Numerical solution of the two-yield elastoplastic minimization problem

Andreas Hofinger, Jan Valdman*

Abstract

This paper concentrates on fast calculation techniques for the two-yield elastoplastic problem, a locally defined, convex but non-smooth minimization problem for unknown plastic-strain increment matrices P_1 and P_2. So far, the only applied technique was an alternating minimization, whose convergence is known to be geometrical and global. We show that symmetries can be utilized to obtain a more efficient implementation of the alternating minimization. For the first plastic time-step problem, which describes the initial elasto-plastic transition, the exact solution for P_1 and P_2 can even be obtained analytically. In the later time-steps used for the computation of the further development of elastoplastic zones in a continuum, an extrapolation technique as well as a Newton-algorithm are proposed. Finally, we present a realistic example for the first plastic time-step, where the new techniques decrease the computation time by a factor of 10.

1 Introduction

In the following we briefly present the equations that appear in the two-yield plasticity problem. An elaborate discussion of the model is given in [BCV04, BCV05].

The time-dependent deformation process of the the two-yield elastoplastic continuum (which is defined by an open bounded domain Ω in $d = 1, 2, 3$ dimensions) is described by a displacement field u and two plastic strain fields p_1, p_2. These fields are considered space- and time-dependent,

$$u = u(x, t), \quad p_1 = p_1(x, t), \quad p_2 = p_2(x, t)$$

with the space parameters $x \in \Omega$ and the time parameter $t \in [0, T]$. The displacement $u(x, t)$ is represented pointwise as a vector in $\mathbb{R}^d$; the plas-

*Special Research Program SFB F013 'Numerical and Symbolic Scientific Computing', Johannes Kepler University Linz, Altenbergerstrasse 69, A-4040 Linz, Austria, email: Jan.Valdman@sfb013.uni-linz.ac.at

tic strains $p_1(x,t)$, $p_2(x,t)$ as symmetric and trace-free matrices in $\mathbb{R}^{d\times d}_{\mathrm{sym}}$, i. e., $\operatorname{tr} p_1(x,t) = \operatorname{tr} p_2(x,t) = 0$. The trace operator tr is defined as $\operatorname{tr} A = \sum_{i=1}^{d} A_{ii}$ for a matrix $A \in \mathbb{R}^{d\times d}$. Furthermore, $\operatorname{dev} : \mathbb{R}^{d\times d} \to \mathbb{R}^{d\times d}$ defines the *deviatoric operator* which transforms a matrix A into a trace-free form via $\operatorname{dev} A = A - \frac{1}{d}(\operatorname{tr} A)\,\mathbb{I}$ (here $\mathbb{I}$ denotes the identity matrix in d dimensions). Then, the conditions on the pointwise values of the plastic strains read

$$p_1(x,t),\;\; p_2(x,t) \in \operatorname{dev} \mathbb{R}^{d\times d}_{\mathrm{sym}}.$$

Additional mechanical fields, such as the (linearized) deformation field ε and the stress field σ can be calculated pointwise as

$$\varepsilon(x,t) = \frac{1}{2}\left((\nabla_x u(x,t))^T + \nabla_x u(x,t)\right), \tag{1}$$

$$\sigma(x,t) = \mathbb{C}(\varepsilon(x,t) - p_1(x,t) - p_2(x,t)), \tag{2}$$

using an elasticity matrix $\mathbb{C}$ from the isotropic case, defined by $\mathbb{C}\varepsilon(x,t) = 2\mu\varepsilon(x,t) + \lambda(\operatorname{tr}\varepsilon(x,t))\mathbb{I}$, for the (positive) Lamé coefficients μ and λ.

After discretization in time, using the values $0 = t_0 < t_1 < \cdots < t_{n-1} < t_n = T$, we define the *plastic strain increment fields*

$$P_1(x,t_i) := p_1(x,t_i) - p_1(x,t_{i-1}), \quad P_2(x,t_i) := p_2(x,t_i) - p_2(x,t_{i-1}).$$

Since the elastoplastic continuum is expected undeformed at time $t = 0$, we have initial conditions $u(x,0) = 0$ and $p_1(x,0) = p_2(x,0) = 0$. Until the stresses in the continuum (which are caused by external deforming forces) exceed plasticity limits in the discrete time $t_p, p \in 1\ldots n$, there are only elastic deformations indicated by conditions $p_1(x,t_i) = p_2(x,t_i) = 0, i < p$ observed. The discrete time t_p is denoted as *the first plastic time-step.* After t_p is reached, the plastic strains in *the later time-steps* $t_i > t_p$ remain permanently nonzero even though the external deforming forces might vanish. This behaviour occurs due to the hysteresis effect in plasticity [BS96].

The plastic strain increment fields are collected in a generalized plastic strain increment field

$$P(x,t_i) := (P_1(x,t_i), P_2(x,t_i))^T,$$

for all $x \in \Omega, i = 1\ldots n$. It is shown in [BCV05], that the value $P(x,t_i)$ satisfies

$$\left\{\hat{A}(x,t_i) - (\hat{\mathbb{C}} + \hat{\mathbb{H}})P(x,t_i)\right\} : (Q - P(x,t_i)) \le |Q|_{\sigma^y} - |P(x,t_i)|_{\sigma^y} \tag{3}$$

for arbitrary $Q = (Q_1, Q_2)^T$, $Q_1, Q_2 \in \operatorname{dev} \mathbb{R}^{d\times d}_{\mathrm{sym}}$. The generalized elasticity matrix $\hat{\mathbb{C}}$ and the generalized hardening matrix $\hat{\mathbb{H}}$ read

$$\hat{\mathbb{C}} := \begin{pmatrix} \mathbb{C} & \mathbb{C} \\ \mathbb{C} & \mathbb{C} \end{pmatrix} \quad \text{and} \quad \hat{\mathbb{H}} := \begin{pmatrix} \mathbb{H}_1 & 0 \\ 0 & \mathbb{H}_2 \end{pmatrix}, \tag{4}$$

where $\mathbb{H}_1 = h_1\mathbb{I}$, $\mathbb{H}_2 = h_2\mathbb{I}$ denote hardening matrices; h_1, $h_2 > 0$ are hardening coefficients. The generalized loading $\hat{A}$ reads

$$\hat{A}(x,t_i) := \begin{pmatrix} A_1(x,t_i) \\ A_2(x,t_i) \end{pmatrix} = \begin{pmatrix} \mathbb{C}\varepsilon(x,t_i) \\ \mathbb{C}\varepsilon(x,t_i) \end{pmatrix} - (\hat{\mathbb{C}} + \hat{\mathbb{H}}) \begin{pmatrix} p_1(x,t_{i-1}) \\ p_2(x,t_{i-1}) \end{pmatrix}. \quad (5)$$

The matrix norm $|\cdot|_{\sigma^y}$ is defined as $|Q|_{\sigma^y} := \sigma_1^y|Q_1| + \sigma_2^y|Q_2|$, where $|\cdot|$ denotes the Frobenius norm. Due to the modeling of two-yield plasticity (cf. [BCV04]), we assume

$$0 < \sigma_1^y \le \sigma_2^y. \quad (6)$$

The importance of the matrix inequality (3) lies in the postprocessing of the plastic strain fields from a displacement field. Once an approximation of the displacement field $u(x,t_i)$ in the discrete time t_i is provided (e.g. from various nonlinear methods [KV03]), one can compute the deformation field $\varepsilon(x,t_i)$ and the generalizes loading $\hat{A}(x,t_i)$ from (1) and (5) under the knowledge of $p_1(x,t_{i-1})$, $p_2(x,t_{i-1})$ in the previous discrete time t_{i-1}. Then the values of the plastic strain fields $p_1(x,t_i)$, $p_2(x,t_i)$ follow easily from the upgrades $p_1(x,t_i) = P_1(x,t_i) + p_1(x,t_{i-1})$, $p_2(x,t_i) = P_2(x,t_i) + p_2(x,t_{i-1})$, where $P_1(x,t_i)$, $P_2(x,t_i)$ solves the matrix inequality (3).

