

## Multidimensional rank-one convexification of incremental damage models at finite strains

Daniel Balzani, Maximilian Köhler, Timo Neumeier, Malte A. Peter, Daniel Peterseim

### Angaben zur Veröffentlichung / Publication details:

Balzani, Daniel, Maximilian Köhler, Timo Neumeier, Malte A. Peter, and Daniel Peterseim. 2024. "Multidimensional rank-one convexification of incremental damage models at finite strains." *Computational Mechanics* 73: 27–47.  
<https://doi.org/10.1007/s00466-023-02354-3>.



# Multidimensional rank-one convexification of incremental damage models at finite strains

D. Balzani<sup>1</sup> · M. Köhler<sup>1</sup> · T. Neumeier<sup>2</sup> · M. A. Peter<sup>3</sup> · D. Peterseim<sup>3</sup>

Received: 10 February 2023 / Accepted: 12 May 2023 / Published online: 17 June 2023  
© The Author(s) 2023

## Abstract

This paper presents computationally feasible rank-one relaxation algorithms for the efficient simulation of a time-incremental damage model with nonconvex incremental stress potentials in multiple spatial dimensions. While the standard model suffers from numerical issues due to the lack of convexity, our experiments showed that the relaxation by rank-one convexification delivering an approximation to the quasiconvex envelope prevents mesh dependence of the solutions of finite element discretizations. By the combination, modification and parallelization of the underlying convexification algorithms, the novel approach becomes computationally feasible. A descent method and a Newton scheme enhanced by step-size control prevent stability issues related to local minima in the energy landscape and the computation of derivatives. Numerical techniques for the construction of continuous derivatives of the approximated rank-one convex envelope are discussed. A series of numerical experiments demonstrates the ability of the computationally relaxed model to capture softening effects and the mesh independence of the computed approximations. An interpretation in terms of microstructural damage evolution is given, based on the rank-one lamination process.

**Keywords** Numerical relaxation · Rank-one convexification · Continuum damage mechanics

**Mathematics Subject Classification** 65K10 · 74G65 · 74A45

## 1 Introduction

The degradation of materials is a critical issue in continuum mechanics. Hence, the modeling of damage processes is important. The formation and evolution of voids in a material on the microscale deteriorates the mechanical properties and thus weakens the material on the macroscale. The microscopic damage affects the macroscale material response and effects such as stress softening and strain softening occur. The first one describes a reduction of material stiffness with an increase of stress, while strain softening refers to the reduction of stress with increasing strain. The phenomenological scalar-valued  $(1 - D)$ -approach for modeling damage and capturing these softening effects was introduced in [35]. This approach has been rigorously analyzed and further extended to the finite strain setting in [45, 57]. While this model describes real materials accurately, the direct description of strain softening possesses mathematical complications. Namely, strain softening often implies some degree of non-convexity in the underlying incremental stress potential (generalized energy density) and, thus, mesh

---

✉ M. Köhler  
maximilian.koehler@rub.de

D. Balzani  
daniel.balzani@rub.de

T. Neumeier  
timo.neumeier@uni-a.de

M. A. Peter  
malte.peter@uni-a.de

D. Peterseim  
daniel.peterseim@uni-a.de

<sup>1</sup> Continuum Mechanics, Ruhr-Universität Bochum, Universitätsstr. 150, 44801 Bochum, Germany

<sup>2</sup> Institute of Mathematics, University of Augsburg, Universitätsstr. 12a, 86159 Augsburg, Germany

<sup>3</sup> Institute of Mathematics & Centre for Advanced Analytics and Predictive Sciences (CAAPS), University of Augsburg, Universitätsstr. 12a, 86159 Augsburg, Germany

dependence will be observed. Mesh dependence refers to the behaviour that integral values of the finite element simulation will change despite choosing a very fine mesh, see e.g. [22, Section 6.4.2], [47, Section 11.2]. Another mathematical perspective on this issue is that the underlying Euler–Lagrange equation loses ellipticity. To analyze this, a framework that yields a pseudo-elastic potential is required, since most mathematical results on the existence of minimizers cover the case of a potential formulation with dependence on the gradient. The appropriate tool is the incremental variational framework, which was introduced in a series of papers, see e.g. [15, 32, 46, 51, 52]. This framework provides thermodynamically consistent potentials and *condenses* the dependence on the dissipative internal variable out of the problem, leading to a pseudo-elastic formulation per incremental step. The resulting generalized energy density incorporates, due to the condensation of the internal variable, non-convexity, which reflects the dissipative process. To circumvent the loss of ellipticity and, thus, mesh dependence, various regularization methodologies exist. The temporal regularization introduces viscous terms, see e.g. [29, 42, 43, 49, 59], while spatial regularization introduces non-locality either through neighbourhood integrals or gradient extensions regarding the damage variable [24, 25, 34, 40, 53, 55]. In the context of crack propagation, an alternative is the incorporation of non-local normalizations of crack surfaces, see [54, 61]. All these methodologies have in common that an additional parameter is introduced, which corresponds either to viscosity or an internal length scale.

