Solution schemes for variational inequalities with generalized box constraints.

Lothar Banz[a], Gregor Milicic[a], Andreas Schröder[a]

[a]*Department of Mathematics, University of Salzburg, Hellbrunner Straße 34, 5020 Salzburg, Austria*

Abstract

For a class of linear variational inequalities such as Signorini contact problems a basis transformation on the algebraic level can be used to convert the original linear inequality constraints into simple box constraints. This enables the use of many efficient and stable iterative solvers. In this paper we show how this basis transformation can be utilized within the primal-dual active set method as well within an accelerated projected SOR scheme even without the explicit computation of the transformation. For the latter we proof its global convergence. Moreover, we discuss some implementational aspects and show that the accelerated projected SOR schemes are compatible to the primal-dual active set method. For this purpose, we study the algorithms with respect to their convergence behavior and the effect of the basis transformation in some numerical experiments.

Keywords: variational inequalities, Signorini contact problems, iterative solvers

1. Introduction

Linear variational inequalities of the type: Find $x \in K$ such that

$$(y - x)^\top Ax \geq (y - x)^\top L \tag{1}$$

for all $y \in K$ with given $A \in \mathbb{R}^{n\times n}$, $L \in \mathbb{R}^n$, $B \in \mathbb{R}^{m\times n}$ and $g \in \mathbb{R}^m$ and a convex set

$$K := \{y \in \mathbb{R}^n \mid By \leq g\}, \tag{2}$$

appear in many problems of mathematical programming. For instance, they result from the discretization of contact problems and have a vast amount of related applications [12, 8, 7]. Here, the relation $u \leq v$ with $u, v \in \mathbb{R}^m$ is defined componentwise, i.e. $u \leq v$ if and only if $u_i \leq v_i$ for all $1 \leq i \leq m$. Typically, the matrix A is symmetric positive definite if finite elements or boundary elements are used for the discretization. However, the application of more advanced discretization methods like interior penalty discontinuous Galerkin methods leads to an unsymmetric and almost unconditionally positive definite matrix A [1, 2] so that the weak assumption on A being a positive definite, but not necessarily symmetric matrix is justified and is assumed in this paper. Provided that K is non-empty, closed and convex, it is well known that the variational inequality (1) has a unique solution. However, for a non-symmetric matrix A the computation of this solution is by no means trivial. Projection-contraction methods like [9, 3] are guaranteed to converge but their practicability depends on an efficient projection back onto the set K. And even then, the number of iterations depends on many user-chosen and problem depended parameters.

In case that the system matrix A is symmetric and positive definite, the variational inequality problem (1) is equivalent to the strictly convex, quadratic minimization problem: Find $x \in K$ such that

$$E(x) = \min_{y\in K} E(y) \tag{3}$$

Email addresses: `lothar.banz@sbg.ac.at` (Lothar Banz), `gregor.milicic@sbg.ac.at` (Gregor Milicic), `andreas.schroeder@sbg.ac.at` (Andreas Schröder)

with the energy function $E(y) := \frac{1}{2}y^\top Ay - y^\top L$. For such a problem a huge number of algorithms are described in literature and we refer to [5, 6, 11, 15] for an overview. Many of these utilize the first order or KKT conditions

$$Ax + B^\top\lambda = L \tag{4a}$$

$$Bx \le g \tag{4b}$$

$$\lambda \ge 0 \tag{4c}$$

$$\lambda^\top(Bx - g) = 0. \tag{4d}$$

Here, $\lambda \in \mathbb{R}^m$ acts as a Lagrange multiplier which is used to incorporate the inequality constraints. An efficient algorithm which is based on the solution of the KKT-System is the dual active set algorithm [13]. Usually, general inequality constraints like $Bx \le g$ and simple box constraints like $x \le g$ are distinguished in many implementations of optimization schemes for the sake of computational efficiency. This is even more important for projection-contraction methods as the projection onto a convex set with box-constraints can be computed efficiently by taking a simple maximum or minimum, whereas in general the projection itself is a non-trivial minimization problem. Moreover, the primal-dual active set method [13] may converge for box constraints but not necessarily for general inequality constraints [4].

The aim of this work is to introduce a basis transformation $I + M^{(\sigma)}$ such that $B(I + M^{(\sigma)})$ is a (generalized) permutation matrix for a special class of matrices B, i.e. the constraints $By \le g$ are transformed to box constraints $\hat{y} \le \hat{g}$ which enables the use of efficient solution schemes. The basis transformation and its inverse is easily and efficiently computable and, equally important, fits well to iterative solution schemes which exploit the possibly sparse structure of the matrix A. In particular, the explicit computation of the transformed matrix $\hat{A} = (I + M^{(\sigma)})^\top A(I + M^{(\sigma)})$ which also results from this transformation can be avoided. Obviously, inequality constraints of the type $g_1 \le By \le g_2$ can also be handled using this basis transformation.

In this paper, we assume that B have at most one non-zero component in each column. This matrix structure generalizes box constraints and arises in the discretization of contact problems and, in particular, in Signorini contact problems of linear elasticity (cf. e.g. [10, 12]). Here, each row of the matrix B contains a vector which is the outer normal vector of a certain boundary point.

The paper is structured as follows: In Section 2 the basis transformation is introduced as an extionsion of a transformation approach originally proposed by Hlavacek et al. in [10]. In Section 3, we discuss an efficient realization of this basis transformation within the successive overrelaxation-procedure with projection (PSOR), see [10, p. 20], [8, p. 67], [7, p. 40] for obstacle problems and [10, p. 20], [12, p. 128] for simplified Signorini problems. In Section 4 a different realization type is presented exemplarily for the primal-dual active set method [13]. Extensive numerical studies analyzing the acceleration technique for P(S)SOR schemes, the effect of the basis transformation on the number of iterations and the CPU-time are presented in Section 5. Detailed proofs, in particular of the global convergence of the sparse and/or accelerated PSOR schemes (c.f. 7, 8,15), are placed in the appendix.

2. Transformation to Box Constraints

In this section, we present a variable transformation which converts the generalized box constraints $Bx \le g$ in the definition of K to the usual box constraints. For this purpose, we formalize the assumed structure of B. For each row index $i \in \{1, \ldots, m\}$ we define the set of non-zero entries as

$$\alpha_i := \{j \in \{1, \ldots, n\} | B_{ij} \neq 0\} \tag{5}$$

and assume that

$$\alpha_i \cap \alpha_k = \emptyset \tag{6}$$

for all $i, k \in \{1, \ldots, m\},\ i \neq k$. The condition (6) implies that in each column $j \in \{1, \ldots, n\}$ at most one non-zero entry exists. Furthermore, for a column index $j \in \{1, \ldots, n\}$, we define the number

$$\gamma_j := \begin{cases} i, & \exists i \in \{1, \ldots, m\} : B_{ij} \neq 0 \\ 0, & \text{otherwise,} \end{cases}$$

which indicates the row index with a non-zero component (if it exists). Finally, we will use the column index $\rho(i) \in \alpha_i$ for a row index $i \in \{1, \ldots, m\}$ of B, so that

$$|B_{i,\rho(i)}| = \max\{|B_{ij}| \mid j \in \alpha_i\}.$$

Without loss of generality $B_{i,\rho(i)} \neq 0$ as rows with only zero components are to be eliminated in a preprocessing step (assuming K is not empty). The index $\rho(i)$ can be interpreted as the pivot index of the row with index i. It is obvious that $\gamma_{\rho(i)} = i$. Furthermore, we define the vectors $\sigma, \kappa \in \mathbb{R}^n$ by

$$\sigma_i := \begin{cases} (1 - B_{\gamma_i,i})/B_{\gamma_i,i} & \text{if } \gamma_i \geq 1, i = \rho(\gamma_i) \\ -B_{\gamma_i,i}/B_{\gamma_i,\rho(\gamma_i)} & \text{if } \gamma_i \geq 1,\ i \neq \rho(\gamma_i) \\ 0 & \text{otherwise,} \end{cases} \qquad \kappa_i := \begin{cases} B_{\gamma_i,i} - 1 & \text{if } \gamma_i \geq 1, i = \rho(\gamma_i) \\ B_{\gamma_i,i} & \text{if } \gamma_i \geq 1,\ i \neq \rho(\gamma_i) \\ 0 & \text{otherwise} \end{cases}$$

with $i \in \{1, \ldots, n\}$. For any vector $\xi \in \mathbb{R}^n$, we define the matrix $M^{(\xi)} \in \mathbb{R}^{n\times n}$ by $(1 \leq i, j \leq n)$

$$M^{(\xi)}_{ij} := \begin{cases} \xi_j & \text{if } \gamma_j \geq 1,\ i = \rho(\gamma_j) \\ 0 & \text{otherwise.} \end{cases}$$

For the basis transformation $\left(I + M^{(\sigma)}\right)\hat{x} = x$ described by the matrix $I + M^{(\sigma)}$, the following results hold. Here and in the following we associate x with the original basis, and the hatted variable $\hat{x}$ with the transformed basis.

