.74	259	344.97

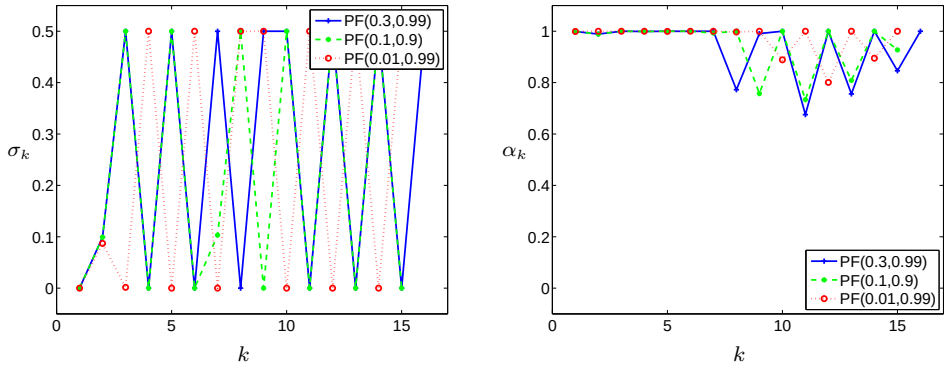

Figure 7. Tresca friction, σ_k and α_k for J_{SC}.

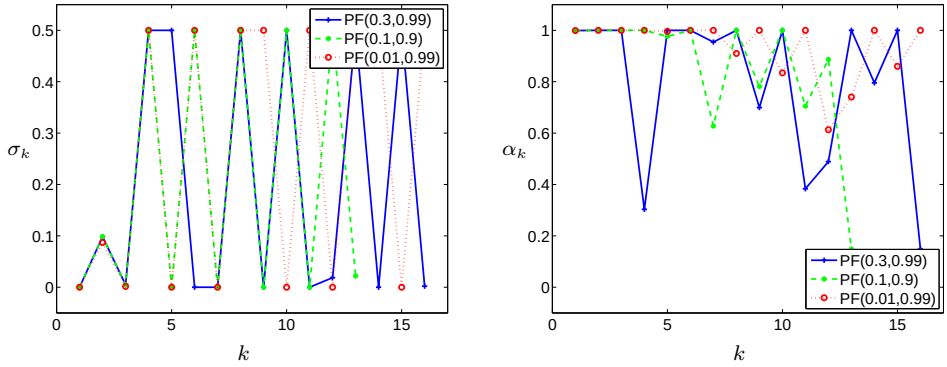

Figure 8. Tresca friction, σ_k and α_k for J_{AM}.

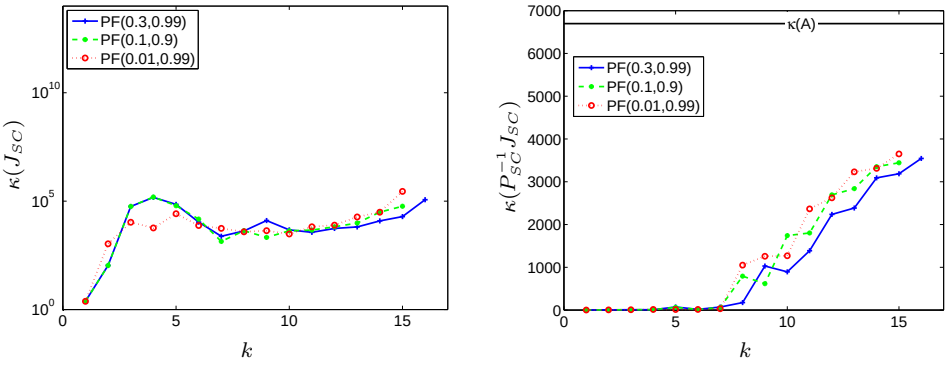

Figure 9. Tresca friction, the preconditioning effect for J_{SC}.