The identification of these parameters is a difficult task and sometimes contradictory to the required numerical values. Another possible regularization technique is relaxation, which is the focus of this paper. In relaxation, one approach is to replace the original non-convex incremental stress potential by a (semi)convex envelope, which is the greatest (semi)convex function below the original function. This was done for the one dimensional setting of damage at finite strains in [9, 36, 37, 56], where all semiconvex notions coincide with convexity. For higher spatial dimensions, the microsphere approach, cf. [5, 28, 44], was used to obtain a three-dimensional material model via relaxed one-dimensional contributions. However, in the finite-strain setting of continuum damage mechanics, a full higher dimensional relaxation with a suitable semiconvex envelope has not been accomplished so far. To the authors' best knowledge, [58] is the only contribution with a numerical relaxation in the multidimensional case. There, a macroscopic averaged response is obtained by emulating a representative volume element under certain assumptions which yields a convex envelope per time step, thereby violating the compatibility conditions. On the contrary, semiconvex relaxation does not violate the compatibility condition; however, the construction of a semiconvex envelope is a sophisticated task. The

construction of a semiconvex envelope can be done analytically or numerically. The former is hard to realize for complex material laws and, thus, only few results are known as e.g. St.-Venant–Kirchhoff elasticity [41] on  $\mathbb{R}^{d \times d}$  (does not constrain  $\det \mathbf{F} > 0$ ), single slip plasticity [10, 12, 16, 18, 20] and nematic elastomers [23]. The latter can be approached in diverse ways either through enforcing laminate patterns in microstructural computations, cf. [6, 19, 39], or by solving several optimization problems. This contribution addresses the higher dimensional numerical rank-one relaxation in direct extension to [2, 26, 27] embedded in finite element simulations to show mesh insensitivity and properties of the obtained rank-one envelope which may enable the description of strain softening. The information tracked in the iterative convexification procedure is used to give a microstructural interpretation in terms of damage evolution.

This article is structured as follows. Section 2 discusses the  $(1 - D)$  approach and the resulting pseudo-time-incremental damage model. Afterwards, the notions of semiconvexity and the numerical relaxation via rank-one convexification, especially the algorithmic aspects, are discussed in Sects. 3 and 4. Further, some general implementation aspects are described in Sect. 5 and followed by the numerical simulation of boundary-value problems which show the mesh insensitivity as well as in-depth studies of the convexifications and their interpretation in terms of microstructural damage in Sect. 6. We conclude in Sect. 7 with some remarks on the results and future research topics, especially in the context of the efficiency regarding the numerical convexification.

## 2 Incremental damage modeling

We first introduce notation and basic concepts of continuum mechanics. We consider a physical body (connected, Lipschitz boundary) which in reference configuration is denoted by  $\mathcal{B} \subset \mathbb{R}^3$  with coordinates  $\mathbf{X} \in \mathcal{B}$ . The nonlinear deformation map  $\varphi_t : \mathcal{B} \rightarrow \mathcal{B}_t := \varphi_t(\mathcal{B})$  describes the deformation of the body relative to its reference configuration for times  $t \in \mathbb{R}^+$ . The deformation gradient is therefore given by  $\mathbf{F}(\mathbf{X}) = \text{Grad } \varphi_t(\mathbf{X})$  and we consider the right Cauchy–Green tensor  $\mathbf{C} = \mathbf{F}^T \mathbf{F}$  as deformation measure. Isotropic hyperelastic strain energy densities can be formulated in terms of the first, second and third principal invariants

$$I_1 = \text{tr } \mathbf{C}, \quad I_2 = \text{tr}(\text{Cof } \mathbf{C}), \quad I_3 = \det \mathbf{C} \quad (2.1)$$

of the Cauchy–Green tensor. The corresponding strain energy densities of virtually undamaged materials can then be phrased in the format

$$\psi^0(\mathbf{C}) = \psi^0(I_1, I_2, I_3).$$

As examples we utilize the St. Venant–Kirchhoff

$$\psi_{\text{St.V-K}}^0(\mathbf{C}) := \frac{\lambda}{8} (I_1 - 3)^2 + \frac{\mu}{4} (I_1^2 - 2I_1 - 2I_2 + 3) \tag{2.2}$$

and the compressible Neo-Hookean effective strain energy density

$$\psi_{\text{NH}}^0(\mathbf{C}) := \frac{\mu}{2} (I_1 - 3) - \mu \ln(J) + \frac{\lambda}{2} \ln(J)^2 \tag{2.3}$$