Lemma 1. *The matrices $I + M^{(\sigma)}$ and $I + M^{(\kappa)}$ are non-singular. Moreover, $(I + M^{(\sigma)})^{-1} = I + M^{(\kappa)}$.*
Proof. *First, we note $((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ki} = \delta_{ki} + M^{(\kappa)}_{ki} + M^{(\sigma)}_{ki} + (M^{(\kappa)}M^{(\sigma)})_{ki}$ with*

$$(M^{(\kappa)}M^{(\sigma)})_{ki} = \sum_{r=1}^{n} M^{(\kappa)}_{kr}M^{(\sigma)}_{ri} = \begin{cases} \kappa_r\sigma_i & \text{if } \gamma_i \geq 1,\ k = \rho(\gamma_r),\ r = \rho(\gamma_i) \\ 0 & \text{otherwise.} \end{cases}$$

For $i, k, r \in \{1, \ldots, n\}$ with $\gamma_i \geq 1$, $k = \rho(\gamma_r)$ and $r = \rho(\gamma_i)$, we obtain $k = \rho(\gamma_{\rho(\gamma_i)}) = \rho(\gamma_i)$. Moreover, it holds $\kappa_{\rho(\gamma_i)} = B_{\gamma_{\rho(\gamma_i)},\rho(\gamma_i)} - 1 = B_{\gamma_i,\rho(\gamma_i)} - 1$. Hence,

$$(M^{(\kappa)}M^{(\sigma)})_{ki} = \begin{cases} (B_{\gamma_i,\rho(\gamma_i)} - 1)\sigma_i & \text{if } \gamma_i \geq 1,\ k = \rho(\gamma_i) \\ 0 & \text{otherwise,} \end{cases}$$

and

$$((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ki} = \delta_{ki} + \begin{cases} \kappa_i + B_{\gamma_i,\rho(\gamma_i)}\sigma_i & \text{if } \gamma_i \geq 1,\ k = \rho(\gamma_i) \\ 0 & \text{otherwise.} \end{cases}$$

For $k, i \in \{1, \ldots, n\}$ with $\gamma_i = 0$ we immediately obtain that $((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ki} = \delta_{ki}$.
Let $i \in \{1, \ldots, n\}$ with $\gamma_i \geq 1$ and $i \neq \rho(\gamma_i)$, then $((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ii} = 1$. For $k \in \{1, \ldots, n\}\backslash\{i\}$ with $k \neq \rho(\gamma_i)$, we obtain $((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ki} = 0$. For $k = \rho(\gamma_i)$, we get

$$((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ki} = \kappa_i + B_{\gamma_i,\rho(\gamma_i)}\sigma_i = B_{\gamma_i,i} - B_{\gamma_i,\rho(\gamma_i)}B_{\gamma_i,i}/B_{\gamma_i,\rho(\gamma_i)} = 0.$$

Let $\gamma_i \geq 1, i = \rho(\gamma_i)$, then

$$((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ii} = 1 + \kappa_i + B_{\gamma_i,\rho(\gamma_i)}\sigma_i = 1 + \kappa_i + B_{\gamma_i,i}\sigma_i = 1 + (B_{\gamma_i,i} - 1) + B_{\gamma_i,i}(1 - B_{\gamma_i,i})/B_{\gamma_i,i} = 1.$$

If $k \in \{1, \ldots, n\}\backslash\{i\}$, then $k \neq \rho(\gamma_i)$, and therefore $((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ki} = 0$. Summing up all these cases, we obtain $((I + M^{(\kappa)})(I + M^{(\sigma)}))_{ki} = \delta_{ki}$ which completes the proof. ■

Lemma 2. *There holds $B(I + M^{(\sigma)}) =: P$ with $P \in \{0, 1\}^{m\times n}$ such that $\sum_{j=1}^{n} P_{i,j} \leq 1$, i.e. P is a (generalized) permutation matrix. In particular, $(P\hat{y})_j = \hat{y}_{\rho(j)}$ and $PP^\top = I \in \mathbb{R}^{m\times m}$.*
Proof. *For $\hat{y} \in \mathbb{R}^n$ let $\tilde{y} := M^{(\sigma)}\hat{y}$. Then, there holds*

$$\tilde{y}_i = \begin{cases} \sum_{r\in\alpha_{\gamma_i}} \sigma_r\hat{y}_r & \text{if } i = \rho(\gamma_i) \\ 0 & \text{otherwise} \end{cases}$$

and therewith

$$
\begin{aligned}
(B(I+M^{(\sigma)})\hat{y})_j &= (B\hat{y})_j + (B\tilde{y})_j = (B\hat{y})_j + B_{j,\rho(j)} \sum_{r\in\alpha_j} \sigma_r \hat{y}_r = (B\hat{y})_j + B_{j,\rho(j)}\sigma_{\rho(j)}\hat{y}_{\rho(j)} + B_{j,\rho(j)} \sum_{r\in\alpha_j,\, r\neq\rho(j)} \sigma_r\hat{y}_r \\
&= (B\hat{y})_j + B_{j,\rho(j)}(1-B_{\gamma_{\rho(j)},\rho(j)})/B_{\gamma_{\rho(j)},\rho(j)}\hat{y}_{\rho(j)} - \sum_{r\in\alpha_j,\, r\neq\rho(j)} B_{j,r}\hat{y}_r \\
&= (B\hat{y})_j + (1-B_{j,\rho(j)})\hat{y}_{\rho(j)} - \sum_{r\in\alpha_j,\, r\neq\rho(j)} B_{j,r}\hat{y}_r = \hat{y}_{\rho(j)}.
\end{aligned}
$$

Hence, $P_{i,j} = 1$ *if* $\rho(i) = j$ *and* $P_{i,j} = 0$ *otherwise. Therewith,*

$$
(PP^\top)_{i,k} = \sum_{j=1}^{n} P_{i,j}P_{k,j} = P_{i,\rho(j)}P_{k,\rho(j)} = \delta_{i,k}
$$

which completes the proof. ■

From these two results the following desired equivalence results follow immediately.

Theorem 3. *Let* $\hat{A} := (I+M^{(\sigma)})^\top A(I+M^{(\sigma)})$, $\hat{L} := (I+M^{(\sigma)})^\top L$ *and* $\hat{K} := \hat{K}_1 \times \hat{K}_2 \times \cdots \times \hat{K}_n$ *where*

$$
\hat{K}_i := \begin{cases} (-\infty, g_{\gamma_i}], & \gamma_i \geq 1, i = \rho(\gamma_i) \\ (-\infty,\infty), & \text{otherwise,} \end{cases} \tag{7}
$$

i.e. $\hat{v} \in \hat{K} \Leftrightarrow: \hat{v}_i \leq \hat{g}_i$ *for all* $1 \leq i \leq n$*. Then,* $x \in \mathbb{R}^n$ *solves* (1) *if and only if* $\hat{x} = (I+M^{(\sigma)})^{-1}x$ *solves the variational inequality: Find* $\hat{x} \in \hat{K}$ *such that*

$$
(\hat{y}-\hat{x})^\top \hat{A}\hat{x} \geq (\hat{y}-\hat{x})^\top \hat{L} \tag{8}
$$

for all $\hat{y} \in \hat{K}$.

Proof. *Let* $y \in \mathbb{R}^n$ *and* $\hat{y} := (I+M^{(\sigma)})^{-1}y$*. by Lemma 1 it follows that*

$$
(y-x)^\top(I+M^{(\sigma)})^{-\top}(I+M^{(\sigma)})^\top A(I+M^{(\sigma)})(I+M^{(\sigma)})^{-1}x \geq (y-x)^\top(I+M^{(\sigma)})^{-\top}(I+M^{(\sigma)})^\top L
$$

which by construction is the same as

$$
(\hat{y}-\hat{x})^\top\hat{A}\hat{x} \geq (\hat{y}-\hat{x})^\top\hat{L}\,.
$$

Analogously, we obtain for the constraint in $By \leq g$ *by Lemma 2*

$$
By = B(I+M^{(\sigma)})(I+M^{(\sigma)})^{-1}y = P\hat{y} \leq g
$$

and, therewith, $\hat{y}_i \in \hat{K}_i$ *as* $\gamma_{\rho(i)} = i$ *which completes the proof.* ■

Analogously, there hold the following results.

Corollary 4. *Let* $\hat{E}(\hat{y}) := \frac{1}{2}\hat{y}^\top\hat{A}\hat{y} - \hat{y}^\top\hat{L}$*. Then,* x *solves* (3)*, if and only if* $\hat{x} = (I+M^{(\sigma)})^{-1}x$ *solves the minimization problem: Find* $\hat{x} \in \hat{K}$ *such that*

$$
\hat{E}(\hat{x}) = \min_{\hat{y}\in\hat{K}} \hat{E}(\hat{y}). \tag{9}
$$

Moreover, it holds $\hat{E}(\hat{y}) = E(y)$ *for all* $y = (I+M^{(\sigma)})\hat{y}$.

Corollary 5. *The pair* (x,λ) *solves* (4) *if and only if the pair* $(\hat{x},\hat{\lambda})$ *with* $\hat{x} = (I+M^{(\sigma)})^{-1}$ *and* $\hat{\lambda} = \lambda$ *solves the KKT-system*

$$
\begin{aligned}
\hat{A}\hat{x} + P^\top\hat{\lambda} &= \hat{L} && \text{(10a)} \\
P\hat{x} &\leq g && \text{(10b)} \\
\hat{\lambda} &\geq 0 && \text{(10c)} \\
\hat{\lambda}^\top(P\hat{x}-g) &= 0. && \text{(10d)}
\end{aligned}
$$

By construction the matrices A and $\hat{A}$ are congruent. Thus the following lemma follows directly from Lemma 1 and Sylvester's law of inertia.

Lemma 6. *If A is symmetric so is $\hat{A}$ and both matrices have the same number of positive, negative and zero eigenvalues.*

Theorem 3 suggests to solve (1) just by using an appropriate solution procedure for the transformed problem. However, keeping in mind that A and $M^{(\sigma)}$ are sparse, it is not always efficient to expand the matrix product $\hat{A} = (I+M^{(\sigma)})^\top A(I+M^{(\sigma)})$. Therefore, the direct application of solution algorithms to (8) might not be efficient. In the following two sections we demonstrate exemplarily how the above described basis transformation can be embedded into iterative solvers as the accelerated projected successive overrelaxation (PSOR) and a primal-dual active set algorithm. In the PSOR algorithms the basis transformation appears implicitly with all the iterates being stated in the original basis. Whereas, in the active set algorithm that transformation appears explicitly with matrix-matrix-matrix-vector multiplications and with all the iterates being stated in the transformed basis.