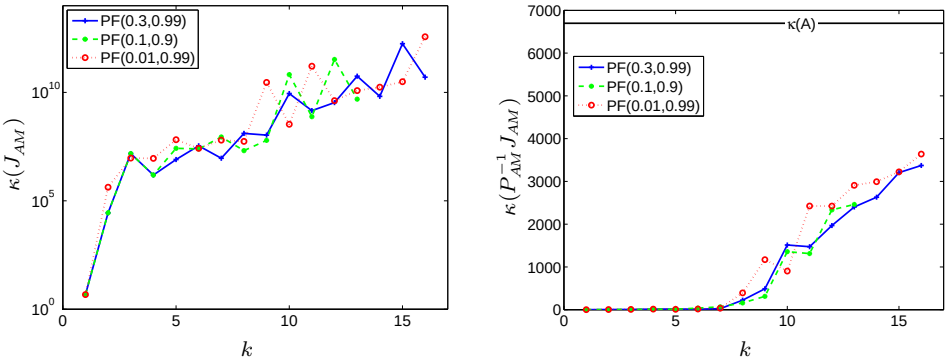

Figure 10. Tresca friction, the preconditioning effect for J_{AM}.

6.3. *Coulomb friction*

Let us replace (50) by:

$$\left.\begin{aligned} &\|\lambda_{\tau,i}\| \le \mathcal{F}\lambda_{\nu,i}, \\ &\|\lambda_{\tau,i}\| < \mathcal{F}\lambda_{\nu,i} \Rightarrow u_{\tau,i} = 0, \\ &\|\lambda_{\tau,i}\| = \mathcal{F}\lambda_{\nu,i} \Rightarrow \exists c_i \ge 0 : \ u_{\tau,i} = c_i\lambda_{\tau,i}, \end{aligned}\right\} \quad i \in \mathcal{M}_1, \tag{51}$$

where $\mathcal{F} \in \mathbb{R}_+$ is the coefficient of Coulomb friction. The discrete contact problem with Coulomb friction consists in seeking $(u^*, \lambda_\nu^*, \lambda_{\tau_1}^*, \lambda_{\tau_2}^*) \in \mathbb{R}^{n_n} \times \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^m$ satisfying (48), (49), and (51). Although there is no (evident) equivalency with any minimization problem, we will solve this problem by a simple modification of Algorithm PF.

Let us define the mapping $\Psi : \mathbb{R}_+^m \mapsto \mathbb{R}_+^m$, $\Psi(g) = \mathcal{F}\lambda_\nu$, $g \in \mathbb{R}_+^m$, where $\lambda_\nu = \lambda_\nu(g)$ is the second component of the solution to the contact problem with Tresca friction. It is easily seen that a fixed point of Ψ, i.e., the point $g^* \in \mathbb{R}_+^m$ satisfying $\Psi(g^*) = g^*$, determines a solution to the contact problem with Coulomb friction. The natural way how to compute fixed points is the method of successive approximations:

$$\textit{Initialize: } g^{(0)}, \quad g^{(p)} = \Psi(g^{(p-1)}), \quad p = 1, 2, \dots . \tag{52}$$

These iterations converge when Ψ is contractive that is guaranteed by sufficiently small $\mathcal{F}$ [14]. Note that each evaluation of the mapping Ψ in (52) requires to solve the contact problem with Tresca friction for that Algorithm PF may be used. In order to achieve high computational efficiency, we prefer an inexact implementation of (52), in which one iteration of Algorithm PF is performed in each successive approximation. As the first m components of $x^{(k)}$ approximates λ_ν, we can equivalently perform this idea by modifying Algorithm PF so that g is updated in the beginning of each iteration, i.e., in step (1°), by

$$g := g^{(k)} = \mathcal{F}((x_1^{(k)})^+, \dots, (x_m^{(k)})^+)^\top,$$

where $y^+ = \max\{0, y\}$ is the non-negative part of $y \in \mathbb{R}$. Since the component $z^{(k)}$ of $v^{(k)}$ depends also on g, we modify appropriately its value ($z_{i+m}^{(k)} := z_{i+m}^{(k)} - g_i^{(k-1)} + g_i^{(k)}$, $1 \le i \le m$, $k > 0$). We refer to this algorithm as Algorithm PFC. The convergence analysis of Section 4 is not valid, but it is obvious that, *if the modified algorithm converges, then the limit point is a solution to the contact problem with Coulomb friction* [1].