with the Lamé parameters  $\mu, \lambda \in \mathbb{R}$  and  $J = \sqrt{I_3}$ . In presence of damage, the modified strain energy density

$$\psi(\mathbf{C}, \beta) = (1 - D(\beta)) \psi^0(\mathbf{C}) \tag{2.4}$$

is often considered, see e.g. [45]. Here,  $\beta$  is an internal variable and evolves according to the discontinuous damage approach

$$\beta_t := \max_{s \leq t} [\psi^0(\mathbf{C}_s)] \quad \text{for } s, t \in \mathbb{R}^+ \tag{2.5}$$

and  $D : \mathbb{R} \rightarrow [0, 1]$  is the non-decreasing damage function of the form

$$D(\beta) = D_\infty \left( 1 - \exp\left(-\frac{\beta}{D_0}\right) \right), \tag{2.6}$$

where  $D_\infty \in (0, 1)$  is the asymptotic limit of the damage function and  $D_0 \in \mathbb{R}^+$  is the damage saturation parameter.

To relax the phenomenological scalar-valued continuum damage mechanics energy density, we briefly recapitulate the derivation of the associated incremental stress potential as obtained in [9]. Let  $t_0, \dots, t_N$  denote a time discretization of the time interval  $[0, T]$  and  $\Delta t_{k+1} := t_{k+1} - t_k$  denote the incremental time steps. Furthermore, let  $\dot{\psi}$  denote the time derivative of strain energy density and  $\phi$  the dissipation potential. According to [51], the generalized work done in the body within the time increment  $\Delta t_{k+1}$  is then given by

$$\mathcal{W}(\mathbf{F}_{k+1}, \beta_{k+1}) := \int_{t_k}^{t_{k+1}} \dot{\psi} + \phi \, dt. \tag{2.7}$$

The incremental stress potential

$$W(\mathbf{F}_{k+1}) = \inf_{\beta_{k+1}} [\mathcal{W}(\mathbf{F}_{k+1}, \beta_{k+1})], \tag{2.8}$$

which only depends on the deformation gradient  $\mathbf{F}_{k+1}$ , minimizes  $\mathcal{W}$  with respect to the internal variable  $\beta_{k+1}$  in each incremental time step. Via the second law of thermodynamics, one can derive that the dissipation potential is of the form

$\phi := \psi^0 \dot{D}$ . With this in mind, the analytical integration and minimization of (2.8) yields

$$W(\mathbf{F}) = \psi(\mathbf{F}, D) - \psi(\mathbf{F}_k, D_k) + \beta D - \beta_k D_k - \bar{D} + \bar{D}_k. \tag{2.9}$$

For a more detailed derivation, the reader is referred to [9]. Here,  $\bar{D}$  denotes the antiderivative of the damage function. For an improved readability, the time-step dependence was shortened by dropping the index  $k + 1$  and the abbreviations  $D := D(\beta)$ ,  $D_k := D(\beta_k)$ ,  $\bar{D} := \bar{D}(\beta)$ , and  $\bar{D}_k := \bar{D}_k(\beta)$  denote evaluations of the function  $D$  and its antiderivative  $\bar{D}$  at the minimizers  $\beta$  of the current time step and the values of the internal variable at the last time step  $\beta_k$ .

Let the potential energy of external forces  $\hat{t}$ , which are applied at the Neumann boundary  $\partial \mathcal{B}_\sigma$ , be denoted by  $\Pi^{\text{ext}}$ . Further, let  $\partial \mathcal{B}_\varphi$  denote the Dirichlet boundary and  $\hat{\varphi}$  the Dirichlet boundary conditions on  $\partial \mathcal{B}_\varphi$ . We consider the total potential energy

$$\Pi(\varphi) := \int_{\mathcal{B}} W(\mathbf{F}(\varphi)) \, dV + \Pi^{\text{ext}}(\hat{t}, \varphi).$$

The goal is now to follow the principle of minimum potential energy to obtain the state of mechanical equilibrium and to find minimizers of this energy subject to the boundary conditions, i.e.

$$\inf_{\varphi} \{ \Pi(\varphi) \mid \varphi = \hat{\varphi} \text{ on } \partial \mathcal{B}_\varphi, t = \hat{t} \text{ on } \partial \mathcal{B}_\sigma \}. \tag{2.10}$$

Setting the first variation of  $\Pi$  to zero leads to the weak form in the total Lagrangian setting

$$\int_{\mathcal{B}} \mathbf{P} : \text{Grad} \delta \mathbf{u} \, dV - \int_{\partial \mathcal{B}_\sigma} \hat{t} \cdot \delta \mathbf{u} \, dS = 0, \tag{2.11}$$

subject to Dirichlet boundary conditions  $\mathbf{u} = \hat{\mathbf{u}}$  on  $\partial \mathcal{B}_\varphi$ , see e.g. [33, Sec. 8.3] for a detailed derivation. The first Piola–Kirchhoff stress tensor  $\mathbf{P}$  and its derivative, the nominal tangent moduli  $\mathbb{A}$ , are computed as

$$\mathbf{P} = \frac{\partial W(\mathbf{F})}{\partial \mathbf{F}}, \quad \mathbb{A} = \frac{\partial \mathbf{P}}{\partial \mathbf{F}} = \frac{\partial^2 W(\mathbf{F})}{\partial \mathbf{F} \partial \mathbf{F}}. \tag{2.12}$$