3. Accelerated Projective Successive Overrelaxation Solvers

In the following, we present an accelerated projective successive overrelaxation procedure (APSOR) for sparse matrices. This procedure needs the matrix-vector product of the matrix A at most twice. Hence, the computational effort is not significantly higher than in the standard case of box constraints. Only, one additional auxiliary vector $z \in \mathbb{R}^m$ is needed, which enables us to have access to the transformed variables without calculating them explicitly, further details can be found in Appendix A. For its definition we use $\mathcal{M}^{(\xi)} : \mathbb{R}^n \to \mathbb{R}^m$ given by

$$(\mathcal{M}^{(\xi)}(x))_i := \sum_{k\in\alpha_i} \xi_k x_k$$

for $\xi \in \mathbb{R}^n$. In the following $A_{i,:}$ denotes the i-th row vector of A and $\epsilon \in \mathbb{R}^n$ is defined as

$$\epsilon_i := \begin{cases} \omega((1+\sigma_i)A_{ii})^{-1}, & \gamma_i \geq 1, i = \rho(\gamma_i) \\ \omega(A_{ii} + 2\sigma_i A_{\rho(\gamma_i),i} + (\sigma_i)^2 A_{\rho(\gamma_i),\rho(\gamma_i)})^{-1}, & \gamma_i \geq 1,\ i \neq \rho(\gamma_i) \\ \omega A_{ii}^{-1}, & \text{otherwise} \end{cases} \tag{11}$$

with the relaxation parameter $0 < \omega < 2$.

The convergence of the Algorithm 1 without an acceleration step is shown in Theorem 15 and with a specific acceleration in Theorem 8. As mentioned in the algorithm, the procedure is called projective *symmetric* successive overrelaxation (PSSOR) solvers if an additional backward PSOR step is used. Clearly, the possible rate of convergence of the PSOR procedure is limited by the order of convergence of the unconstrained SOR-procedures. The aim of the acceleration step is to reduce the iteration error $\|x - x^{(l)}\|$ by solving a very low dimensional problem. From [14] it is well known that the energy difference is an upper bound for the error, i.e.

$$(x - x^{(l)})^\top A(x - x^{(l)}) =: \left\|x - x^{(l)}\right\|_A^2 \leq 2E(x^{(l)}) - 2E(x) \quad \forall x^{(l)} \in K. \tag{12}$$

That upper bound can be reduced by minimizing $E(\cdot)$ over an appropriate conforming low dimensional subset of K. Let $\{s_i^{(l)}\}_{i=1}^M$ be M linearly independent search directions $s_i^{(l)} \in \mathbb{R}^n$, then the new iterate $\tilde{x}^{(l)} = x^{(l)} + \sum_{i=1}^M \alpha_i s_i^{(l)}$ after the acceleration step is the minimizer of

$$\min_{\alpha\in\mathbb{R}^M} E(x^{(l)} + \sum_{i=1}^M \alpha_i s_i^{(l)}) \quad \text{s.t.} \quad x^{(l)} + \sum_{i=1}^M \alpha_i s_i^{(l)} \in K. \tag{13}$$

In light of Corollary 4, this is equivalent to solving

$$\min_{\alpha\in\mathbb{R}^M} \hat{E}(\hat{x}^{(l)} + \sum_{i=1}^M \alpha_i \hat{s}_i^{(l)}) \quad \text{s.t.} \quad \hat{x}^{(l)} + \sum_{i=1}^M \alpha_i \hat{s}_i^{(l)} \leq \hat{g}. \tag{14}$$

Algorithm 1 Accelerated sparse projective successive overrelaxation solver.

$z^{(0)} = -\mathcal{M}^{(\kappa)} x^{(0)}$ % initialize auxiliary vector
for $l = 0, 1, \ldots$ **do**
  % A single PSOR step
  for $i = 1, \ldots, n$ **do**
    if $\gamma_i = 0$ **then**
      $x_i^{(l+1)} = x_i^{(l)} + \epsilon_i (L_i - A_{i,:} x^{(l)})$ % standard SOR step
    else
      if $i \neq \rho(\gamma_i)$ **then**
        $a = \epsilon_i (L_i - A_{i,:} x^{(l)} + \sigma_i (L_{\rho(\gamma_i)} - A_{\rho(\gamma_i),:} x^{(l)}))$
        $x_i^{(l+1)} = x_i^{(l)} + a$
        $x_{\rho(\gamma_i)}^{(l+1)} = x_{\rho(\gamma_i)}^{(l)} + \sigma_i a$
        $z_{\gamma_i}^{(l+1)} = z_{\gamma_i}^{(l)} + \sigma_i a$
      else
        $b = x_i^{(l)} - z_{\gamma_i}^{(l)}$
        $c = \min\{g_i, b + \epsilon_i (L_i - A_{i,:} x^{(l)})\}$
        $z_{\gamma_i}^{(l+1)} = z_{\gamma_i}^{(l)} + \sigma_i (c - b)$
        $x_i^{(l+1)} = c + z_{\gamma_i}^{(l)}$
      end if
    end if
  end for
  Perform an additional backward PSOR step for Projective symmetric successive overrelaxation
  % Acceleration step
  $s = \frac{x^{(l+1)} - x^{(l)}}{\|x^{(l+1)} - x^{(l)}\|_2}$ % search direction
  $sA = s^\top A$
  $b = s^\top L - sA * s$
  $a = sA * s$
  $ub = g - Bx$
  $t = Bs$
  $\alpha = \frac{b}{a}$ % unconstrained minimizer
  for $i = 1, \ldots, n$ **do**
    if $t_i > 0$ **then**
      $\alpha = \min\{\alpha, ub_i / t_i\}$ % box constraints on α
    else if $t_i < 0$ **then**
      $\alpha = \max\{\alpha, ub_i / t_i\}$
    end if
  end for
  $x^{(l+1)} = x^{(l+1)} + \alpha s$
  $z^{(l+1)} = -\mathcal{M}^{(\kappa)} x^{(l+1)}$
end for

with $\hat{x}^{(l)} = (I + M^{(\sigma)})^{-1} x^{(l)}$ and $\hat{s}_i^{(l)} = (I + M^{(\sigma)})^{-1} s_i^{(l)}$. The choice of the search directions $s_i^{(l)}$ is arbitrary. Choosing $s_i^{(l)}$ to be a basis of the M-th Krylov subspace around $x^{(l)}$ or simply using the difference to the last M iterations might be a practical selection. Due to the multigrid analysis of the SOR we know that depending on ω certain frequency contributions/modes of the initial error $\|x - x^{(0)}\|$ are damped very out fast, and thus span$\{s_i^{(l)}\}$ should best include those error modes which are damped out very slowly.

Theorem 7. *Let $\hat{A}$ be symmetric and positive definite, $\omega \in (0, 2)$ and the acceleration step be the solution procedure to* (14) *with $\hat{s}_i^{(l)}$ linear independent. Then, the accelerated PSOR scheme converges globally to the solution of* (8).
Proof. *The proof can be found in Appendix B.* ■

We can combine Theorem 7 and Theorem 15 by Lemma 6 to proof convergence of the accelerated sparse PSOR scheme given in Algorithm 1.

Theorem 8. *Let A be symmetric and positive definite, $\omega \in (0, 2)$ and the acceleration step be the solution procedure to* (13) *with $s_i^{(l)}$ linear independent. Then, the accelerated sparse PSOR scheme converges globally to the solution of* (1).
Proof. *By Lemma 6 $\hat{A}$ is symmetric and positive definite. By Corollary 4* (13) *and* (14) *are equivalent. By Theorem 7 the accelerated PSOR scheme converges for $\hat{x}$. By construction $z^{(l+1)} = -\mathcal{M}^{(\kappa)} x^{(l+1)}$ for all iterates, and thus Theorem 15 holds verbatim for the accelerated case which gives the desired global convergence.* ■

4. Sparse Primal Dual Active Set Solver

An often used algorithm to solve the KKT problem (10) is the so-called active set method. In this algorithm an index set, the active set $\mathcal{A}$, is iteratively dermined, which indicates those indices, for which the constraint (10b) is active. In each iteration step a corresponding equality constrained minimization problem is solved. If informations of both the primal and dual variable $\hat{x}$ and $\hat{s}$, respectively, is used to update the active set $\mathcal{A}$ the algorithm is called a primal-dual active set method. The basic procedure of the primal-dual active set algorithm by Kunisch and Rendl [13] for box-constraint problems is: Given an active set $\mathcal{A}$ and its complement $\mathcal{I}$, compute $\hat{x}$ and $\hat{s}$ such that

$$\begin{pmatrix} \hat{A}_{\mathcal{A},\mathcal{A}} & \hat{A}_{\mathcal{A},\mathcal{I}} \\ \hat{A}_{\mathcal{I},\mathcal{A}} & \hat{A}_{\mathcal{I},\mathcal{I}} \end{pmatrix} \begin{pmatrix} \hat{x}_{\mathcal{A}} \\ \hat{x}_{\mathcal{I}} \end{pmatrix} + \begin{pmatrix} \hat{s}_{\mathcal{A}} \\ \hat{s}_{\mathcal{I}} \end{pmatrix} = \begin{pmatrix} \hat{L}_{\mathcal{A}} \\ \hat{L}_{\mathcal{I}} \end{pmatrix} \tag{15}$$

with $\hat{s}_{\mathcal{I}} = 0$, $\hat{x}_{\mathcal{A}} = \hat{g}$ and update $\mathcal{A} = \{i \mid \hat{x}_i > \hat{g}_i \text{ or } \hat{s}_i > 0\}$. If the hatted matrix $\hat{A}$ is readily available, $\hat{A}_{\mathcal{I},\mathcal{I}}\hat{x}_{\mathcal{I}} = \hat{L}_{\mathcal{I}} - \hat{A}_{\mathcal{I},\mathcal{A}}\hat{g}$ may be solved efficiently by a preconditioned CG method. In general, with only A and the matrix-vector multiplication $(I + M^{(\sigma)})x$ being available, we find that the matrix-vector multiplication $\hat{A}_{\mathcal{I},\mathcal{I}}\hat{x}_{\mathcal{I}}$ needed for iterative solvers can be easily computed by Algorithm 2. The auxiliary Lagrange multiplier $\hat{s}$ is nothing but the residuum computed by simple matrix-vector multiplications, $\hat{s} = (I + M^{(\sigma)})^\top(L - A(I + M^{(\sigma)})\hat{x})$. From Lemma 2 and Corollary 5 we obtain for the Lagrange multiplier $P^\top\hat{\lambda} = P^\top\lambda = \hat{s}$ the post-processing $\hat{\lambda} = \lambda = PP^\top\hat{\lambda} = P\hat{s}$. We may use Lemma 10 to construct a diagonal preconditioner to (15).