We compare the efficiency of Algorithm PFC with the method of successive approximations (52), in which KPRGP is used for solving inner problems with Tresca friction. We refer to this algorithm as KPRGPC. As KPRGP in KPRGPC is the inner iterative method, we apply the adaptive inner precision control analogous to this one described in Section 5.1. Based on our experiences with algorithms of this type [16, 23], we chose optimal values of the parameters defining the inner precisions control so that also solution characteristics introduced below for KPRGPC are near to optimal. The value *iter* for KPRGPC denotes the number of successive iterations. In Algorithm PFC we use $\epsilon = 10^{-4}$. The comparability of results computed by Algorithm PFC and KPRGPC is checked by the accuracy in satisfying (48), (49), and (51).

In Tables 5, 6, 7, and 8, we test Algorithm PFC for different coefficients of friction $\mathcal{F} = 0.1$ and 0.4. Note that Algorithm PFC behaves reasonably in all

cases. One can see that the computations based on J_{SC} are more efficient than these ones based on J_{AM}. Moreover, the comparison with KPRGPC shows better performance of ALGORITHM PFC in the most cases.

Table 5. Coulomb friction, $\mathcal{F} = 0.1$, inner solver based on J_{SC}, $\delta = 0.9$.

	PFC(0.3, 0.99)		PFC(0.1, 0.9)		PFC(0.01, 0.99)		KPRGPC	
n/m	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$
900/180	32/206	0.62	31/214	0.66	30/251	0.72	15/483	1.06
2646/378	31/217	4.06	29/223	4.09	32/293	5.12	17/735	10.08
5832/648	38/330	21.62	42/456	28.47	50/746	44.15	14/434	22.23
10890/990	38/290	55.75	37/346	63.66	37/415	73.85	17/635	92.54
18252/1404	40/348	151.09	36/321	139.14	37/420	173.46	21/781	265.50
28350/1890	44/350	319.30	38/323	291.46	44/488	415.62	15/730	513.26
41616/2448	44/364	625.67	35/354	588.61	45/497	806.32	18/684	911.25

Table 6. Coulomb friction, $\mathcal{F} = 0.1$, inner solver based on J_{AM}, $\delta = 0.9$.

	PFC(0.3, 0.99)		PFC(0.1, 0.9)		PFC(0.01, 0.99)		KPRGPC	
n/m	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$
900/180	30/236	0.70	30/256	0.73	29/292	0.83	15/483	1.06
2646/378	31/245	4.45	30/283	4.96	30/364	6.07	17/735	10.08
5832/648	37/426	26.43	41/582	34.84	37/589	34.91	14/434	22.23
10890/990	33/403	70.87	37/407	72.62	35/554	93.38	17/635	92.54
18252/1404	33/393	161.68	36/472	190.66	36/547	216.31	21/781	265.50
28350/1890	44/448	387.87	41/585	479.52	43/773	614.38	15/730	513.26
41616/2448	44/417	696.81	43/459	751.33	44/674	1041.88	18/684	911.25

Table 7. Coulomb friction, $\mathcal{F} = 0.4$, inner solver based on J_{SC}, $\delta = 0.9$.

	PFC(0.3, 0.99)		PFC(0.1, 0.9)		PFC(0.01, 0.99)		KPRGPC	
n/m	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$
900/180	32/206	0.64	31/214	0.66	30/251	0.73	15/483	1.06
2646/378	31/217	4.07	29/223	4.10	32/293	5.24	17/735	10.12
5832/648	38/330	21.75	42/456	28.58	50/746	44.59	14/434	22.59
10890/990	38/290	56.22	37/346	64.01	37/415	74.13	17/635	93.18
18252/1404	40/348	152.30	36/321	140.46	37/420	174.89	21/781	267.48
28350/1890	44/350	321.44	38/323	293.80	44/488	418.57	15/730	516.36
41616/2448	44/364	630.07	35/354	592.82	45/497	810.63	18/684	918.69

Table 8. Coulomb friction, $\mathcal{F} = 0.4$, inner solver based on J_{AM}, $\delta = 0.9$.