For a given discrete time t_i, $i \in 1 \ldots n$ and a space point $x \in \Omega$, the dependence on (x,t_i) may be dropped and the inequality (3) is simplified as

$$\{\hat{A} - (\hat{\mathbb{C}} + \hat{\mathbb{H}})P\} : (Q - P) \le |Q|_{\sigma^y} - |P|_{\sigma^y}, \quad (7)$$

which must hold for all $Q = (Q_1, Q_2)^T, Q_1, Q_2 \in \operatorname{dev}\mathbb{R}^{d\times d}_{\mathrm{sym}}$.

Depending on the values $P_1(x,t_i) \in \operatorname{dev}\mathbb{R}^{d\times d}_{\mathrm{sym}}$ and $P_2(x,t_i) \in \operatorname{dev}\mathbb{R}^{d\times d}_{\mathrm{sym}}$, there are four interesting cases:

(i). $P_1 = P_2 = 0$ (the elastic upgrade)

(ii). $P_1 \ne 0$, $P_2 = 0$ (the first plastic upgrade).

(iii). $P_1 = 0$, $P_2 \ne 0$ (the second plastic upgrade).

(iv). $P_1 \ne 0$, $P_2 \ne 0$ (the first and the second plastic upgrades).

The classification into the four cases allows for the identification of an elastoplastic interface, whose shape plays a crucial role in the development of highly efficient methods in computational elastoplasticity, e.g., [NDR05]. Therefore, apart from the the exact (or approximated) values of P_1 and P_2, we are also interested in the above classification.

There is an equivalent way of expressing (7) using the concept of a subdifferential from convex analysis (e. g., [ET99]). It is well known that $P^* =$

$(P_1^*, P_2^*)^T, P_1^*, P_2^* \in \mathbb{R}^{d\times d}_{\mathrm{sym}}$ belongs to a subdifferential $\partial|\cdot|_{\sigma^y}(P)$ of the convex function $|\cdot|_{\sigma^y}$ at the the matrix $P = (P_1, P_2)^T, P_1, P_2 \in \mathbb{R}^{d\times d}_{\mathrm{sym}}$ iff

$$P^* : (Q - P) \le (|Q|_{\sigma^y} - |P|_{\sigma^y})$$

for all $Q = (Q_1, Q_2)^T, Q_1, Q_2 \in \mathbb{R}^{d\times d}_{\mathrm{sym}}$. By comparison with (7) it is easy to check an inclusion $\{\hat{A} - (\hat{\mathbb{C}} + \hat{\mathbb{H}})P\} \in \partial|\cdot|_{\sigma^y}(P)$. Note that for trace-free arguments Q_i, P_i from (7), the equalities

$$A_i : (Q_i - P_i) = \operatorname{dev} A_i : (Q_i - P_i) \quad \text{and} \quad (\mathbb{C} + \mathbb{H}_i)P_i = (2\mu + h_i)P_i$$

hold for $i = 1, 2$. Thus we reformulate the inequality (7) as an inclusion

$$\begin{pmatrix} \operatorname{dev} A_1 \\ \operatorname{dev} A_2 \end{pmatrix} - \begin{pmatrix} (2\mu + h_1)\mathbb{I} & 2\mu\mathbb{I} \\ 2\mu\mathbb{I} & (2\mu + h_2)\mathbb{I} \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} \in (\partial|\cdot|_{\sigma^y}(P_1, P_2))^T. \quad (8)$$

In addition to the characterization of P using the inequality (7) or the inclusion (8), there is also an equivalent minimization problem.

Lemma 1 ([BCV05]). *For a given $\hat{A} = (A_1, A_2)^T, A_1, A_2 \in \mathbb{R}^{d\times d}_{sym}$, there exists a unique $P = (P_1, P_2)^T$, $P_1, P_2 \in \operatorname{dev} \mathbb{R}^{d\times d}_{sym}$ that satisfies the inequality (7) for all $Q = (Q_1, Q_2)^T, Q_1, Q_2 \in \operatorname{dev} \mathbb{R}^{d\times d}_{sym}$. This P is characterized as the minimizer of*

$$f(Q) = \frac{1}{2}(\hat{\mathbb{C}} + \hat{\mathbb{H}})Q : Q - \hat{A} : Q + |Q|_{\sigma^y} \quad (9)$$

(amongst trace-free symmetric $d \times d$ matrices Q_1, Q_2).

2 Two-yield elastoplastic problem

After these preparations we now define the two-yield minimization problem.

Problem 1 (Two-yield minimization problem). *For given positive material parameters μ, h_1, h_2, σ_1^y, σ_2^y and trace-free matrices* $\operatorname{dev} A_1$, $\operatorname{dev} A_2 \in \mathbb{R}^{d\times d}_{sym}$ *find $P = (P_1, P_2)^T$, $P_1, P_2 \in \operatorname{dev} \mathbb{R}^{d\times d}_{sym}$ that minimizes (9) (amongst trace-free symmetric $d \times d$ matrices Q_1, Q_2).*

In the next section we present a globally convergent method for solving Problem 1; the aim of this paper is to speed up the presented algorithm.

2.1 Iterative method

As noticed in [BCV05], no analytical solution of Problem 1 seems to exist. With the help of the operator

$$\mathcal{F}(M, \sigma, h) := \frac{(|M| - \sigma)_+}{2\mu + h} \frac{M}{|M|}, \quad (10)$$

where $(\cdot)_+ := \max\{0, \cdot\}$, an iterative algorithm can be formulated that provides the exact solution of Problem 1 within arbitrary small given tolerance.
Algorithm 1 (Iterative calculation of P_1, P_2)**.** *Input* μ*,* h_1*,* h_2*,* σ_1^y*,* σ_2^y*,* $\operatorname{dev} A_1$*,* $\operatorname{dev} A_2$ *and* $\mathrm{tol} \geq 0$*.*

(i). Set $i := 0$ *and set the initial approximation*

$$P_1^i = P_2^i = 0.$$

(ii). Update P_2^i *via*

$$P_2^{i+1} = \mathcal{F}(\operatorname{dev} A_2 - 2\mu P_1^i, \sigma_2^y, h_2).$$

(iii). Update P_1^i *via*

$$P_1^{i+1} = \mathcal{F}(\operatorname{dev} A_1 - 2\mu P_2^{i+1}, \sigma_1^y, h_1)$$

(iv). If the desired accuracy is reached, i. e., if

$$|P_1^{i+1} - P_1^i| + |P_2^{i+1} - P_2^i| \leq \mathrm{tol} \cdot (|P_1^{i+1}| + |P_1^i| + |P_2^{i+1}| + |P_2^i|)$$

then output solution $(P_1, P_2) = (P_1^{i+1}, P_2^{i+1})$*. Otherwise, set* $i := i + 1$ *and go to step (ii).*

The value of $\mathcal{F}(\cdot)$ in steps (ii) and (iii) in the case $M = 0$ is defined to be the zero matrix (this follows from assumption (6) by continuous extension).

In [BCV05, Proposition 1] it was shown that Algorithm 1 converges as a geometrical sequence, i. e., for arbitrary choice of initial approximation matrices P_1^0 and P_2^0, the distance to the fixed point P_1, P_2 behaves as

$$\left\|P_1^i - P_1\right\|^2 + \left\|P_2^i - P_2\right\|^2 \leq C q^i . \tag{11}$$

The constant C depends on the quality of the initial guess; the factor q is always strictly smaller than 1 and depends on material parameters and the domain Ω only.