The problem (2.11) can be solved numerically by using the finite element method with a suitable solver for the resulting system of nonlinear equations. For a steepest descent scheme like [4, Algorithm 9.1], only the first Piola–Kirchhoff stresses  $\mathbf{P}$  are required, whereas for second order methods, such as the Newton method (cf. [4, Algorithm 4.2]), the derivative of these, the tangent moduli  $\mathbb{A}$ , need to be computed or approximated additionally.

Due to the dissipative nature of damage evolution processes, the time-incremental energy density (2.9) becomes non-convex. Therefore, the direct numerical simulation of

this problem suffers from stability issues and mesh dependency.

### 3 Relaxation by rank-one convexification

To treat the drawbacks resulting from non-convexity, we apply a relaxation approach and replace the energy density  $W$  by a (semi)convex envelope. From a mathematical perspective, the quasiconvex envelope  $W^{qc}$  would be the hull of choice; however, for general energy densities no efficient algorithm is known for the numerical approximation. Apart from that, the growth conditions required in order to prove weak lower semicontinuity of the integral functional when dealing with quasiconvex energy densities limit physical requirements such as  $W(F) \rightarrow \infty$  for  $\det(F) \rightarrow 0$ . Hence, the incorporation of lower and upper bounds for  $W^{qc}$  – the convex, polyconvex and rank-one convex envelopes – is reasonable. Their relations are described by the inequality chain

$$W^c \leq W^{pc} \leq W^{qc} \leq W^{rc} \leq W.$$

Replacing  $W$  by the convex envelope  $W^c$  may lead to a nonphysical material behavior and is therefore of limited relevance in continuum mechanics models, cf. [17, Theorem 4.8-1]. The rank-one and the polyconvex hull are the closest available approximations, for which computational algorithms exist. In this paper, we focus on the rank-one convex hull. In particular, we will show that this choice is able to capture strain softening to some extent.

#### 3.1 Definition and alternative characterizations

A function  $W : \mathbb{R}^{d \times d} \rightarrow \mathbb{R}$  is called *rank-one convex*, if

$$W(\lambda \mathbf{A} + (1 - \lambda) \mathbf{B}) \leq \lambda W(\mathbf{A}) + (1 - \lambda) W(\mathbf{B}) \quad (3.1)$$

holds for all  $\lambda \in [0, 1]$  and all matrices  $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{d \times d}$  that are rank-one connected, i.e.  $\text{rank}(\mathbf{A} - \mathbf{B}) = 1$ . This condition is equivalent to the property that  $W$  is convex along rank-one directions, i.e. the function  $g : \eta \mapsto W(\mathbf{F} + \eta \mathbf{a} \otimes \mathbf{b})$  is convex for all  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$  and  $\mathbf{F} \in \mathbb{R}^{d \times d}$ . The *rank-one convex envelope*  $W^{rc}$  of a function  $W$  is the largest rank-one convex function below  $W$ , i.e.

$$W^{rc}(\mathbf{F}) = \sup\{\tilde{W}(\mathbf{F}) \mid \tilde{W} : \mathbb{R}^{d \times d} \rightarrow \mathbb{R} \text{ with } \tilde{W} \leq W, \tilde{W} \text{ is rank-one convex}\}. \quad (3.2)$$

Various characterizations for the rank-one convex hull can be found for example in [21, Chapter 5]. The first example is the construction by successive lamination. This is done by

setting  $W^0 = W$  and for  $k > 0$  the next laminate is obtained by

$$W^{k+1}(\mathbf{F}) = \inf_{\substack{\lambda \in [0, 1] \\ \mathbf{A}, \mathbf{B} \in \mathbb{R}^{d \times d}}} \left\{ \lambda W^k(\mathbf{A}) + (1 - \lambda) W^k(\mathbf{B}) \mid \begin{array}{l} \lambda \mathbf{A} + (1 - \lambda) \mathbf{B} = \mathbf{F}, \\ \text{rank}(\mathbf{A} - \mathbf{B}) = 1 \end{array} \right\} \quad (3.3)$$

at all points  $\mathbf{F} \in \mathbb{R}^{d \times d}$ . The lamination-convex envelope  $W^{lc}$  is defined in a pointwise manner through the limit

$$W^{lc}(\mathbf{F}) := \lim_{k \rightarrow \infty} W^k(\mathbf{F}).$$

In [38, 5.C], it is shown that the rank-one convex envelope can be obtained by successive lamination, that is  $W^{rc} = W^{lc}$ . This characterization of the rank-one convex envelope is exploited in the algorithmic computations below.