Algorithm 2 Matrix-vector-multiplication $\hat{y} = \hat{A}_{\mathcal{I},\mathcal{I}}\hat{x}_{\mathcal{I}}$ for active set algorithm.

% Given: $A \in \mathbb{R}^{n\times n}$, inactive set $\mathcal{I}$, $x \in \mathbb{R}^{|\mathcal{I}|}$
$z = 0 \in \mathbb{R}^n$ % initialize auxiliary vector
$z(\mathcal{I}) = x$ % expand x by zero to dimension n
$v = (I + M^{(\sigma)})^\top A(I + M^{(\sigma)})z$ % perform standard matrix-vector multiplication with Algorithms 3 and 4
$y = v(\mathcal{I})$ % extract part from inactive set

Algorithm 3 $(I + M^{(\sigma)})$-vector multiplication $z = (I + M^{(\sigma)})x$.

% gather data (maybe precomputed)
$\gamma_{>0} = \gamma > 0$ % position of non-zero entries in γ
$tmp = \rho(\gamma(\gamma_{>0}))$ % row position of non-trivial entry in $M^{(\sigma)}$
$m = length(tmp)$
$[a, perturb] = sort(tmp)$
$\rho = sort(\rho)$
% actual computation
$z = x$ % identity part
$lower = 1$
for $i = 1, \ldots, length(\rho)$ **do**
  for $j = lower, \ldots, m$ **do**
    $k = perturb(j)$
    if $\rho(i) = tmp(k)$ **then**
      $tmp = \gamma_{>0}(k)$
      $z(\rho(i)) = z(\rho(i)) + \sigma(tmp) \cdot x(tmp)$
    else if $\rho(i) < tmp(k)$ **then**
      $lower = j$ % exploit sorting
      break
    end if
  end for
end for

Algorithm 4 $(I + M^{(\sigma)})^\top$-vector multiplication $z = (I + M^{(\sigma)})^\top x$.

% gather data (maybe precomputed)
$\gamma_{>0} = \gamma > 0$
$tmp = \rho(\gamma(\gamma_{>0}))$
$m = length(tmp)$
$[a, perturb] = sort(tmp)$
$\rho = sort(\rho)$
% actual computation
$z = x$
$lower = 1$
for $i = 1, \ldots, length(\rho)$ **do**
  for $j = lower, \ldots, m$ **do**
    $k = perturb(j)$
    if $\rho(i) = tmp(k)$ **then**
      $tmp = \gamma_{>0}(k)$
      $z(tmp) = z(tmp) + \sigma(tmp) \cdot x(\rho(i))$
    else if $\rho(i) < tmp(k)$ **then**
      $lower = j$
      break
    end if
  end for
end for

5. Numerical Results

5.1. *Effect of the acceleration step and parameter ω for an obstacle problem*

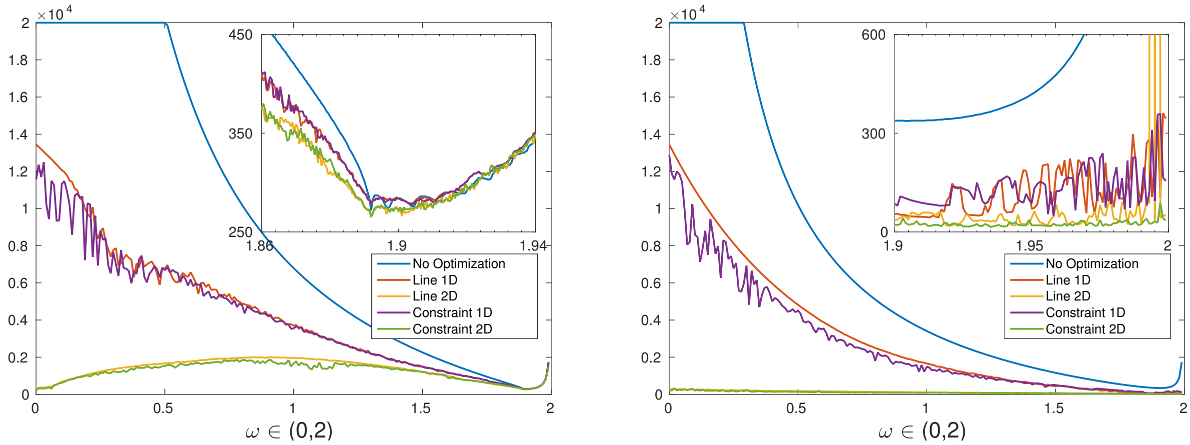
As a model problem to demonstrate the effect of the acceleration step and the choice of the parameter ω within the PSOR and PSSOR we consider the one dimensional obstacle problem:

$$-u'' \geq 1, \quad u \leq 0.35 \quad \text{and} \quad (u - 0.35)(u'' - 1) = 0 \quad \text{in } (-1, 1), \qquad u(\pm 1) = 0. \tag{16}$$

Solving that problem approximately by the lowest order conforming finite element method on a uniform mesh yields

$$A_{ij} = \begin{cases} 2h^{-1}, & \text{if } i = j \\ -h^{-1}, & \text{if } |i - j| = 1 \\ 0, & \text{otherwise} \end{cases}, \qquad L_j = h, \qquad g_j = 0.35 \qquad (1 \leq i, j \leq n) \tag{17}$$

with $h = 2/(n + 1)$ and dimension n. As the problem has box constraints, no transformation is used and the problem can be solved directly by the PSOR and PSSOR.
Figure 1 shows the dependency of the number of iterations of the different accelerated P(S)OR schemes on the relaxation parameter ω. For that we chose $n = 127$ and stop the iteration once $\|x_{P(S)SOR} - x_{AS}\|_A < 10^{-8}$ is reached with the "exact" solution calculated by the primal dual active set algorithm. We use the difference of the current iteration to the last one (two) iterations as search direction(s) s_i in the acceleration step, resulting in an optimization problem in one (two) dimension(s), respectively. We differ between two acceleration procedures. For the first one, indicated by l1D or l2D, the unconstrained optimization problem is solved and the minimizer is projected back into the feasible set minimizing the Euclidean distance. For the second one, called c1D or c2D, the constrained minimization problem (14) is solved. Not surprisingly, the number of iterations depends heavily on the choice of ω. For the optimal ω, the acceleration procedure within the PSOR has no significant effect on the number of iterations. However, with acceleration the number of iterations are much more robust with respect to changes in ω, which eases the burden of selecting an appropriate ω for practical computations. In case of the PSSOR scheme the acceleration step always improves the number of iteration significantly and is by far superior to the PSOR scheme.



(a) PSOR

(b) PSSOR

Figure 1: Number of iterations for the one dimensional obstacle problem with $n = 127$ for different values of $0 < \omega < 2$.

The experimentally determined best choices of $\omega \in (0, 2)$ depending on the level of refinement can be found in Table 1, along with the number of iterations needed for the procedures. We use the same stopping criterion as for the first experiment and make four observations. Firstly, the acceleration step does not effect the optimal value for ω strongly. A value of ω close to, but not exactly 2 should be used for a higher number of degrees of freedom. Secondly, the acceleration does not effect significantly the growth rate of the number of iterations with respect to the dimension n.

Thirdly, using two search directions for the acceleration step significantly reduces the number of iterations needed in comparison to the PSSOR procedure without any acceleration steps. Fourthly, using a two dimensional line search (l2D) halves surprisingly the number of iterations compared to using the constraint two dimensional accelerated step (c2D), making it, together with the easier implementation and faster computation of a solution to the accelerating optimization problem, the better method.

	PSSOR		PSSOR l1D		PSSOR c1D		PSSOR l2D		PSSOR c2D	
Dofs	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}
31	81	1.628	23	1.7325	19	1.626	16	1.6005	9	1.766
63	167	1.8175	29	1.855	29	1.926	15	1.93	11	1.9435
127	337	1.9055	42	1.875	43	1.869	18	1.9555	15	1.9345
255	674	1.952	71	1.933	61	1.9575	22	1.973	23	1.9435
511	1350	1.976	140	1.9715	106	1.967	29	1.9725	39	1.963
1023	2702	1.9875	322	1.9735	213	1.99	33	1.9895	74	1.9745
2047	5412	1.9935	736	1.98	401	1.9925	67	1.9885	137	1.9905
4095	10828	1.997	1585	1.9865	962	1.9985	140	1.985	222	1.9935
8191			3198	1.989	1690	1.999	181	1.9955	442	1.997
16383					3173	1.999	315	1.999	605	1.997
32767							686	1.998	1195	1.997
$O(\text{Dofs}^{\alpha})$	1.00		1.15		1.02		0.79		0.76	

Table 1: Minimum number of iterations and optimal values of ω for different levels of refinement for the PSSOR methods.