2.2 Acceleration of the iterative method

Since Algorithm 1 is linearly convergent the *Aitken extrapolation method* can be applied to improve the convergence behavior [Atk89, Sto64]. This idea is similar to the extrapolation method used in Romberg-integration.

To perform the extrapolation, Algorithm 1 is applied 2 times. In a third step, only the matrix P_2^3 is computed. From the available approximations P_2^1, P_2^2 and P_2^3, an extrapolated matrix P_2^{extr} can be obtained as described below. This matrix is then used in step (iii) of Algorithm 1 to obtain a corresponding approximation of P_1.

Algorithm 2 (Iterative calculation of P_1, P_2 combined with Aitken acceleration)**.** *Input* $\mu, h_1, h_2, \sigma_1^y, \sigma_2^y, \operatorname{dev} A_1$*,* $\operatorname{dev} A_2$ *and* $\mathrm{tol} \geq 0$*.*
(i) - (iv) as in Algorithm 1.

(v). If $i = 3$ *then set for all* $s, t = 1, \ldots, d$

$$p_1 := \left[P_2^1\right]_{st}, \; p_2 := \left[P_2^2\right]_{st}, \; p_3 := \left[P_2^3\right]_{st}$$

and calculate the extrapolated matrix component-wise by

$$\left[P_2^{extr}\right]_{st} = \begin{cases} p_1 - \frac{(p_2 - p_1)^2}{p_3 - 2p_2 + p_1} & \text{if } p_3 - 2p_2 + p_1 \neq 0, \\ 0 & \text{if } p_3 - 2p_2 + p_1 = 0. \end{cases}$$

(vi). Update P_1^2 *via*

$$P_1^3 = \mathcal{F}(\operatorname{dev} A_1 - 2\mu P_2^{extr}, \sigma_1^y, h_1)$$

(vii). Set $i := 0$ *and go to (ii).*

In all tested examples the above algorithm speeds up the iteration significantly, and gives a sufficiently accurate solution after one extrapolation (i. e., after 3 iterations of the original algorithm).

To obtain quantitative results on the speedup obtained with Aitken extrapolation, a series expansion of the error behavior is necessary (and not only an upper bound as in (11)). Since the problem under consideration is non-differentiable, it seems not possible to derive such expansions for the error. A precise analysis of the convergence acceleration is therefore an open question.

2.3 Structure and classification of solutions

In the following we take a closer look at the structure of the solutions. In particular in the 3-dimensional case, the results can be used as a simple way to speed up the algorithm, since using the structure of the solutions P_1 and P_2, much less parameters need to be stored and manipulated during the computations. Lemma 2 shows the structure of the solutions, Lemma 3 gives information about the localization of the appearing parameters.
Lemma 2 (solution structure)**.** *The solutions* P_1, P_2 *of Problem 1 can be expressed as linear combinations of the matrices* $\operatorname{dev} A_1, \operatorname{dev} A_2$*, i. e.,*

$$\begin{aligned} P_1 &= c_{11} \operatorname{dev} A_1 + c_{12} \operatorname{dev} A_2, \\ P_2 &= c_{21} \operatorname{dev} A_1 + c_{22} \operatorname{dev} A_2. \end{aligned} \tag{12}$$

Furthermore, it holds $c_{12} = c_{21}$*.*

Proof. Let the solution P_1, P_2 be calculated exactly by using Algorithm 1 with tolerance $\mathrm{tol} = 0$. Taking the iterative steps (ii) and (iii) in Algorithm 1, induction over the number of iteration steps i proves the linear dependence of the solution approximations

$$\begin{aligned} P_1^i &= c_{11}^i \operatorname{dev} A_1 + c_{12}^i \operatorname{dev} A_2, \\ P_2^i &= c_{21}^i \operatorname{dev} A_1 + c_{22}^i \operatorname{dev} A_2. \end{aligned} \tag{13}$$

It remains to show that c_{12}^i and c_{21}^i converge to the same value for increasing i. The condition of the fixed point of a mapping defined by (ii) and (iii) reads

$$\begin{aligned} P_1 &= f_1 \cdot (\operatorname{dev} A_1 - 2\mu P_2), \\ P_2 &= f_2 \cdot (\operatorname{dev} A_2 - 2\mu P_1), \end{aligned} \tag{14}$$

where f_1 and f_2 are scalar non-negative factors (cf. (10)). Via (13), this condition can be reformulated in terms of limit coefficients $c_{11}^i \to c_{11}$, $c_{12}^i \to c_{12}$, $c_{21}^i \to c_{21}$, $c_{22}^i \to c_{22}$ as

$$\begin{aligned} c_{11} \operatorname{dev} A_1 + c_{12} \operatorname{dev} A_2 &= f_1 \cdot ((1 - 2\mu c_{21}) \operatorname{dev} A_1 - 2\mu c_{22} \operatorname{dev} A_2), \\ c_{21} \operatorname{dev} A_1 + c_{22} \operatorname{dev} A_2 &= f_2 \cdot ((1 - 2\mu c_{12}) \operatorname{dev} A_2 - 2\mu c_{11} \operatorname{dev} A_1). \end{aligned} \tag{15}$$

If $\operatorname{dev} A_1$ and $\operatorname{dev} A_2$ are linearly dependent, then the choice $c_{12} = c_{21} = 0$ gives the decomposition (12). So suppose now that $\operatorname{dev} A_1$ and $\operatorname{dev} A_2$ are linearly independent. Then, (15) implies

$$\begin{pmatrix} c_{11} \\ c_{12} \end{pmatrix} = f_1 \cdot \begin{pmatrix} 1 - 2\mu c_{21} \\ -2\mu c_{22} \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} c_{22} \\ c_{21} \end{pmatrix} = f_2 \cdot \begin{pmatrix} 1 - 2\mu c_{12} \\ -2\mu c_{11} \end{pmatrix}. \tag{16}$$

Both, $1-2\mu c_{12}$ and $1-2\mu c_{21}$ must be nonzero. (If for instance $1-2\mu c_{12} = 0$, then $c_{22} = 0$, and consequently $c_{12} = 0$; a contradiction.) Therefore, we obtain from (16)

$$\frac{c_{12}}{1 - 2\mu c_{12}} = -2\mu f_1 f_2 = \frac{c_{21}}{1 - 2\mu c_{21}} \tag{17}$$

which yields $c_{12} = c_{21}$. □

The technique of the proof also shows that the decomposition (12) is unique for linearly independent matrices $\operatorname{dev} A_1$, $\operatorname{dev} A_2$. The material parameters μ, h_1 and h_2 provide immediate information about the localization of the parameters c_{ij} in Lemma 2.

Lemma 3 (Coefficient Localization)**.** *There exist non-negative real numbers f_1, f_2 with $0 \leq f_1 < \frac{1}{2\mu+h_1}$, $0 \leq f_2 < \frac{1}{2\mu+h_2}$ such that*

$$c_{11} = \frac{f_1}{1 - 4\mu^2 f_1 f_2}, \tag{18a}$$

$$c_{22} = \frac{f_2}{1 - 4\mu^2 f_1 f_2}, \tag{18b}$$

$$c_{12} = c_{21} = \frac{-2\mu f_1 f_2}{1 - 4\mu^2 f_1 f_2}. \tag{18c}$$

In particular, the coefficients satisfy

$$0 \leq c_{11} < \frac{2\mu + h_2}{2\mu(h_1 + h_2) + h_1 h_2},$$

$$0 \leq c_{22} < \frac{2\mu + h_1}{2\mu(h_1 + h_2) + h_1 h_2},$$

$$0 \geq c_{12} = c_{21} > \frac{-2\mu}{2\mu(h_1 + h_2) + h_1 h_2}.$$

Proof. The factors f_1 and f_2 are just those we introduced in (14) in the proof of Lemma 2 above. The lower and upper bounds on f_1 and f_2 follow immediately from (10) since

$$0 \leq \frac{1}{2\mu + h} \frac{(|M| - \sigma)_+}{|M|} < \frac{1}{2\mu + h},$$

for $|M| \in \mathbb{R}_0^+$. In order to derive the representation (18) we turn back to (17), which gives (18c). From (18c), the expressions for c_{11} and c_{22} can be directly deduced via (16). The lower and upper bounds on c_{11}, c_{12} and c_{22} follow from the corresponding bounds on f_1 and f_2. $\square$

In some cases one (or even both) of the matrices P_1 and P_2 are zero-matrices. The following lemma provides an analytic criterion to detect these situations.