	PFC(0.3, 0.99)		PFC(0.1, 0.9)		PFC(0.01, 0.99)		KPRGPC	
n/m	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$	$iter/n_A$	$time$
900/180	30/236	0.70	30/256	0.75	29/292	0.83	15/483	1.06
2646/378	31/245	4.45	30/283	4.95	30/364	6.07	17/735	10.12
5832/648	37/426	26.58	41/582	35.19	37/589	34.99	14/434	22.59
10890/990	33/403	71.31	37/407	72.99	35/554	94.22	17/635	93.18
18252/1404	33/393	163.44	36/472	192.21	36/547	217.79	21/781	267.48
28350/1890	44/448	390.95	41/585	483.29	43/773	619.60	15/730	516.36
41616/2448	44/417	703.17	43/459	755.78	44/674	1049.09	18/684	918.69

7. Conclusions and comments

We have presented the new interior-point algorithm for the minimization of strictly quadratic functions subject to the simple bounds and separable quadratic inequality constraints. In the theoretical part of the paper, we proved that the algorithm

converges to the solution, when it generates a bounded sequence of iterations. Another theoretical result consists in the spectral analysis of the preconditioned matrices of the inner linear systems. It is shown that the eigenvalues lay in the positive interval whose bounds are determined solely by the matrix A. We arrived at the result, in a sense optimal, that the condition numbers of the preconditioned matrices are uniformly bounded independently of the current interior-point iteration.

The second part of the paper is devoted to numerical tests. First, we found experimentally optimal values of the parameters determined the high computational performance of the algorithm for solving large scale problems. Then, we compared the efficiency with the active set algorithm proposed by the first author [10, 20, 21] in solving 3D contact problems of linear elasticity with friction. The efficiency of the interior-point algorithm is higher in almost all numerical experiments. Its lack consists in weaker theoretical results that do not comprise the prove of the convergence rate. Nevertheless, it seems that the algorithm is the promising tool for solving realistic problems of the real world.

The final comment is the qualitative comparison with other methods for solving contact problems of linear elasticity. One can divide them on the primal approaches, e.g., the non-smooth multiscale method of Krause [19], the primal-dual approaches, e.g., the semi-smooth Newton method of Hueber et al. [18], and the dual approaches, e.g., the FETI (finite element tearing and interconnecting) based domain decomposition method of Dostál et al. [7, 22]. Such classification is rough, since each of this approaches depends strongly on the algorithm for solving inner problems. Our interior-point algorithm may be conceived as the inner solver for the dual approach. The preconditioners discussed in the paper are related to the algorithm structure so that they remove singularities inherently given by the realization of the interior-point idea. In other words, they are the *inner* preconditioners of the interior-point algorithm. The *outer* preconditioner for the dual approach is the usage of the FETI method itself. Based on our spectral analysis, we deduce that the inner preconditioners presented in the paper are well situated to profit also from the FETI method. This topic together with other techniques exploiting the PDE structure are postponed to future works.

Acknowledgements

This work was supported by the European Development Fund in the IT4Innovations Centre of Excellence project CZ.1.05/1.1.00/02.0070 (RK), by the grant 101/08/0574 (RK) of the Grant Agency of the Czech Republic, by the Research Projects MSM6198910027 (RK) of the Czech Ministry of Education and by the IGA UPOL grant PrF-2012-017 (JM,HN,PŽ).

References

[1] P. Alfeld: Fixed point iteration with inexact function value. Mathematics of Computations, **38** (1982), pp. 87–98.
[2] A. Ben-Tal, A. Nemirovski: *Lectures on modern convex optimization*. SIAM, Philadelphia (2001)
[3] M. Benzi, G. H. Golub, J. Liesen: Numerical solution of saddle point systems. Acta Numerica, **14** (2005), pp. 1–137.
[4] L. Bergamaschi, J. Gondzio, G. Zilli: Preconditioning indefinite systems in interior point methods for optimization. Computational Optimization and Applications, **28** (2004), pp. 149–171.
[5] S. Boyd, L. Vandenberghe: *Convex optimization*. Cambridge University Press, Cambridge 2004.
[6] A. Conn, N. Gould, P. Toint: *Trust region methods*. SIAM, Philadelphia 2000.
[7] Z. Dostál, T. Kozubek, A. Markopoulos, T. Brzobohatý, V. Vondrák, P. Horyl: Theoretically sup-

ported scalable TFETI algorithm for the solution of multibody 3D contact problems with friction. Accepted in Comput. Methods Appl. Mech. Engrg., (2011).

[8] Z. Dostál, T. Kozubek, V. Vondrák, T. Brzobohatý, A. Markopoulos: Scalable TFETI algorithm for the solution of coercive multibody contact problems of elasticity. Internat. J. Numer. Methods Engrg., **82** (2010), pp. 1384–1405.