5.2. Effect of basis transformation on contact solvers

To analyze the effect of the basis transformation, we consider a standard, lowest order finite element discretization of the Signorini problem to find a function $u \in [H^1(\Omega)]^3$ such that

$$
\begin{aligned}
-\operatorname{div}\sigma(u) &= f && \text{in } \Omega && \text{(18a)}\\
\sigma(u) &= C : \epsilon(u) && \text{in } \Omega && \text{(18b)}\\
u &= 0 && \text{on } \Gamma_D && \text{(18c)}\\
\sigma(u)n &= 0 && \text{on } \Gamma_N && \text{(18d)}\\
\sigma_n \le 0,\ u_n \le g,\ \sigma_n(u_n - g) = 0,\ \sigma_t &= 0 && \text{on } \Gamma_C. && \text{(18e)}
\end{aligned}
$$

As usual, n is the outward unit normal and $\sigma_n, u_n \in \mathbb{R}$, $\sigma_t, u_t \in \mathbb{R}^2$ are the normal, tangential components of $\sigma(u)n$, u, respectively and (18b) describes Hooke's law with Young's modulus of elasticity $E = 2$, Poisson's ratio $\nu = 0.42$ and linearized strain tensor $\epsilon(u) = \frac{1}{2}(\nabla u + \nabla u^\top)$.

The domain Ω with the polar coordinates (r, φ) and the origin in $(-1.625, 0)$ is a transformation of the set

$$
\left\{ (x_1, x_2, x_3) \in \mathbb{R}^3 \,\middle|\, \begin{array}{l} r(x_1, x_3) \in (1.295, 1.625),\\ \varphi(x_1, x_3) \in [0, \pi/4) \cup (7\pi/4, 0]),\\ x_2 \in (-0.575, 0.575) \end{array} \right\}
$$

so that the resulting boundary is curved and has normal vectors in different directions. The volume force $f = (0, 0, -2)^\top$ and the gap function

$$
g := \begin{cases} d + 1 - \sqrt{1 - (x_3 + 0.5x_2)^2}, & |x_3 + 0.5x_2| \le r \\ d + 1, & |x_3 + 0.5x_2| > r, \end{cases}
$$

are given. Parameter $d \in \mathbb{R}$ describes the displacement of the obstacle in x_1 direction, i.e. how far the obstacle is pushed against the workpiece. We choose $d = 0.25$ for our experiments. The boundary of the domain Ω is decomposed into $\Gamma_D := \{(x_1, x_2, x_3) \in \bar{\Omega} \mid r(x_1, x_3) = 1.295\}$, $\Gamma_N := \partial\Omega \setminus (\Gamma_D \cup \Gamma_C)$ and $\Gamma_C := \{(x_1, x_2, x_3) \in \Gamma_1 \mid r(x_1, x_3) = 1.625\}$.

The reference configuration and deformation of the workpiece is shown in Figure 2. The discrete weak formulation of (18) is in fact (1) where B consists of the normal vectors n associated with the global degrees of freedom. Due to the curvature of the surface, all normal vectors show in different directions. To be able to solve that problem by a PSSOR scheme or primal-dual active set method, the basis transformation must be carried out either explicitly or implicitly. Here, the latter is called the sparse variant. Throughout this section we use the same stopping criterion as for the obstacle problem of section 5.1 but with the smaller tolerance 10^{-12}.

(a) Workpiece (b) Workpiece after deformation process

Figure 2: Workpiece before and after deformation process.

Theoretically, as stated in Theorem 15, the explicitly transformed and the sparse variant of the PSSOR scheme should give exactly the same sequence of iterates, thus we only implemented the line search acceleration variants in the sparse procedures. However, due to small rounding errors the iterates of the non-accelerated PSSOR are slightly different. As the search direction/low dimensional set over which (14) is solved depends explicitly on the sequences of iterates, the minimization problems of the sparse and non-sparse variants are no longer equivalent, and hence, the rounding errors are blown up, leading to two different but both converging sequences of iterates. To compensate for that effect we cut off the search directions after the 8th decimal place. The results are stated in Table 2. As for the obstacle problem, an acceleration step significantly improves the number of iterations and computation time. Moreover, the growth rate of the number of iterations with respect to the degrees of freedom is halved by the acceleration techniques l2D and c2D, compared to l1D, c1D and no acceleration.

	PSSOR		PSSOR l1D		PSSOR c1D		PSSOR l2D		PSSOR c2D	
Dofs	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}
18	56	1.0	30	1.0	28	1.0	18	1.0	14	1.0
90	132	0.9	71	1.0	71	1.0	32	1.0	30	0.875
540	408	0.81	175	1.186	67	0.79	53	0.675	59	0.7955
3672	1534	0.925	733	0.895	433	0.766	106	0.925	99	0.825
26928	5933	1.0	2965	0.99	1696	0.87	208	0.95	198	0.835
$O(\text{Dofs}^{\alpha})$	0.67		0.67		0.60		0.33		0.32	
	SPSSOR		SPSSOR l1D				SPSSOR l2D			
	$Iter_{min}$	ω_{opt}	$Iter_{min}$	ω_{opt}			$Iter_{min}$	ω_{opt}		
18	56	1.0	30	1.0			13	0.96		
90	132	0.88	71	1.0			31	1.0		
540	408	0.805	178	1.1595			52	0.805		
3672	1534	0.92	772	0.91			103	0.77		
26928	5933	1.0	2971	1.0			208	1.0		
$O(\text{Dofs}^{\alpha})$	0.67		0.67				0.34			

Table 2: Minimum number of iterations and optimal values of ω for different levels of refinement for the PSSOR and SPSSOR methods.

Table 3 shows the needed CPU time in seconds of our MATLAB implementation on a desktop computer with

8 Intel i7-2700 CPU @ 3.40GHz cores. The iterative solver within the primal-dual active set (AS) algorithm is the diagonal preconditioned CG-method.

Dofs	Transformation	PSSOR l2D	PSSOR c2D	SPSSOR l2D	AS transformed	AS sparse
18	0.0035	0.0069	0.0127	0.0041	0.0022	0.0026
90	0.0030	0.0048	0.0099	0.0074	0.0024	0.0061
540	0.0159	0.0168	0.1419	0.0390	0.0115	0.0220
3672	0.0925	0.1584	0.4335	0.2739	0.2917	0.3932
26928	0.7642	2.9856	8.2209	3.6433	3.8786	4.4129
205920	6.9481	49.8209	233.7480	57.4561	124.5383	131.1108
1609920	75.8780	719.6330	6459.7379	780.9860	2243.3318	2354.1681
$\mathcal{O}(\text{Dofs}^\alpha)$	1.10	1.38	1.59	1.31	1.49	1.45

Table 3: Time in seconds required for different levels of refinement for the different methods with $\omega = 0.95$ for the PSSOR procedures.

The explicitly transformed methods for both, the active set and the PSSOR procedures, need less computation time to converge, as no computation of the basis transformation on the fly is carried out. But when adding the time for the transformation, the transformed and sparse variants are about equality fast, but with the sparse variant requiring less memory. We observe, that the PSSOR and SPSSOR method using a two dimensional line search acceleration step are faster then the active set procedures, leading us to the conclusion that the proposed basis transformation in conjunction with the simple to implement PSSOR scheme is a very practical choice to solve the class of variational inequalities (1) with B satisfying (6).

Appendix A. Analysis of the Sparse PSOR

A single sparse PSOR step in the i-th component can be described by the mappings $S_i^* : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $T_i^* : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ where $S_i^*(x,z)$ returns the updated x and $T_i^*(x,z)$ returns the updated z. Thus, a single sparse PSOR step is the composition of the mappings S_i^* and T_i^* to

$$(S^*, T^*) := (S_n^*, T_n^*) \circ (S_{n-1}^*, T_{n-1}^*) \circ \ldots \circ (S_2^*, T_2^*) \circ (S_1^*, T_1^*).$$

And, therewith, the sparse PSOR generates the sequence $\{x^l, z^l\}_{l\in\mathbb{N}}$ defined by

$$(x^{l+1}, z^{l+1}) := (S^*, T^*)(x^l, z^l) \tag{A.1}$$

for a given initial solution $(x^0, -\mathcal{M}^{(\kappa)}(x^0)) \in \mathbb{R}^n \times \mathbb{R}^m$. For the convergence result we need additionally the mapping $\hat{S}_i : \mathbb{R}^n \to \mathbb{R}^n$ defined by

$$(\hat{S}_i(x))_j := \begin{cases} \hat{P}_i(x_i + \delta_i(\hat{L}_i - \hat{A}_{i,\cdot}x)), & i = j \\ x_j, & \text{otherwise.} \end{cases}$$

Here, $\delta \in \mathbb{R}^n$ is given by $\delta_i := \omega \hat{A}_{ii}^{-1}$ and $\hat{P}_i$ is the projection $\hat{P}_i : \mathbb{R} \to \hat{K}_i$:

$$\hat{P}_i(s) := \begin{cases} g_i, & \gamma_i \geq 1, i = \rho(\gamma_i),\ s > g_i \\ s, & \text{otherwise.} \end{cases}$$

Finally, we need the mapping $\hat{S} : \mathbb{R}^n \to \mathbb{R}^n$ with $\hat{S} := \hat{S}_n \circ \hat{S}_{n-1} \circ \ldots \circ \hat{S}_2 \circ \hat{S}_1$.