Lemma 4 (solution classification)**.** *The following equivalences hold:*

$$P_1 = 0 \quad \Leftrightarrow \quad |\operatorname{dev} A_1 - 2\mu \ \mathcal{F}(\operatorname{dev} A_2, \sigma_2^y, h_2)| \leq \sigma_1^y, \tag{19}$$

$$P_2 = 0 \quad \Leftrightarrow \quad |\operatorname{dev} A_2 - 2\mu \ \mathcal{F}(\operatorname{dev} A_1, \sigma_1^y, h_1)| \leq \sigma_2^y. \tag{20}$$

Proof. Due to the symmetry of (19) and (20) it is sufficient to prove (19) only. Let P_1=0. In the following we use the knowledge that the alternating algorithm converges to the correct solution ([BCV05, Proposition 1]). According to step (ii) of Algorithm 1 in the limit case

$$P_2 = \mathcal{F}(\operatorname{dev} A_2, \sigma_2^y, h_2). \tag{21}$$

It follows from step (iii) of Algorithm 1 that $|\operatorname{dev} A_1 - 2\mu P_2| - \sigma_1^y)_+ = 0$ or equivalently

$$|\operatorname{dev} A_1 - 2\mu P_2| \le \sigma_1^y.$$

Substitution of (21) into the last inequality proves the first part of the equivalence.

Vice versa, suppose that the right hand side of (19) is valid, and let us consider the iteration sequence of Algorithm 1 with zero initial matrices $P_1^0 = P_2^0 = 0$. It is easy to see that the iterate P_2^1 is given exactly by (21). The assumption

$$|\operatorname{dev} A_1 - 2\mu \;\mathcal{F}(\operatorname{dev} A_2, \sigma_2^y, h_2)| \le \sigma_1^y$$

implies $P_1^1 = 0$. Thus, Algorithm 1 terminates with the solution $P_1 = 0$ and $P_2 = P_2^1$. □

Lemma 4 perfectly divides the analysis of Problem 1 into four cases in dependence of the values P_1 and P_2 as discussed on page 3.

(i). $P_1 = P_2 = 0$. Note that conditions (19), (20) are further simplified as

$$|\operatorname{dev} A_1| \le \sigma_1^y \quad \text{and} \quad |\operatorname{dev} A_2| \le \sigma_2^y. \tag{22}$$

(ii). $P_1 \neq 0$, $P_2 = 0$. Then following the proof of Lemma 4

$$P_1 = \;\mathcal{F}(\operatorname{dev} A_1, \sigma_1^y, h_1).$$

(iii). $P_1 = 0$, $P_2 \neq 0$. [1]). Then analogously

$$P_2 = \;\mathcal{F}(\operatorname{dev} A_2, \sigma_2^y, h_2).$$

(iv). $P_1 \neq 0$, $P_2 \neq 0$. For this case it was shown in [BCV05] that the following nonlinear system holds

$$\begin{pmatrix} \operatorname{dev} A_1 \\ \operatorname{dev} A_2 \end{pmatrix} - \begin{pmatrix} (2\mu + h_1)\mathbb{I} & 2\mu\mathbb{I} \\ 2\mu\mathbb{I} & (2\mu + h_2)\mathbb{I} \end{pmatrix} \begin{pmatrix} P_1 \\ P_2 \end{pmatrix} = \begin{pmatrix} \sigma_1^y \frac{P_1}{|P_1|} \\ \sigma_2^y \frac{P_2}{|P_2|} \end{pmatrix}. \tag{23}$$

Notice that only the case (iv) of both plastic increments leading to the nonlinear system (23) represents the difficulty in solving Problem 1 analytically.

[1] This situation can not happen in the first plastic time-step, cf. Subsection 3.1.

3 Analysis of the case $P_1 \neq 0, P_2 \neq 0$

Applying substitutions $P_i = \xi_i X_i$, where $|X_i| = 1, i = 1, 2$, (23) becomes a system of nonlinear equations for positive scalar parameters $\xi_1 = |P_1|$, $\xi_2 = |P_2|$, namely

$$\begin{pmatrix} \operatorname{dev} A_1 \\ \operatorname{dev} A_2 \end{pmatrix} = \begin{pmatrix} (\sigma_1^y + (2\mu + h_1)\xi_1)\mathbb{I} & 2\mu\xi_2\mathbb{I} \\ 2\mu\xi_1\mathbb{I} & (\sigma_2^y + (2\mu + h_2)\xi_2)\mathbb{I} \end{pmatrix} \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}. \quad (24)$$

In the following, we present analytical solutions of this system for a special case, as well as a numerical technique for a general setup.

3.1 Exact solution at the first plastic time-step

Now we turn to an important case for which the analytical solution of the nonlinear system (24) can be obtained analytically. Let us consider the first plastic time-step problem, i.e., the problem where the plastic strains $p_1(x, t_{p-1}) = P_2(x, t_{p-1}) = 0$ are zero in the time step t_{p-1} for a given space point $x \in \Omega$. Via (5) this yields

$$\operatorname{dev} A_1 = \operatorname{dev} A_2.$$

In this situation, the choice

$$X_1 = X_2 = \frac{\operatorname{dev} A_1}{|\operatorname{dev} A_1|}$$

transforms (24) into the system

$$\begin{aligned} &(|\operatorname{dev} A_1| - \sigma_2^y)\sigma_1^y + ((|\operatorname{dev} A_1| - \sigma_2^y)h_1 - 2\mu\sigma_2^y)\xi_1 \\ &\quad - \sigma_1^y(2\mu + h_2)\xi_2 - (h_1 h_2 + 2\mu(h_1 + h_2))\xi_1\xi_2 = 0 \\ &(|\operatorname{dev} A_1| - \sigma_1^y)\sigma_2^y + ((|\operatorname{dev} A_2| - \sigma_1^y)h_2 - 2\mu\sigma_1^y)\xi_2 \\ &\quad - \sigma_2^y(2\mu + h_1)\xi_1 - (h_1 h_2 + 2\mu(h_1 + h_2))\xi_1\xi_2 = 0. \end{aligned} \quad (25)$$

This system has two solutions. The first one is always negative, $(\xi_1, \xi_2) = (-\frac{\sigma_1}{h_1}, -\frac{\sigma_2}{h_2})$. The second one is given by

$$(\xi_1, \xi_2) = \frac{1}{2\mu(h_1 + h_2) + h_1 h_2} \begin{pmatrix} (|\operatorname{dev} A_1| - \sigma_1^y)h_2 + 2\mu(\sigma_2^y - \sigma_1^y) \\ (|\operatorname{dev} A_1| - \sigma_2^y)h_1 - 2\mu(\sigma_2^y - \sigma_1^y) \end{pmatrix}^t. \quad (26)$$

Moreover, when $\operatorname{dev} A_1 = \operatorname{dev} A_2$, the solution classification from Lemma 4 can be further simplified, and it is possible to prove e. g., that the situation $P_1 = 0$, $P_2 \neq 0$ can never appear. Summarizing, we obtain the following algorithm.

Algorithm 3 (Exact calculation of P_1, P_2 in the first plastic time-step). *Input* μ, h_1, h_2, $\sigma_1^y \leq \sigma_2^y$, $\operatorname{dev} A_1$.