An alternative characterization links the rank-one characterization via the successive lamination to so called  $\mathcal{H}_M$  sequences. We utilize the notation from [26]. Shorthand,  $\mathcal{H}_M$  is a condition on a set of matrices  $\mathbf{F}_1, \dots, \mathbf{F}_M$  that they are in some sense hierarchically rank-one connected, the mathematically rigorous definition is given in the following. The sequence of tuples  $(\xi_i, \mathbf{F}_i) \in [0, 1] \times \mathbb{R}^{d \times d}$  for  $i = 1, \dots, M$  is said to be an  $\mathcal{H}_M$  sequence (notation:  $(\xi_i, \mathbf{F}_i) \in \mathcal{H}_M$ ) if  $\sum_{i=1}^M \xi_i = 1$  and the following condition holds:

- in case  $M = 2$  it holds  $\text{rank}(\mathbf{F}_1 - \mathbf{F}_2) = 1$ ;
- in case  $M > 2$  up to a permutation of the indices  $\{1, \dots, M\}$  it holds  $\text{rank}(\mathbf{F}_1 - \mathbf{F}_2) = 1$  and for all  $i \in \{2, \dots, M - 1\}$  define

$$\begin{aligned} \mu_1 &= \xi_1 + \xi_2, & \hat{\mathbf{F}}_1 &= \frac{1}{\mu_1} (\xi_1 \mathbf{F}_1 + \xi_2 \mathbf{F}_2), \\ \mu_i &= \xi_{i+1}, & \hat{\mathbf{F}}_i &= \mathbf{F}_{i+1} \end{aligned}$$

and it holds  $(\mu_i, \hat{\mathbf{F}}_i) \in \mathcal{H}_{M-1}$ .

With this definition at hand the rank-one convex envelope can be characterized by

$$W^{rc}(\mathbf{F}) = \inf \left\{ \sum_{i=1}^M \xi_i W(\mathbf{F}_i) \mid M \in \mathbb{N}, (\xi_i, \mathbf{F}_i) \in \mathcal{H}_M, \mathbf{F} = \sum_{i=1}^M \xi_i \mathbf{F}_i \right\} \quad (3.4)$$

where  $M \in \mathbb{N}$  cannot be bounded in general. The equivalence of the characterization is shown in [21, Proposition 5.16].

### 4 Numerical rank-one convexification

This section presents algorithms for the numerical approximation of rank-one convex envelopes along with micro-mechanical interpretations in terms of iterative rank-one lamination.

#### 4.1 The basic algorithm

We begin by recalling the important aspects of the algorithm presented in [2] and [4, Chapter 9.3.4]. By  $|\cdot|_\infty$  we denote the component-wise maximum norm of a vector or a matrix, e.g.  $|\mathbf{A}|_\infty = \max_{ij} |A_{ij}|$ . We assume a discretization of the energy density  $W : \mathbb{R}^{d \times d} \rightarrow \mathbb{R}$  as a continuous piecewise linear approximation  $W_{\delta,r}$  on a mesh

$$\mathcal{N}_{\delta,r} = \delta \mathbb{Z}^{d \times d} \cap \left\{ \mathbf{A} \in \mathbb{R}^{d \times d} \mid |\mathbf{A}|_\infty \leq r \right\} \tag{4.1}$$

with parameters for the mesh width  $\delta \in \mathbb{R}$  and the radius of the ball  $r \in \mathbb{R}$ . The radius needs to be chosen sufficiently large so that the mesh is able to capture all relevant minima contributing to the convexification. Further, a discrete set of rank-one directions is considered

$$\mathcal{R}_{\delta,r}^1 = \{ \mathbf{a} \otimes \mathbf{b} \mid \mathbf{a}, \mathbf{b} \in \delta \mathbb{Z}^d, \|\mathbf{a}\|_\infty \leq 2dr, 1-d\delta \leq \|\mathbf{b}\|_\infty \leq 1+d\delta \}. \tag{4.2}$$