Lemma 9. *Let $x \in \mathbb{R}^n$. Then there holds $-M^{(\kappa)}x = M^{(\sigma)}(I + M^{(\kappa)})x$ and $-\mathcal{M}^{(\kappa)}(x) = \mathcal{M}^{(\sigma)}((I + M^{(\kappa)})x)$.*

Proof. *$M^{(\sigma)}(I + M^{(\kappa)})x = (I + M^{(\sigma)})(I + M^{(\kappa)})x - (I + M^{(\kappa)})x = x - (I + M^{(\kappa)})x = -M^{(\kappa)}x$. The second assertion follows with the first assertion and the definition of $\mathcal{M}^{(\xi)}$.* ■

Lemma 10. *For all $1 \le i \le n$ there holds $\omega \hat{A}_{ii}^{-1} = \hat{\delta}_i$ with $\omega \in (0,2)$ arbitrary, ϵ defined in* (11) *and*

$$\hat{\delta}_i = \begin{cases} \epsilon_i(1+\sigma_i)^{-1}, & \gamma_i \ge 1, i = \rho(\gamma_i) \\ \epsilon_i, & \text{otherwise.} \end{cases}$$

Proof. *If $\gamma_i = 0$, then $\hat{A}_{ii} = A_{ii}$. If $\gamma_i \ge 1$ and $i \neq \rho(\gamma_i)$, we have $\hat{A}_{ii} = A_{ii} + 2((M^{(\sigma)})^\top A)_{ii} + ((M^{(\sigma)})^\top A M^{(\sigma)})_{ii} = A_{ii} + 2\sigma_i A_{\rho(\gamma_i),i} + (\sigma_i)^2 A_{\rho(\gamma_i),\rho(\gamma_i)}$. Finally, if $\gamma_i \ge 1, i = \rho(\gamma_i)$, we obtain $\hat{A}_{ii} = A_{ii} + 2(M^{(\sigma)}A)_{ii} + ((M^{(\sigma)})^\top A M^{(\sigma)})_{ii} = A_{ii} + 2\sigma_i A_{ii} + (\sigma_i)^2 A_{ii} = (1+\sigma_i)^2 A_{ii}$. Therefore, in all three cases we obtain $\omega \hat{A}_{ii}^{-1} = \hat{\delta}_i$.* ■

Lemma 11. *For all $1 \le i \le n$ and $x \in \mathbb{R}^n$ there holds*

$$(S_i^*, T_i^*)(x, -\mathcal{M}^{(\kappa)}(x)) = ((I + M^{(\sigma)})\hat{S}_i((I + M^{(\kappa)})x), \mathcal{M}^{(\sigma)}(\hat{S}_i((I + M^{(\kappa)})x))).$$

Proof. *Let $\hat{x} := (I + M^{(\kappa)})x$ and $z := -\mathcal{M}^{(\kappa)}(x) = \mathcal{M}^{(\sigma)}(\hat{x})$. From Lemma 10 we have $(\hat{S}_i(\hat{x}))_i = \hat{P}_i(\hat{x}_i + \hat{\delta}_i(\hat{L}_i - \hat{A}_{i,\cdot}\hat{x}))$. If $\gamma_i = 0$, then, for $k \neq i$, it holds*

$$\hat{x}_k = (\hat{S}_i(\hat{x}))_k \qquad \text{and} \qquad (M^{(\sigma)}\hat{x})_k = (M^{(\sigma)}\hat{S}_i(\hat{x}))_k. \tag{A.2}$$

Thus, we have

$$(S_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_k = x_k = ((I + M^{(\sigma)})\hat{x})_k = ((I + M^{(\sigma)})\hat{S}_i(\hat{x}))_k = ((I + M^{(\sigma)})\hat{S}_i((I + M^{(\kappa)})x))_k. \tag{A.3}$$

Furthermore, there is $\hat{x}_i = x_i$ and $\hat{L}_i = L_i$. Since $i \neq \rho(\gamma_r)$ for all $r \in \{1, \ldots, n\}$ with $\gamma_r \ge 1$, we obtain $((I + M^{(\sigma)})^\top)_{ir} = \delta_{ir}$ and

$$\hat{A}_{i,\cdot}\hat{x} = ((I + M^{(\sigma)})^\top A (I + M^{(\sigma)})(I + M^{(\kappa)})x)_i = ((I + M^{(\sigma)})^\top A x)_i = A_{i,\cdot}x.$$

Consequently, it follows

$$(S_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_i = x_i + \epsilon_i(L_i - A_{i,\cdot}x) = \hat{x}_i + \epsilon_i(\hat{L}_i - \hat{A}_{i,\cdot}\hat{x}) = (\hat{S}_i(\hat{x}))_i = ((I + M^{(\sigma)})\hat{S}_i((I + M^{(\kappa)})x))_i.$$

Now we consider T_i^*. Since $\mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{x})) = \mathcal{M}^{(\sigma)}(\hat{x})$, we have

$$T_i^*(x, -\mathcal{M}^{(\kappa)}(x)) = T_i^*(x, z) = z = \mathcal{M}^{(\sigma)}(\hat{x}) = \mathcal{M}^{(\sigma)}(\hat{S}_i((I + M^{(\kappa)})x)).$$

If $\gamma_i \ge 1$ and $i \neq \rho(\gamma_i)$, then, for $k \notin \{i, \rho(\gamma_i)\}$, we also have (A.2) *and, therefore,* (A.3). *Moreover, it holds:*

$$\begin{aligned} &\hat{x}_i = x_i, \ \hat{L}_i = ((I + M^{(\sigma)})^\top L)_i = L_i + \sigma_i L_{\rho(\gamma_i)} \quad \text{and} \\ &\hat{A}_{i,\cdot}\hat{x} = ((I + M^{(\sigma)})^\top A (I + M^{(\sigma)}))_{i,\cdot}(I + M^{(\kappa)})x = (A + (M^{(\sigma)})^\top A)_{i,\cdot}x = A_{i,\cdot}x + \sigma_i A_{\rho(\gamma_i),\cdot}x. \end{aligned}$$

With $a := \epsilon_i(L_i - A_{i,\cdot}x + \sigma_i(L_{\rho(\gamma_i)} - A_{\rho(\gamma_i),\cdot}x)) = \epsilon_i(\hat{L}_i - \hat{A}_{i,\cdot}\hat{x})$, we get

$$(S_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_i = x_i + a = \hat{x}_i + \epsilon_i(\hat{L}_i - \hat{A}_{i,\cdot}\hat{x}) = (\hat{S}_i(\hat{x}))_i = ((I + M^{(\sigma)})\hat{S}_i((I + M^{(\kappa)})x))_i.$$

From

$$\mathcal{M}^{(\sigma)}(\hat{x})_{\gamma_i} + \sigma_i a = \sigma_i(\hat{x}_i + a) + \sum_{k \in \alpha_{\gamma_i}, k \neq i} \sigma_k \hat{x}_k = \sigma_i(\hat{S}_i(\hat{x}))_i + \sum_{k \in \alpha_{\gamma_i}, k \neq i} \sigma_k \hat{x}_k = \mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{x}))_{\gamma_i}, \tag{A.4}$$

we get

$$\begin{aligned} (S_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_{\rho(\gamma_i)} &= x_{\rho(\gamma_i)} + \sigma_i a = \hat{x}_{\rho(\gamma_i)} + (\mathcal{M}^{(\sigma)}(\hat{x}))_{\gamma_i} + \sigma_i a = \hat{x}_{\rho(\gamma_i)} + (\mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{x})))_{\gamma_i} = ((I + M^{(\sigma)})\hat{S}_i(\hat{x}))_{\rho(\gamma_i)} \\ &= ((I + M^{(\sigma)})\hat{S}_i((I + M^{(\kappa)})x))_{\rho(\gamma_i)}. \end{aligned}$$

Furthermore, we obtain

$$(T_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_j = (T_i^*(x, z))_j = z_j = (\mathcal{M}^{(\sigma)}(\hat{x}))_j = (\mathcal{M}^{(\sigma)}((I + M^{(\kappa)})x))_j \tag{A.5}$$

for $j \neq \gamma_i$ and, by using (A.4),

$$(T_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_{\gamma_i} = (T_i^*(x,z))_{\gamma_i} = z_{\gamma_i} + \sigma_i a = (\mathcal{M}^{(\sigma)}(\hat{x}))_{\gamma_i} + \sigma_i a = (\mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{x}))_{\gamma_i} = (\mathcal{M}^{(\sigma)}(\hat{S}_i((I + M^{(\kappa)})x))_{\gamma_i}.$$

If $\gamma_i \geq 1, i = \rho(\gamma_i)$, then, for $k \neq i$, we also obtain (A.2) *and, therefore,* (A.3) *holds. Furthermore, there is*

$$\hat{x}_i = x_i + (M^{(\kappa)}x)_i = x_i + (\mathcal{M}^{(\kappa)}(x))_{\gamma_i} = x_i - z_{\gamma_i},\ \hat{L}_i = ((I + M^{(\sigma)})^\top L)_i = L_i + \sigma_i L_i = (1 + \sigma_i)L_i \quad \text{and}$$
$$\hat{A}_{i,\cdot}\hat{x} = ((I + M^{(\sigma)})^\top A(I + M^{(\sigma)}))_{i,\cdot}(I + M^{(\kappa)})x = (A + (M^{(\sigma)})^\top A)_{i,\cdot}x = A_{i,\cdot}x + \sigma_i A_{\rho(\gamma_i),\cdot}x = (1 + \sigma_i)A_{i,\cdot}x.$$

For $j \neq \gamma_i$, we have (A.5). *In addition, it holds*

$$(\mathcal{M}^{(\sigma)}(\hat{x}))_{\gamma_i} + \sigma_i((\hat{S}_i(\hat{x})_i - \hat{x}_i) = \sigma_i \hat{S}_i(\hat{x})_i + \sum_{k \in \alpha_{\gamma_i}, k \neq i} \sigma_i \hat{x}_i = (\mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{x})))_{\gamma_i}.$$