(i). If $|\operatorname{dev} A_1| \leq \sigma_1^y$ *then output solution*

$$(P_1, P_2) = (0, 0).$$

(ii). If $|\operatorname{dev} A_1| \leq \sigma_2^y + \frac{2\mu}{h_1}(\sigma_2^y - \sigma_1^y)$ *then output solution*

$$(P_1, P_2) = (\mathcal{F}(A_1, \sigma_1^y, h_1), 0).$$

(iii). Output solution

$$(P_1, P_2) = (\xi_1, \xi_2)\frac{\operatorname{dev} A_1}{|\operatorname{dev} A_1|},$$

where (ξ_1, ξ_2) *is given by (26).*

The advantage of Algorithm 3 over Algorithm 1 becomes obvious in the comparison of the algorithms speed later.

In the next section we consider the general case, i. e., $\operatorname{dev} A_1 \neq \operatorname{dev} A_2$.

3.2 Reduction to the polynomial system

Let us now turn to the general case of later time-steps. By elimination of X_1, X_2 in (24), one obtains the system of nonlinear equations [BCV05],

$$|l_i(\xi_i)| - r(\xi_1, \xi_2) = 0 \quad \text{for } i = 1, 2, \tag{27}$$

where

$$\begin{aligned} l_1(\xi_1) &= (\sigma_1^y + (2\mu + h_1)\xi_1) \operatorname{dev} A_2 - 2\mu\xi_1 \operatorname{dev} A_1, \\ l_2(\xi_2) &= (\sigma_2^y + (2\mu + h_2)\xi_2) \operatorname{dev} A_1 - 2\mu\xi_2 \operatorname{dev} A_2, \\ r(\xi_1, \xi_2) &= (\sigma_1^y + (2\mu + h_2)\xi_1)(\sigma_2^y + (2\mu + h_2)\xi_2) - 4\mu^2\xi_1\xi_2. \end{aligned} \tag{28}$$

Instead of handling (27) we prefer to solve a squared system

$$\Phi_i(\xi_1, \xi_2) := |l_i(\xi_i)|^2 - (r(\xi_1, \xi_2))^2 = 0 \quad \text{for } i = 1, 2. \tag{29}$$

In terms of ξ_1, ξ_2, this gives two polynomials of second degree [BCV05]

$$\begin{aligned} \Phi_1(\xi_1, \xi_2) =& A + B\xi_1 + C\xi_1^2 - (G + H\xi_1 + I\xi_2 + J\xi_1\xi_2)^2 = 0, \\ \Phi_2(\xi_1, \xi_2) =& D + E\xi_2 + F\xi_2^2 - (G + H\xi_1 + I\xi_2 + J\xi_1\xi_2)^2 = 0, \end{aligned} \tag{30}$$

where the scalar parameters A, B, ..., J read

$$
\begin{aligned}
A &:= |\sigma_1^y \operatorname{dev} A_2|^2, \\
D &:= |\sigma_2^y \operatorname{dev} A_1|^2, \\
B &:= 2\sigma_1^y \operatorname{dev} A_2 : ((2\mu + h_1) \operatorname{dev} A_2 - 2\mu \operatorname{dev} A_1), \\
E &:= 2\sigma_2^y \operatorname{dev} A_1 : ((2\mu + h_2) \operatorname{dev} A_1 - 2\mu \operatorname{dev} A_2), \\
C &:= |(2\mu + h_1) \operatorname{dev} A_2 - 2\mu \operatorname{dev} A_1|^2, \\
F &:= |(2\mu + h_2) \operatorname{dev} A_1 - 2\mu \operatorname{dev} A_2|^2, \\
G &:= \sigma_1^y \sigma_2^y > 0, \\
H &:= \sigma_2^y (2\mu + h_1) > 0, \\
I &:= \sigma_1^y (2\mu + h_2) > 0, \\
J &:= 2\mu(h_1 + h_2) + h_1 h_2 > 0.
\end{aligned}
\tag{31}
$$

Note that the Cauchy-Schwarz inequality provides the estimates

$$
B^2 - 4CA \leq 0, \qquad E^2 - 4FD \leq 0. \tag{32}
$$

Expressing ξ_1 from the second equation in (30) and a substitution into the first equation using MAPLE 5 leads to an eighth-degree polynomial in ξ_2 only, see Lemma 5 in [BCV05].

3.3 Numerical approach to solve the polynomial system (30)

Observe that eight of ten parameters in (30) are positive, and that the solution (ξ_1, ξ_2) we are looking for has to be positive as well. This enables us to construct a fast numerical method to solve (30). Therefore we introduce an auxiliary parameter t and consider the system

$$
|l_1(\xi_1)|^2 = t^2, \tag{33a}
$$
$$
|l_2(\xi_2)|^2 = t^2, \tag{33b}
$$
$$
r(\xi_1, \xi_2) = t. \tag{33c}
$$

Note that it is sufficient to consider a linear equation for t in (33c), since r in (28) is always positive. Inserting the notations as introduced in (31) we obtain the following system of equations

$$
A + B\xi_1 + C\xi_1^2 = t^2, \tag{34a}
$$
$$
D + E\xi_2 + F\xi_2^2 = t^2, \tag{34b}
$$
$$
G + H\xi_1 + I\xi_2 + J\xi_1\xi_2 = t. \tag{34c}
$$

For given t we can immediately solve (34a) and (34b) for ξ_1 and ξ_2 respectively. Although there are in principle two solutions in each case, only one of them can be positive. Given these values of ξ_1 and ξ_2 we can interpret (34c) as an equation for t and solve this equation via a (one-dimensional) Newton's method.

Let $\varphi(t)$ be defined as

$$\varphi(t) := G + H\xi_1(t) + I\xi_2(t) + J\xi_1(t)\xi_2(t) - t\,.$$

The Newton-Iteration for solving $\varphi(t) = 0$ is then given as

$$t_{i+1} = t_i - \frac{\varphi(t_i)}{\frac{\partial\varphi}{\partial t} + \frac{\partial\varphi}{\partial\xi_1}\frac{\partial\xi_1}{\partial t} + \frac{\partial\varphi}{\partial\xi_2}\frac{\partial\xi_2}{\partial t}}$$

To compute the partial derivatives of ξ_1 and ξ_2 with respect to t we use the implicit function theorem, and obtain for instance via (34a)

$$\frac{\partial\xi_1}{\partial t} = \frac{2t}{2C\xi_1 + B}\,.$$

The Newton algorithm for solving (27) can now be given as follows.
Algorithm 4 (Newton Iteration for ξ_1, ξ_2)**.** *Input A, B,..., J and an initial approximation t_0.*

(i). Set $i := 0$.

(ii). Calculate

$$\begin{aligned}
\xi_1 &= -\frac{B}{2C} + \frac{1}{2C}\sqrt{B^2 - 4C(A - t_i^2)}, \\
\xi_2 &= -\frac{E}{2F} + \frac{1}{2F}\sqrt{E^2 - 4F(D - t_i^2)}, \\
t_{i+1} &= t_i - \frac{G + H\xi_1 + I\xi_2 + J\xi_1\xi_2 - t_i}{2t_i\left(\frac{I+J\xi_2}{2C\xi_1+B} + \frac{H+J\xi_1}{2F\xi_2+E}\right) - 1}.
\end{aligned}$$

(iii). If convergence then output (ξ_1, ξ_2); otherwise $i := i + 1$ and go to (ii).
Remark 1 (Starting value)**.** Observe that setting $t_0 = 0$ would yield imaginary values for ξ_1 and ξ_2 in the iteration (cf. (32)). To avoid this situation we must start with t_0 sufficiently large. The case $\xi_1 = \xi_2 = 0$ would give $t = \sqrt{A}$, $t = \sqrt{D}$ and $t = G$ from (34a), (34b) and (34c) respectively. A suitable initial guess t_0 is now for instance

$$t_0 = 2\max\{\sqrt{A}, \sqrt{D}, G\}. \tag{35}$$

This choice also ensures $\xi_1 \geq 0$ and $\xi_2 \geq 0$.