For the sake of readability, we denote the discretized rank-one set by  $\mathcal{R}$ . Set  $W_{\delta,r}^0(\mathbf{F}) = W(\mathbf{F})$  for all  $\mathbf{F} \in \mathcal{N}_{\delta,r}$ . The iterative lamination is now performed in a pointwise manner. Therefore, for all  $\mathbf{F} \in \mathcal{N}_{\delta,r}$  the optimization problem

$$W_{\delta,r}^{k+1}(\mathbf{F}) = \inf \left\{ \lambda W_{\delta,r}^k(\mathbf{F} + \delta l_1 \mathbf{R}) + (1 - \lambda) W_{\delta,r}^k(\mathbf{F} + \delta l_2 \mathbf{R}) \mid \begin{matrix} \mathbf{R} \in \mathcal{R}, \lambda \in [0, 1], l_1, l_2 \in \mathbb{Z} \\ \lambda l_1 + (1-\lambda) l_2 = 0 \end{matrix} \right\} \tag{4.3}$$

is solved. Function values corresponding to points that are not represented in the mesh  $\mathcal{N}_{\delta,r}$  are obtained by linear interpolation. The constraint on  $l_1, l_2$  and  $\lambda$  in the above optimization problem ensures that the convex combination of the arguments of  $W_{\delta,r}^k$  leads to  $\mathbf{F}$ . The iteration is stopped until either a number of maximal iterations  $k_{\max}$  is reached or the change in two consecutive iterates is small enough, i.e.  $|W_{\delta,r}^{k+1} - W_{\delta,r}^k|_\infty \leq \text{tol}$ .

The solution of the minimization problem can be performed by considering for every rank-one direction  $\mathbf{R} \in \mathcal{R}$  the one-dimensional convexification along the line  $\mathbf{F} + \eta \mathbf{R}$  for  $\eta \in \mathbb{R}$ . We adapt the notation of [27] and denote by  $\ell_\delta(\mathbf{F}, \mathbf{R})$  the points along the rank-one direction  $\mathbf{R} \in \mathcal{R}$  through the point  $\mathbf{F}$ , i.e.

$$\ell_\delta(\mathbf{F}, \mathbf{R}) = \{ \mathbf{F} + l \delta \mathbf{R} \mid l \in \mathbb{Z} \} \cap \text{conv}(\mathcal{N}_{\delta,r}). \tag{4.4}$$

For fixed iteration  $k$  and fixed mesh point  $\mathbf{F} \in \mathcal{N}_{\delta,r}$  as well as direction  $\mathbf{R} \in \mathcal{R}$ , let  $w_l = W_{\delta,r}^k(\mathbf{F} + l \delta \mathbf{R})$  denote the function values of  $W_{\delta,r}^k(\ell_\delta(\mathbf{F}, \mathbf{R}))$  and  $I_\ell$  the index set of all  $l$  of (4.4). Since the matrices in  $\ell_\delta(\mathbf{F}, \mathbf{R})$  are not directly represented in the mesh in general, the function values of these intermediate points are obtained by interpolation. The mapping  $q : I_\ell \rightarrow W_{\delta,r}^k(\ell_\delta(\mathbf{F}, \mathbf{R}))$  can now be interpreted as a one-dimensional function, mapping each index  $l \in I_\ell$  to the value  $w_l$ . This one-dimensional function needs now to be convexified. This allows for the application of the convexification procedure presented in Algorithm 1. Afterwards the minimum over all  $\mathbf{R} \in \mathcal{R}$  is stored as the resulting relaxed value for the mesh point  $\mathbf{F}$  in the current iteration.

#### 4.2 One-dimensional convexification

Let  $L$  be the number of matrices in the set  $\ell_\delta(\mathbf{F}, \mathbf{R})$ , i.e.  $L := |\ell_\delta(\mathbf{F}, \mathbf{R})|$ , and assume that the distribution of points along the rank-one line is equidistant with distance  $\delta$ . Let the  $x_i \in \mathbb{Z}$  for  $i = 0, \dots, L$  denote the indices in the set  $I_\ell$ . The convexification of the mapping  $x_i \mapsto w_i$  (a shifted/renumbered version of  $q$ ) can then be realized through the following procedure described in Algorithm 1.

---

**Algorithm 1** convexify( $\mathbf{x}, \mathbf{w}$ )

---

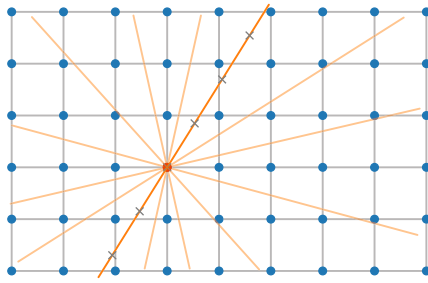
```

1: Input:  $\mathbf{x}, \mathbf{w}$  ▷ arrays of length  $L$ 
2: Output:  $\mathbf{y}, \mathbf{c}$  ▷ arrays of length  $n$ 
3:  $\mathbf{y}[1] = \mathbf{w}[1], \mathbf{y}[2] = \mathbf{w}[2], \mathbf{c}[1] = \mathbf{w}[1], \mathbf{c}[2] = \mathbf{w}[2]$ 
4:  $n = 2$ 
5: for  $i = 3, 4, \dots, L$  do
6:   while  $(\mathbf{c}[n] - \mathbf{c}[n-1]) * (\mathbf{x}[i] - \mathbf{y}[n]) >=$ 
7:      $(\mathbf{w}[i] - \mathbf{c}[n]) * (\mathbf{y}[n] - \mathbf{y}[n-1])$  and  $n >= 1$  do
8:      $n -= 1$ 
9:      $n += 1$ 
10:   $\mathbf{y}[n] = \mathbf{x}[i], \mathbf{c}[n] = \mathbf{w}[i]$ 
11: return  $\mathbf{y}, \mathbf{c}$ 

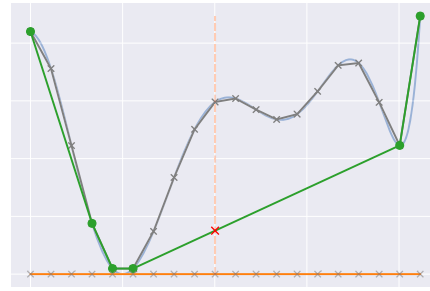
```