With $b := x_i - z_{\gamma_i} = \hat{x}_i$ and $c := \hat{P}_i(b + \epsilon_i(L_i - A_{i,\cdot}x)) = (\hat{S}_i(\hat{x}))_i$, we get

$$\begin{aligned}(T_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_{\gamma_i} &= (T_i^*(x,z))_{\gamma_i} = z_{\gamma_i} + \sigma_i(c - b) = (\mathcal{M}^{(\sigma)}(\hat{x}))_{\gamma_i} + \sigma_i((\hat{S}_i(\hat{x})_i - \hat{x}_i) = (\mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{x})))_{\gamma_i} \\ &= (\mathcal{M}^{(\sigma)}(\hat{S}_i((I + M^{(\kappa)})x)))_{\gamma_i}\end{aligned}$$

and

$$\begin{aligned}(S_i^*(x), -\mathcal{M}^{(\kappa)}(x))_i &= c + (T_i^*(x, -\mathcal{M}^{(\kappa)}(x)))_{\gamma_i} = (\hat{S}_i(\hat{x}))_i + (\mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{x})))_{\gamma_i} = ((I + M^{(\sigma)})\hat{S}_i(\hat{x}))_i \\ &= ((I + M^{(\sigma)})\hat{S}_i((I + M^{(\kappa)})x))_i.\end{aligned}$$

■

Lemma 12. *Let $x \in \mathbb{R}^n$, then*

$$(S^*, T^*)(x, -\mathcal{M}^{(\kappa)}(x)) = ((I + M^{(\sigma)})\hat{S}((I + M^{(\kappa)})x), \mathcal{M}^{(\sigma)}(\hat{S}((I + M^{(\kappa)})x))).$$

Proof. *Let $(S^{*,i}, T^{*,i}) := (S_i^*, T_i^*) \circ \ldots \circ (S_1^*, T_1^*)$ and $\hat{S}^i := \hat{S}_i \circ \ldots \circ \hat{S}_1$ for $i \in \{1, \ldots, n\}$. Lemma 11 yields $(S^{*,1}, T^{*,1})(x, -\mathcal{M}^{(\kappa)}(x)) = ((I + M^{(\sigma)})\hat{S}^1((I + M^{(\kappa)})x), \mathcal{M}^{(\sigma)}(\hat{S}^1((I + M^{(\kappa)})x))$. By induction, it follows from Lemma 9 and Lemma 11 for $i > 1$, that*

$$\begin{aligned}(S^{*,i}, T^{*,i})(x, -\mathcal{M}^{(\kappa)}(x)) &= ((S_i^*, T_i^*) \circ (S^{*,i-1}, T^{*,i-1}))(x, -\mathcal{M}^{(\kappa)}(x)) \\ &= (S_i^*, T_i^*)((I + M^{(\sigma)})\hat{S}^{i-1}((I + M^{(\kappa)})x), \mathcal{M}^{(\sigma)}(\hat{S}^{i-1}((I + M^{(\kappa)})x)) \\ &= (S_i^*, T_i^*)((I + M^{(\sigma)})\hat{S}^{i-1}((I + M^{(\kappa)})x), -\mathcal{M}^{(\kappa)}((I + M^{(\sigma)})\hat{S}^{i-1}((I + M^{(\kappa)})x))) \\ &= ((I + M^{(\sigma)})\hat{S}_i((I + M^{(\kappa)})(I + M^{(\sigma)})\hat{S}^{i-1}((I + M^{(\kappa)})x)),\end{aligned}$$

$$\begin{aligned}\mathcal{M}^{(\sigma)}(\hat{S}_i((I + M^{(\kappa)})(I + M^{(\sigma)})\hat{S}^{i-1}((I + M^{(\kappa)})x))) &= ((I + M^{(\sigma)})\hat{S}_i(\hat{S}^{i-1}((I + M^{(\kappa)})x)), \mathcal{M}^{(\sigma)}(\hat{S}_i(\hat{S}^{i-1}((I + M^{(\kappa)})x))) \\ &= ((I + M^{(\sigma)})\hat{S}^i((I + M^{(\kappa)})x)), \mathcal{M}^{(\sigma)}(\hat{S}^i((I + M^{(\kappa)})x))).\end{aligned}$$

Since $(S^*, T^*) = (S^{*,n}, T^{*,n})$ and $\hat{S} = \hat{S}^n$, the proof is complete. ■

Lemma 13. *Let $\{x^l, z^l\}_{l \in \mathbb{N}}$ be given as in* (A.1). *Then $z^{l+1} = -\mathcal{M}^{(\kappa)}(x^{l+1})$.*

Proof. *By induction, it follows from Lemma 9 and Lemma 12, that*

$$\begin{aligned}z^{l+1} &= T^*(x^l, z^l) = T^*(x^l, -\mathcal{M}^{(\kappa)}(x^l)) = \mathcal{M}^{(\sigma)}(\hat{S}((I + M^{(\kappa)})x^l)) = \mathcal{M}^{(\sigma)}((I + M^{(\kappa)})(I + M^{(\sigma)})\hat{S}((I + M^{(\kappa)})x^l)) \\ &= \mathcal{M}^{(\sigma)}((I + M^{(\kappa)})S^*(x^l, -\mathcal{M}^{(\kappa)}(x^l)) = \mathcal{M}^{(\sigma)}((I + M^{(\kappa)})S^*(x^l, z^l)) = \mathcal{M}^{(\sigma)}((I + M^{(\kappa)})x^{l+1}) = -\mathcal{M}^{(\kappa)}(x^{l+1}).\end{aligned}$$

■

Lemma 14. *Let* $\{x^l, z^l\}_{l\in\mathbb{N}}$ *be given as in* (A.1)*. For any sequence* $\{\hat{x}^l\}_{l\in\mathbb{N}}$*, we have*

$$\hat{x}^l = (I + M^{(\kappa)})x^l \quad \Leftrightarrow \quad \hat{x}^{l+1} = \hat{S}(\hat{x}^l),\ \hat{x}^0 = (I + M^{(\kappa)})x^0.$$

Proof. *Let* $\hat{x}^l = (I + M^{(\kappa)})x^l$ *be fulfilled, then we know from Lemma 12 and 13, that*

$$\begin{aligned}\hat{x}^{l+1} &= (I + M^{(\kappa)})x^{l+1} = (I + M^{(\kappa)})S^*(x^l, z^l) = (I + M^{(\kappa)})S^*(x^l, -\mathcal{M}^{(\kappa)}(x^l)) = (I + M^{(\kappa)})(I + M^{(\sigma)})\hat{S}((I + M^{(\kappa)})x^l)\\ &= \hat{S}(\hat{x}^l).\end{aligned}$$

Let $\hat{x}^{l+1} = \hat{S}(\hat{x}^l)$ *be fulfilled, then it holds by induction*

$$\hat{x}^{l+1} = \hat{S}(\hat{x}^l) = (I + M^{(\kappa)})(I + M^{(\sigma)})\hat{S}((I + M^{(\kappa)})x^l) = (I + M^{(\kappa)})S^*(x^l, -\mathcal{M}^{(\kappa)}(x^l)) = (I + M^{(\kappa)})S^*(x^l, z^l) = (I + M^{(\kappa)})x^{l+1}.$$

■

Theorem 15. *Let* $\{x^l, z^l\}_{l\in\mathbb{N}}$ *be given as in* (A.1) *and let the PSOR scheme for* $\hat{x}$ *be converging. For the solution* $x \in K$ *of* (3)*, it holds*

$$\lim_{l\to\infty} x^l = x.$$

Proof. *Let* $\hat{x}^l := (I + M^{(\kappa)})x^l$*. From Lemma 14 we obtain* $\hat{x}^{l+1} = \hat{S}(\hat{x}^l)$*. The convergence of the PSOR-procedure yields* $\lim_{l\to\infty} \hat{x}^l = \hat{x} \in \hat{K}$*, where* $\hat{E}(\hat{x}) = \min_{\hat{y}\in\hat{K}} \hat{E}(\hat{y})$*. From Corollary 4 it follows, that*

$$x = (I + M^{(\sigma)})\hat{x} = (I + M^{(\sigma)}) \lim_{l\to\infty} \hat{x}^l = \lim_{l\to\infty}(I + M^{(\sigma)})\hat{x}^l = \lim_{l\to\infty} x^l. \tag{A.6}$$

■

Appendix B. Proof of Theorem 7

For the ease of presentation we neglect the hat in $\hat{x}$. The accelerated PSOR scheme is

$$\widetilde{x}_i^{(k)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}\overline{x}_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}\right) \tag{B.1a}$$

$$\overline{x}_i^{(k+1)} = \min\{\widetilde{x}_i^{(k)}, g_i\} \tag{B.1b}$$

for $1 \leq i \leq n$ and then

$$x^{(k+1)} = \arg\min_{\alpha\in\mathbb{R}^M} E(\overline{x}^{(k+1)} + \sum_{i=1}^{M} \alpha_{i,k}s_{i,k}) \quad \text{s.t.} \quad \overline{x}^{(k+1)} + \sum_{i=1}^{M} \alpha_{i,k}s_{i,k} \leq g. \tag{B.2}$$

Due to the projection, it is easy to see that

$$\exists\, \eta_{i,k} \in [0,1]: \quad \overline{x}_i^{(k+1)} = x_i^{(k)} + \eta_{i,k}\left(\widetilde{x}_i^{(k)} - x_i^{(k)}\right).$$