[9] Z. Dostál: *Optimal quadratic programming algorithms: with applications to variational inequalities.* SOIA 23, Springer US, New York 2009.

[10] Z. Dostál, R. Kučera: An optimal algorithm for minimization of quadratic functions with bounded spectrum subject to separable convex inequality and linear equaity constraints. SIAM J. Optim., **20** (2010), pp. 2913–2938.

[11] Z. Dostál, J. Schöberl: Minimizing quadratic functions over non-negative cone with the rate of convergence and finite termination. Comput. Optim. Appl., **30** (2005), pp. 23–44.

[12] G. Golub, C. F. Van Loan: *Matrix computations.* MD: The Johns Hopkins University Press, Baltimore 1996.

[13] C. C. Gonzaga: An algorithm for solving linear programming problems in $O(n^3L)$ operations. In: Progress in Mathematical Programming: Interior-Point and Related Methods, editor N. Megiddo, Springer-Verlag, New York 1989, pp. 1–28.

[14] J. Haslinger. Approximation of the Signorini problem with friction, obeying the Coulomb law. Math. Methods Appl. Sci., **5** (1983), pp. 422–437.

[15] J. Haslinger, R. Kučera, Z. Dostál: An algorithm for the numerical realization of 3D contact problems with Coulomb friction. J. Comput. Appl. Math., **164-165** (2004), pp. 387–408.

[16] J. Haslinger, R. Kučera, T. Ligurský: Qualitative analysis of 3D elastostatic contact problems with orthotropic Coulomb friction and solution-dependent coefficients of friction. J. Comput. Appl. Math., **235** (2011), pp. 3464–3480.

[17] J. Haslinger, T. Kozubek, R. Kučera: Numerical solution of contact problems with orthotropic Coulomb friction based on quadratic programming approach with the elliptic friction cone. Submitted (2011).

[18] S. Hueber, G. Stadler, B. I. Wohlmuth: A primal-dual active set algorithm for three-dimensional contact problems with Coulomb friction. SIAM J. Sci. Comput., **30** (2008), pp. 527–596.

[19] R. Krause: A non-smooth multiscale method for solving frictional two-body contact problems in 2D and 3D with multigrid efficiency. SIAM J. Sci. Comput., **31** (2009), pp. 1399–1423.

[20] R. Kučera: Minimizing quadratic functions with separable quadratic constraints. Optimization Methods and Software, **22** (2007), pp. 453–467.

[21] R. Kučera: Convergence rate of an optimization algorithm for minimizing quadratic functions with separable convex constraints. SIAM J. Optim., **19** (2008), pp. 846–862.

[22] T. Kozubek, A. Markopoulos, T. Brzobohatý, R. Kučera, V. Vondrák, Z. Dostál: MatSol - MATLAB efficient solvers for problems in engineering. *http://matsol.vsb.cz*

[23] T. Ligurský, J. Haslinger, R. Kučera: Approximation and numerical realization of 3D contact problems with Coulomb friction and a solution-dependent coefficient of friction. International Journal for Numerical Methods in Engineering, **82** (2010), pp. 1180–1206.

[24] L. Lukšan, C. Matonoha, J. Vlček: Interior point method for large-scale nonlinear programming. Optimization Methods and Software, **20** (2005), pp. 569–582.

[25] J. Nocedal, A. Wächter, R. A. Waltz: Adaptive barrier strategies for nonlinear interior methods. TR RC 23563, IBM T.J. Watson Research Center, 2005.

[26] D. Ralph, S. J. Wright: Superlinear convergence of an interior-point method for monotone variational inequalities. In: Complementarity and Variational Problems: State of the Art, editors: M. C. Ferris and J.-S. Pang, SIAM Publications, Philadelphia 1997, pp. 345–385.

[27] J. Renegar: A polynomial time algorithm based on Newton's method for linear programming. Mathematical Programming **40** (1988), pp. 59–93.

[28] J. B. Rosen: The gradient projection method for nonlinear programming. Part II. Nonlinear Constraints. J. Soc. Indust. Appl. Math., **9** (1961), pp. 515–532.

[29] S. J. Wright: Primal-Dual Interior-Point Methods. SIAM, Philadelphia 1997.

[30] S. J. Wright, D. Ralph: A Superlinear infeasible-interior-point algorithm for monotone complementarity problems. Mathematics of Operations Research **21** (1996), pp. 815–838.