In the following, we test the new algorithms on two examples.

4 Example I

Here we consider the first plastic time-step problem, i. e., Algorithm 3 is applicable.

This example is taken from [BCV05]. Let us assume material parameters

$$\mu = 1, \sigma_1^y = 1, \sigma_2^y = 2, h_1 = 1, h_2 = 1 \tag{36}$$

and loading matrices

$$\operatorname{dev} A_1 = \operatorname{dev} A_2 = \begin{pmatrix} 10 & 0 \\ 0 & -10 \end{pmatrix}.$$

4.1 Results for Algorithm 1 and Algorithm 2

The original Algorithm 1 generates the iteration sequence

$$\begin{aligned}
P_1^1 &\approx \begin{pmatrix} 1.1897 & 0 \\ 0 & -1.1897 \end{pmatrix}, & P_2^1 &\approx \begin{pmatrix} 2.8619 & 0 \\ 0 & -2.8619 \end{pmatrix}, \\
P_1^2 &\approx \begin{pmatrix} 1.7184 & 0 \\ 0 & -1.7184 \end{pmatrix}, & P_2^2 &\approx \begin{pmatrix} 2.0688 & 0 \\ 0 & -2.0688 \end{pmatrix}, \\
P_1^3 &\approx \begin{pmatrix} 1.9534 & 0 \\ 0 & -1.9534 \end{pmatrix}, & P_2^3 &\approx \begin{pmatrix} 1.7163 & 0 \\ 0 & -1.7163 \end{pmatrix}, \\
P_1^4 &\approx \begin{pmatrix} 2.0579 & 0 \\ 0 & -2.0579 \end{pmatrix}, & P_2^4 &\approx \begin{pmatrix} 1.5596 & 0 \\ 0 & -1.5596 \end{pmatrix}, \\
P_1^5 &\approx \begin{pmatrix} 2.1043 & 0 \\ 0 & -2.1043 \end{pmatrix}, & P_2^5 &\approx \begin{pmatrix} 1.4900 & 0 \\ 0 & -1.4900 \end{pmatrix}, \\
P_1^6 &\approx \begin{pmatrix} 2.1249 & 0 \\ 0 & -2.1249 \end{pmatrix}, & P_2^6 &\approx \begin{pmatrix} 1.4591 & 0 \\ 0 & -1.4591 \end{pmatrix},
\end{aligned}$$

and terminates for given tolerance $\mathrm{tol} = 10^{-12}$ after 18 iterations with the approximation

$$P_1^{18} \approx \begin{pmatrix} 2.14142 & 0 \\ 0 & -2.14142 \end{pmatrix}, \qquad P_2^{18} \approx \begin{pmatrix} 1.43431 & 0 \\ 0 & -1.43431 \end{pmatrix}.$$

Note that the norms of both matrices are

$$(|P_1^{18}|, |P_2^{18}|) \approx (3.02842573820811, 2.02842920455331).$$

The Aitken acceleration in Algorithm 2 speeds up this iteration significantly. The matrix P_2^{extr} extrapolated from the matrices P_2^1, P_2^2, P_2^3 is identical to P_2^{18} up to the first five digits.

i	t_i	ξ_1 approximation	ξ_2 approximation
0	56.5	3.00	2.00
1	56.973	3.0286	2.0286
2	56.970563	3.02842714	2.02842714
3	56.970562748477136	3.028427124746190	2.028427124746190

Table 1: Quadratic convergence of the iterates generated by Algorithm 4 in Example I. The initial guess is chosen as in (35), after 3 iterations the result is exact up to machine accuracy.

For comparison, iterations of Algorithms 1 and 2 are displayed in Figure 1 simultaneously.

4.2 Results for Algorithm 3 and Algorithm 4

Via Lemma 4, it is easy to check that the solution of Problem 1 satisfies $P_1 \neq 0, P_2 \neq 0$. Thus, the matrix norms $\xi_1 = |P_1|$ and $\xi_2 = |P_2|$ fulfill the polynomial system of equations (29), i.e.,

$$\begin{aligned} 200 + 400\,\xi_1 + 200\,\xi_1^2 - (2 + 6\,\xi_1 + 3\,\xi_2 + 5\,\xi_1\,\xi_2)^2 &= 0, \\ 800 + 800\,\xi_2 + 200\,\xi_2^2 - (2 + 6\,\xi_1 + 3\,\xi_2 + 5\,\xi_1\,\xi_2)^2 &= 0. \end{aligned} \tag{37}$$

Direct computation [BCV05] provides one positive solution

$$(\xi_1, \xi_2) = (2\sqrt{2} + \frac{1}{5}, -\frac{4}{5} + 2\sqrt{2}) \approx (3.028427124, 2.028427124). \tag{38}$$

Noticing that this is one-time step example, the same solution can be immediately obtained by (26) as a part of Algorithm 3.

Algorithm 4 solves the system (37) iteratively. With the initial guess (35), the method converges quickly to the correct solution. Tables 1 and 2 show the behavior of the approximations generated by Algorithm 4 above. As can be seen in Table 1 the algorithm converges very fast, already after 3 iterations the result is exact up to machine accuracy. Table 2 demonstrates the stability of the algorithm: when the initial value t_0 is chosen 10 times larger than proposed in (35) it takes 9 iterations until the machine accuracy is reached, if t_0 is taken 1.000 times larger, it takes another 7 steps. Thus, Algorithm 4 is very stable with respect to overestimation of the parameter t.

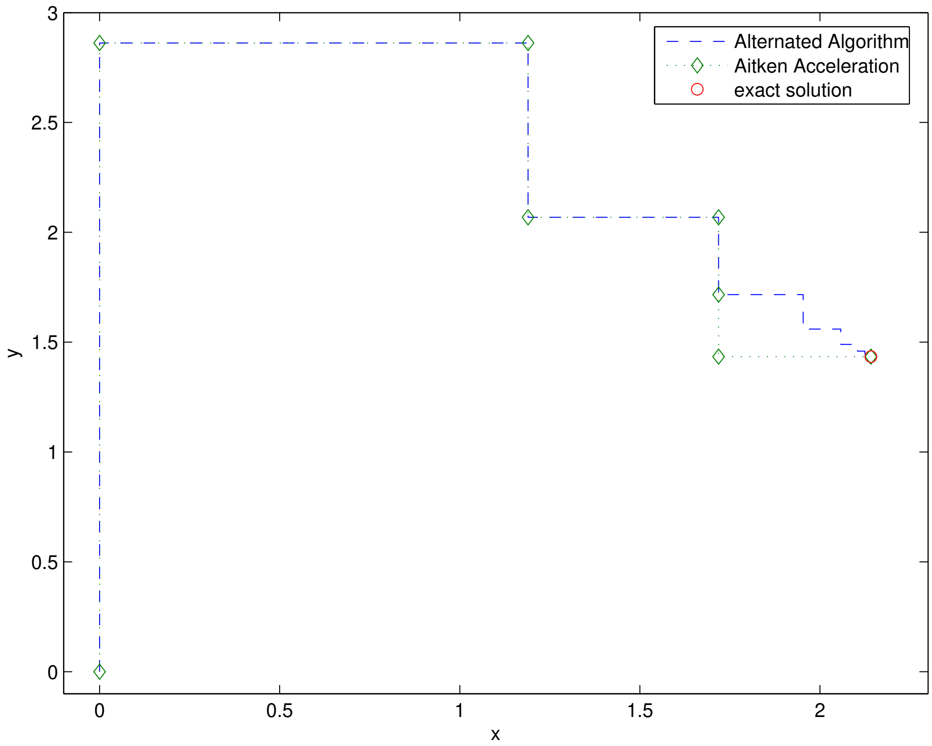

Figure 1: Comparison of Algorithm 1 and Algorithm 2 in Example I. Iterations of the alternated algorithm (Algorithm 1) are displayed as blue lines connecting the points (x, y) corresponding to plastic strain increment matrices $P_1 = \operatorname{diag}(x, -x), P_2 = \operatorname{diag}(y, -y)$. The vertical and horizontal lines indicate the direction of minimization described in steps (ii) and (iii) of Algorithm 1. Convergence is achieved after 18 iterations with the solution $(x, y) \approx (2.14142, 1.43431)$ displayed by a red circle. The diamond-shaped points show the converge speedup due to Aitken acceleration (Algorithm 2), where the extrapolated value of y coincides with the previous solution, but requires three iterations only.