---

This algorithm realizes the convexification by iteration from left to right of the interval, successively computing difference quotients and checking two adjacent slopes for convexity. Depending on the angle of the two slopes, either the points already visited are deleted (by overwriting  $\mathbf{y}[n]$  and  $\mathbf{c}[n]$ ) or the new point is added (line 9). The algorithm realizes the convexification in linear complexity in the number of input points  $L$ , due to the deletion of points already visited and not contributing to the convex hull. The resulting arrays of length  $n$  contain the supporting points of the convex hull. In computational geometry, this procedure is also known as Graham’s scan [31]. An adaptive convexification strategy, which accelerates the convexification significantly by using second order derivative information, can be found in



(A) Mesh  $\mathcal{N}_{\delta,r}$  (blue) and discretized direction set (orange) in a single mesh point



(B) Convexification of the one dimensional mapping  $I_\ell \rightarrow W_{\delta,r}^k(\ell_\delta(\mathbf{F}, \mathbf{R}))$  (gray) by Algorithm 1 delivers the green hull

**Fig. 1** Rank-one convexification. **a** schematic representation of the deformation gradient mesh  $\mathcal{N}$  (projected to two dimensions) and discretized rank-one directions (orange lines) through point  $F$ . **b** one-dimensional convexification of the function based on the interpolations from **a**. Original function is drawn in light blue, piece-wise linear inter-

polation in gray, resulting one-dimensional convex hull in green, the function value for  $F$  (red) is obtained by linear interpolation, which is given by the convex combination of function values corresponding to  $F^+$  and  $F^-$

[37]. After the convexification the function value of the convex hull corresponding to  $F$  is obtained by interpolation of the function  $y \rightarrow c$  at the index  $x_i \in 0, \dots, L$  corresponding to the point  $l = 0$ .

The solution of the optimization problem (4.3) and hence one global iteration  $k \rightarrow k+1$  in the rank-one convexification algorithm is of complexity  $\mathcal{O}(N^{d^2+2d+1})$ , where  $N$  denotes the number of discretization points in one matrix component in the mesh. The overall number of performed iterations can be interpreted as the maximum lamination depth of the microstructure.

**Remark 4.1** (An alternative algorithm by [26, 27]) While the above algorithm originally introduced by [2, 4] only alters the function value of a single mesh point in each one-dimensional convexification, the algorithm presented in [26, 27] considers all mesh points along a specified rank-one line and updates all of them after the one-dimensional convexification; the characterization of the discrete set of rank-one direction matrices is therefore different. The sets of rank-one directions considered are characterized by

$$\mathcal{R}_k^1 = \{\delta \mathbf{a} \otimes \mathbf{b} \mid \mathbf{a}, \mathbf{b} \in \mathbb{Z}^d, |\mathbf{a}|_\infty, |\mathbf{b}|_\infty \leq k\} \quad (4.5)$$

with  $k \in \mathbb{N}$ . Due to the update of function values for all mesh points along the rank-one line, the one-dimensional convexifications can not be carried out over all points simultaneously. However, for a fixed rank one direction parallelization is possible by running lines in parallel through the mesh.