We set

$$\overline{x}^{(k,i)} = \left(\overline{x}_1^{(k+1)}, \cdots, \overline{x}_i^{(k+1)}, x_{i+1}^{(k)}, \cdots x_n^{(k)}\right)^\top,$$

i.e. $\overline{x}^{(k,i)}$ is the i-th iteration within one PSOR step. If $\eta_{i,k} = 0$, then $\overline{x}^{(k,i)} = \overline{x}^{(k,i-1)}$ and thus $E(\overline{x}^{(k,i)}) = E(\overline{x}^{(k,i-1)})$. If $\eta_{i,k} \in (0,1]$, there holds

$$E(\overline{x}^{(k,i)}) - E(\overline{x}^{(k,i-1)}) = \frac{1}{2}(\overline{x}^{(k,i)} - \overline{x}^{(k,i-1)})^\top A(\overline{x}^{(k,i)} - \overline{x}^{(k,i-1)}) + (\overline{x}^{(k,i)} - \overline{x}^{(k,i-1)})^\top(A\overline{x}^{(k,i-1)} - L) \tag{B.3}$$

$$= \frac{a_{ii}}{2}(\overline{x}_i^{(k+1)} - x_i^{(k)})^2 + (\overline{x}_i^{(k+1)} - x_i^{(k)})\frac{a_{ii}}{\omega}(x_i^{(k)} - \widetilde{x}_i^{(k)}) = \frac{a_{ii}}{2}(1 - \frac{2}{\omega\eta_{i,k}})(\overline{x}_i^{(k+1)} - x_i^{(k)})^2 \leq 0 \tag{B.4}$$

since

$$(A\overline{x}^{(k,i-1)} - L)_i = \sum_{j=1}^{i-1} a_{ij}\overline{x}_j^{(k+1)} + a_{ii}x_i^{(k)} + \sum_{j=i+1}^{n} a_{ij}x_j^{(k)} - (\widetilde{x}_i^{(k)} + (\omega-1)x_i^{(k)})\frac{a_{ii}}{\omega} - \sum_{j=1}^{i-1} a_{ij}\overline{x}_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(k)}$$
$$= \frac{a_{ii}}{\omega}(x_i^{(k)} - \widetilde{x}_i^{(k)}).$$

Since $\alpha = 0$ in (B.2) is feasible we have

$$E(x^{(k+1)}) \le E(\overline{x}^{(k+1)}) = E(\overline{x}^{(k,n)}) \le E(\overline{x}^{(k,n-1)}) \le \cdots \le E(\overline{x}^{(k,0)}) = E(x^{(k)}), \tag{B.5}$$

i.e. the sequence $\{E(x^{(k)})\}_k$ is decreasing monotonically and is bounded from below as A is positive definite. Hence, $\{E(x^{(k)})\}_k$ has a limit. Moreover, with (B.3) we get that

$$\|\overline{x}^{(k+1)} - x^{(k)}\|_2^2 = \sum_{i=1}^{n} \frac{2}{a_{ii}}(\frac{2}{\omega\eta_{i,k}} - 1)^{-1}\left(E(\overline{x}^{(k,i-1)}) - E(\overline{x}^{(k,i)})\right) \le \left(E(x^{(k)}) - E(x^{(k+1)})\right)\frac{2n}{\min_i a_{ii}}\frac{\omega}{2-\omega}$$

and since $x^{(k+1)}$ is the solution of (B.2) and $\overline{x}^{(k+1)}$ is a feasible point to that minimization problem we have

$$\|x^{(k+1)} - \overline{x}^{(k+1)}\|_2^2 \le \lambda_{\min}(A)^{-2}\|x^{(k+1)} - \overline{x}^{(k+1)}\|_A^2 \le 2\lambda_{\min}(A)^{-2}\left(E(\overline{x}^{(k+1)}) - E(x^{(k+1)})\right) \le 2\lambda_{\min}(A)^{-2}\left(E(x^{(k)}) - E(x^{(k+1)})\right).$$

Consequently we have with $\alpha_{i,k}^*$ the minimizer of (B.2)

$$\lim_{k\to\infty}\|x^{(k+1)} - x^{(k)}\|_2 \le \lim_{k\to\infty}\left(2\lambda_{\min}(A)^{-2} + \frac{2n}{\min_i a_{ii}}\frac{\omega}{2-\omega}\right)^{1/2}\left(E(x^{(k)}) - E(x^{(k+1)})\right)^{1/2} = 0, \tag{B.6}$$

$$\lim_{k\to\infty}\|x^{(k+1)} - \overline{x}^{(k+1)}\|_2^2 = \lim_{k\to\infty}\|\sum_{i=1}^{M}\alpha_{i,k}^* s_{i,k}\|_2^2 \le \lim_{k\to\infty} 2\lambda_{\min}(A)^{-2}\left(E(x^{(k)}) - E(x^{(k+1)})\right) = 0 \Rightarrow \lim_{k\to\infty}\alpha_{i,k}^* s_{i,k} = 0. \tag{B.7}$$

Since $E(\cdot)$ is strictly convex and $\{E(x^{(k)})\}_k$ is bounded, the sequence $\{x^{(k)}\}_k$ must be bounded. Thus $\{x^{(k)}\}_k$ has at least one accumulation point. Let x^* be such an accumulation point, i.e. $\lim_{l\to\infty} x^{(k_l)} = x^*$ for the subsequence k_l. By (B.1) and (B.2) we have that

$$x_i^{(k_l+1)} - x_i^{(k_l)} = \sum_{i=1}^{M}\alpha_{i,k_l}^* s_{i,k_l} + \min\left\{(1-\omega)x_i^{(k_l)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1}a_{ij}\overline{x}_j^{(k_l+1)} - \sum_{j=i+1}^{n}a_{ij}x_j^{(k_l)}\right), g_i\right\} - x_i^{(k_l)}$$
$$= \sum_{i=1}^{M}\alpha_{i,k_l}^* s_{i,k_l} + \min\left\{\frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1}a_{ij}\overline{x}_j^{(k_l+1)} - \sum_{j=i}^{n}a_{ij}x_j^{(k_l)}\right), g_i - x_i^{(k_l)}\right\}.$$

Hence,

$$0 = \lim_{l\to\infty}(x_i^{(k_l+1)} - x_i^{(k_l)}) = \lim_{l\to\infty}\sum_{i=1}^{M}\alpha_{i,k_l}^* s_{i,k_l} + \min\left\{\frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1}a_{ij}\overline{x}_j^{(k_l+1)} - \sum_{j=i}^{n}a_{ij}x_j^{(k_l)}\right), g_i - x_i^{(k_l)}\right\}$$
$$= \min\left\{\frac{\omega}{a_{ii}}(b - Ax^*)_i, g_i - x_i^*\right\} \quad (1 \le i \le n),$$

i.e. all accumulation points solve (8). Since (8) has exactly one solution as A is positive definite, we obtain $\lim_{k\to\infty} x^{(k)} = x$ which completes the proof.

Acknowledgement

The authors Gregor Milicic and Andreas Schröder gratefully acknowledge support from the Sparkling Science research program of the Federal Ministry of Science, Research and Economy of Austria (BMWFW) within the Sparkling Science project 'EMMA – Experimentation with mathematical algorithms' (SPA 05/172). All authors wish to thank the high-school students of the HTL Braunau[1] participating in the EMMA-Project for assisting with the simulations, in particular during the workshop *EMMA in the Mathematics Village, Sirinçe, Turkey*.

[1] L. Banz, E.P. Stephan, A posteriori error estimates of hp-adaptive IPDG-FEM for elliptic obstacle problems, Appl. Numer. Math. 76 (2014) 76–92.
[2] L. Banz, E.P. Stephan, hp-adaptive IPDG/TDG-FEM for parabolic obstacle problems, Comput. Math. Appl. 67 (2014) 712–731.
[3] A. Bnouhachem, A new projection and contraction method for linear variational inequalities, J. Math. Anal. Appl. 314 (2006) 513–525.
[4] F.E. Curtis, Z. Han, D.P. Robinson, A globally convergent primal-dual active-set framework for large-scale convex quadratic optimization, Comput. Optim. Appl. 60 (2014) 311–341.
[5] C. Geiger, C. Kanzow, Theorie und Numerik restringierter Optimierungsaufgaben, Springer, 2002.
[6] P.E. Gill, W. Murray, M.H. Wright, Practical optimization, Academic Press, 1981.
[7] R. Glowinski, Numerical methods for nonlinear variational problems, Springer Series in Computational Physics, Springer-Verlag, New York, 1984.
[8] R. Glowinski, J.L. Lions, R. Trémolierès, Numerical analysis of variational inequalities, Studies in Mathematics and its Applications, North-Holland Publishing Company, Amsterdam, 1981.
[9] B. He, A class of projection and contraction methods for monotone variational inequalities, Appl. Math. Optim. 35 (1997) 69–76.
[10] I. Hlávaček, J. Haslinger, J. Nečas, J. Lovíšek, Solution of variational inequalities in mechanics, Applied Mathematical Sciences, Springer-Verlag, New York, 1988.
[11] F. Jarre, J. Stoer, Optimierung, Springer-Verlag, 2013.
[12] N. Kikuchi, J. Oden, Contact problems in elasticity: A study of variational inequalities and finite element methods, SIAM Studies in Applied Mathematics, SIAM, Philadelphia, 1988.
[13] K. Kunisch, F. Rendl, An infeasible active set method for quadratic problems with simple bounds, SIAM J. Optim. 14 (2003) 35–52.
[14] M. Maischak, E.P. Stephan, Adaptive hp-versions of BEM for Signorini problems, Appl. Numer. Math. 54 (2005) 425–449.
[15] J. Nocedal, S.J. Wright, Numerical Optimization, Springer, 2 edition, 1999.

[1] `http://www.htl-braunau.at/htl-live.html`