i	t_i	ξ_1 approximation	ξ_2 approximation
0	565.6	39.0	38.0
1	297.8	20.0	19.0
2	164.6	10.0	9.6
3	99.5	6.0	5.0
4	69.7	3.9	2.9
5	58.9	3.1	2.1
6	57.0	3.03	2.03
7	56.9706	3.02843	2.02843
8	56.9705627485	3.02842712475	2.02842712475
9	56.970562748477128	3.028427124746190	2.028427124746190

Table 2: Evolution of the iterates in Example I when the initial guess is chosen 10 times larger than proposed in (35). Still after 9 iterations the result is numerically exact, after ≈ 4 iterations the convergence becomes quadratic.

5 Example II

Now we turn to a later time-step problem. Let the material parameters be given by (36) as in the previous example. Let the loading matrices read

$$\operatorname{dev} A_1 \approx \begin{pmatrix} 10.7071 & 0 \\ 0 & -10.7071 \end{pmatrix}, \quad \operatorname{dev} A_2 \approx \begin{pmatrix} 11.4142 & 0 \\ 0 & -11.4142 \end{pmatrix}.$$

5.1 Results for Algorithm 1 and Algorithm 2

The original Algorithm 1 with tolerance $\mathrm{tol} = 10^{-12}$ generates the sequence

$$P_1^1 \approx \begin{pmatrix} 1.1111 & 0 \\ 0 & -1.1111 \end{pmatrix}, \qquad P_2^1 \approx \begin{pmatrix} 3.3333 & 0 \\ 0 & -3.3333 \end{pmatrix},$$

$$P_1^2 \approx \begin{pmatrix} 1.6049 & 0 \\ 0 & -1.6049 \end{pmatrix}, \qquad P_2^2 \approx \begin{pmatrix} 2.5926 & 0 \\ 0 & -2.5926 \end{pmatrix},$$

$$P_1^3 \approx \begin{pmatrix} 1.8244 & 0 \\ 0 & -1.8244 \end{pmatrix}, \qquad P_2^3 \approx \begin{pmatrix} 2.2634 & 0 \\ 0 & -2.2634 \end{pmatrix},$$

$$P_1^4 \approx \begin{pmatrix} 1.9220 & 0 \\ 0 & -1.9220 \end{pmatrix}, \qquad P_2^4 \approx \begin{pmatrix} 2.1171 & 0 \\ 0 & -2.1171 \end{pmatrix},$$

$$P_1^5 \approx \begin{pmatrix} 1.9653 & 0 \\ 0 & -1.9653 \end{pmatrix}, \qquad P_2^5 \approx \begin{pmatrix} 2.0520 & 0 \\ 0 & -2.0520 \end{pmatrix},$$

$$P_1^6 \approx \begin{pmatrix} 1.9846 & 0 \\ 0 & -1.9846 \end{pmatrix}, \qquad P_2^6 \approx \begin{pmatrix} 2.0231 & 0 \\ 0 & -2.0231 \end{pmatrix},$$

i	t_i	ξ_1 approximation	ξ_2 approximation
0	60.5	2.4	2.3
1	68.3	2.87	2.89
2	67.46	2.829	2.829
3	67.455818	2.828424	2.8284293
4	67.4558168097922	2.82842399039352	2.828429214314649
5	67.455816809792168	2.828423990393518	2.828429214314643

Table 3: Quadratic convergence of the iterates generated by Algorithm 4 in Example II. The initial guess is chosen as in (35), after 5 iterations the result is exact up to machine accuracy.

and terminates after 18 iterations with the approximation

$$P_1^{18} \approx \begin{pmatrix} 2.000 & 0 \\ 0 & -2.000 \end{pmatrix}, \qquad P_2^{18} \approx \begin{pmatrix} 2.000 & 0 \\ 0 & -2.000 \end{pmatrix}.$$

Note that the norms of both matrices are

$$(|P_1^{18}|, |P_2^{18}|) \approx (2.82842721631375, 2.82842698739485).$$

After Aitken extrapolation as explained in Algorithm 2, the matrix P_2^{extr} calculated from the matrices P_2^1, P_2^2, P_2^3 is identical to P_2^{18} up to the first five digits. Thus, also for the multiple timestep problem, extrapolation yields a significant speedup.

For comparison, iterations of Algorithms 1 and 2 are displayed in Figure 2 simultaneously.

5.2 Results for Algorithm 4

The example configuration leads to the system of polynomials (29)

$$\begin{aligned} A + B\xi_1 + C\xi_1^2 - (2 + 6\,\xi_1 + 3\,\xi_2 + 5\,\xi_1\,\xi_2)^2 &= 0, \\ D + E\xi_2 + F\xi_2^2 - (2 + 6\,\xi_1 + 3\,\xi_2 + 5\,\xi_1\,\xi_2)^2 &= 0, \end{aligned}$$

with coefficients

$$\begin{aligned} A &= 260.568542, \quad B = 585.705965, \quad C = 329.137464, \\ D &= 917.136452, \quad E = 795.998776, \quad F = 172.715317. \end{aligned}$$

Table 3 shows the behavior of the approximations generated by Algorithm 4. The algorithm converges again very fast, already after 5 iterations the result is exact up to machine accuracy.

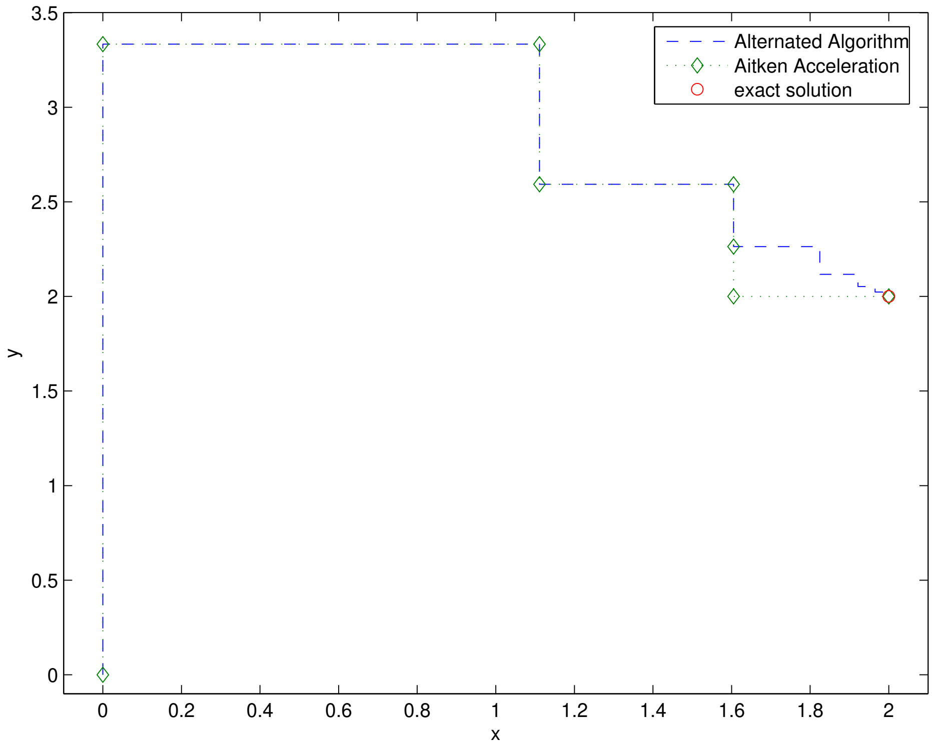

Figure 2: Comparison of Algorithm 1 and Algorithm 2 in Example II. Iterations of the alternated algorithm (Algorithm 1) are displayed as blue lines connecting the points (x, y) corresponding to plastic strain increment matrices $P_1 = \mathrm{diag}\,(x, -x), P_2 = \mathrm{diag}\,(y, -y)$. The vertical and horizontal lines indicate the direction of minimization described in steps (ii) and (iii) of Algorithm 1. Convergence is achieved after 18 iterations with the solution $(x, y) \approx (2.0000, 2.00000)$ displayed by a red circle. The diamond-shaped points show the converge speedup due to Aitken acceleration (Algorithm 2), where the extrapolated value of y coincides with the previous solution, but requires three iterations only.

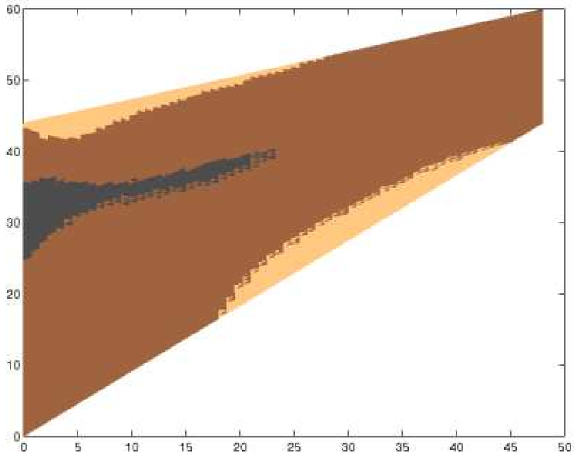

Figure 3: The black colour shows elastic upgrade zones (where $P_1 = P_2 = 0$), brown and lighter gray colours shows the first plastic upgrade ($P_1 \neq 0, P_2 = 0$) and of the both plastic upgrades ($P_1 \neq 0, P_2 \neq 0$) zones.

6 A real 2D computation of the elastoplatic continuum at the first plastic time-step

The presented Algorithms have been implemented in an existing Matlab code for calculation of two-yield elastoplastic deformations [BCV05, COJ05]. Let us consider the first plastic time-step of the Cook's membrane, whose mechanical setup is explained in [COJ05]. This problem was discretized using an uniform triangular mesh with 131072 triangles. Due to linearization steps in the nonlinear plastic problem, Problem 1 had to be solved on each triangle several times. Altogether, there were 393216 minimization problems to solve. Depending on the characteristics of the solutions, there are three types of triangles denoted by a different colour in Figure 3:

- Black colour triangles satisfying the condition $P_1 = P_2 = 0$.
- Brown gray triangles satisfying the condition $P_1 \neq 0$, $P_2 = 0$.
- Light gray triangles satisfying the condition $P_1 \neq 0$, $P_2 \neq 0$.

The following algorithms were taken to account: Algorithm 1, Algorithm 2 and Algorithm 3. Additionally, we also consider simplified versions of Algorithms 1 and 2 under the assumption of the one time-step problem ($\operatorname{dev} A_1 = \operatorname{dev} A_2$). Then, all upgrades in steps (ii) - (iv) requiring matrix

Method	Time (seconds)	Speedup
Algorithm 1	304.57	1.0
Algorithm 2	203.22	1.50
Algorithm 1 (one time-step optimized)	159.68	1.90
Algorithm 2 (one time-step optimized)	136.68	2.23
Algorithm 3	32.21	9.45

Table 4: Comparison of various methods for the two-yield elastoplastic Cook's membrane problem. There were 393216 minimization problems (Problem 1) to solve.

operations can be reduced to scalar operations only,

$$\begin{aligned} |\operatorname{dev} A_1 - 2\mu P_2^i| &= ||\operatorname{dev} A_1| - 2\mu |P_2^i||, \\ |\operatorname{dev} A_2 - 2\mu P_1^i| &= ||\operatorname{dev} A_1| - 2\mu |P_1^i||, \\ |P_1^{i+1} - P_1^i| + |P_2^{i+1} - P_2^i| &= ||P_1^{i+1}| - |P_1^i|| + ||P_2^{i+1}| - |P_2^i||. \end{aligned}$$

Thus only norms $|P_1^i|, |P_2^i|$ and $|\operatorname{dev} A_1|$ are stored and the value $|P_2^{\mathrm{exrt}}|$ is extrapolated in step (v) of Algorithm 2. The resulting Algorithms are denoted as "one time-step optimized". Table 4 reports on the performance of all five algorithms. Note, the highest contribution to the calculation time is spent due to the minimization problems leading to the case $P_1 \neq 0$, $P_2 \neq 0$, i.e., to the light gray colour triangles.

According to the theoretical expectation Algorithm 1 achieves the longest time. Thank to the extrapolation efficiency, Algorithm 2 saves about 30 percent of computational time. Above mentioned simplification reduce the time to another 20 percent. Obviously, the most efficient is 3 which provides exact solution without any iterations.

Conclusions

Algorithm 3 provides an explicit solution at the first plastic time-step and it is computationally the fastest. For later plastic time-steps, we suggest to use Algorithm 2 providing a significant acceleration of the convergence in regions of the first and the second plastic upgrades. However, due to the non-differentiability of the underlying problem (9), the convergence of Algorithm 2 remains an open question.

Acknowledgments

Both authors acknowledge support from the Austrian Science Fund 'Fonds zur Förderung der wissenschaftlichen Forschung (FWF)' for support under grant SFB F013/F1306 and F1308 in Linz, Austria.

References

[Atk89] Kendall E. Atkinson. *An introduction to numerical analysis. 2nd ed.* New York: John Wiley & Sons, Inc. xvi, 693 p., 1989.

[BCV04] M. Brokate, C. Carstensen, and J. Valdman. A quasi-static boundary value problem in multi-surface elastoplasticity: Part 1 – analysis. *Mathematical Models and Methods in Applied Sciences*, 27(14):1697–1710, 2004.

[BCV05] M. Brokate, C. Carstensen, and J. Valdman. A quasi-static boundary value problem in multi-surface elastoplasticity: Part 2 – numerical solution. *Mathematical Models and Methods in Applied Sciences*, 28(8):881–901, 2005.

[BS96] M. Brokate and J. Sprekels. *Hysteresis and Phase Transitions.* Springer-Verlag New York, 1996.

[COJ05] C. Carstensen, V. Orlando, and Valdman. J. A convergent adaptive finite element method for the primal problem of elastoplasticity. Technical Report 2005-12, Institute of Mathematics, Humboldt-Universität zu Berlin, 2005.

[ET99] I. Ekeland and R. Témam. *Convex Analysis and Variational Problems.* SIAM, 1999.

[KV03] J. Kienesberger and J. Valdman. Multi-yield elastoplastic continuum - modeling and computations. In M. Feistauer, Dolejší, P. Knobloch, and K. Najzar, editors, *Numerical mathematics and advanced applications. Proceedings of ENUMATH 2003.*, pages 539–548. Springer, 2003.

[NDR05] V. Nübel, A. Düster, and E. J. Rank. An rp-adaptive finite element method for elastoplastic problems. *Computational Mechanics (submited)*, 2005.

[Sto64] Josef Stoer. *Einführung in die numerische Mathematik.* Springer, Berlin, 1964.