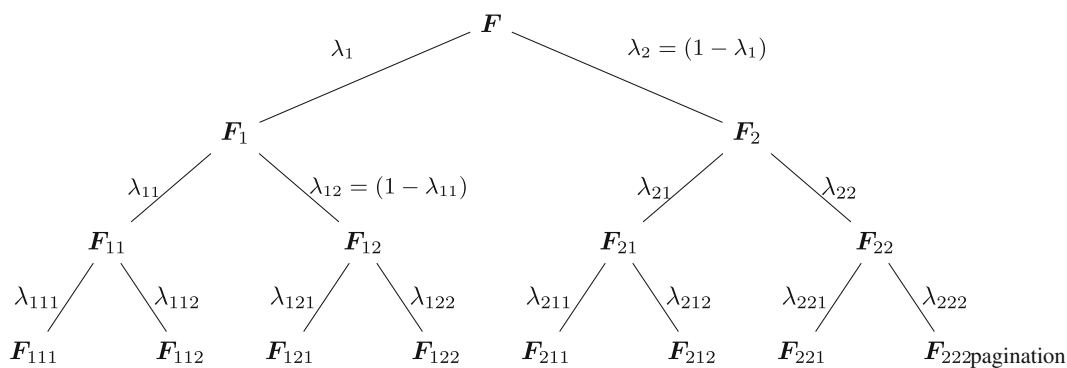
### 4.3 Rank-one lamination tree

The consecutive optimization problems (4.3) only require the function values of the iterates  $W_{\delta,r}^k$  for all mesh points.

Deeper insights into the structure of the hull can be gained if further knowledge is tracked. This is possible by considering the hierarchical construction of the  $\mathcal{H}_M$  sequences. By tracking the solution arguments of the optimization problem (4.3), it is possible to set up a lamination tree for each point  $F$  that is convexified, as illustrated in Fig. 2. For each convexified node  $F$ , a child is appended for  $F^+$  and  $F^-$  and their associated convex coefficient  $\lambda$  is tracked as weight over the edges. After each global iteration  $k$ , the tree has to be updated in terms of the new minimizers and their subtrees (see Algorithm 2 for details). A slight adjustment is required if the tree is computed numerically; the points (gray points on the orange rank-one line in the left picture of Fig. 1)  $F^+ = F + \delta l_2 \mathbf{R}$  and  $F^- = F + \delta l_1 \mathbf{R}$  (where  $\mathbf{R}, l_1, l_2$  denote the minimizing arguments of (4.3)) are in general not represented in the mesh. Due to this aspect, one has to consider the nearest neighbours of these points that contribute to the linear interpolation. Hence the rank-one tree contains branching subject to the interpolation (16 children) and subject to the rank-one lamination (two children).

Even if the setup of the tree seems straightforward from an iterative perspective, the tree can change substantially in one iteration, due to the occurrence of different minima corresponding to new lamination distributions. Hence, whole subtrees are subject to change in an iteration.

The tree can be used to approximate derivatives (or in general subdifferentials) of the convex hull by linear combination of the leaves of the tree. Those leaves correspond to the last branching, i.e. the case  $\mathcal{H}_2$ . Examples of lamination trees and the microstructural interpretation will follow in Sect. 6.



**Fig. 2** A rank-one tree obtained by successive lamination (for iterate  $k = 3$ ) of a function  $W$  at the point  $F$ . Note that only branching due to lamination and not due to interpolation is illustrated. The  $F_1$  and  $F_2$  can be identified with  $F^-$  and  $F^+$ . In the deeper parts of the tree

$F_{121}$  and  $F_{122}$  are for example associated to  $F_{12}^-$  and  $F_{12}^+$ , due to the branching of  $F_{12}$ . In general, the rank-one tree can be unbalanced due to bifurcation only in one subtree

### 5 Complexity reduction and efficient implementation

While the convexification algorithms appear promising on paper, the main challenge is the overall computational workload as outlined in Sect. 4.2 above. To make the rank-one convexification computationally feasible, this section presents various adjustments that significantly accelerate the previous algorithms. In addition, the nontrivial extraction of derivative information from the computed convex hull for the optimization algorithms is addressed.

#### 5.1 Reduction of rank-one lines

Originally, the discretization of the rank-one directions, the set  $\mathcal{R}_{\delta,r}^1$ , is coupled to the discretization of the deformation gradient space, the mesh  $\mathcal{N}_{\delta,r}$ . This coupling is motivated by the theoretical convergence proof presented in [2, 3]. However, the huge set of directions may be pessimistic from a practical perspective. Our numerical experiments show that a reduced set of directions suffices to achieve computationally feasible simulations of simplistic boundary value problems. Therefore, we use the discrete direction set  $\mathcal{R}_k^1$  as defined in (4.5) for  $k = 1$  as proposed in [27, Sec. 5.1]. This results in the set

$$\mathcal{R}_1^1 = \{ \delta \mathbf{a} \otimes \mathbf{b} \mid \mathbf{a}, \mathbf{b} \in \mathbb{Z}^d, |\mathbf{a}|_\infty, |\mathbf{b}|_\infty \leq 1 \} \tag{5.1}$$

consisting of 16 distinct rank-one directions if symmetries are exploited, that is the sign of the directions is omitted. The accuracy of the approximation using the reduced set  $\mathcal{R}_1^1$  is studied in Sect. 6.2 (cf. Figure 4, 5) by comparing the associated results to the ones obtained from the full set of discretized rank-one directions  $\mathcal{R}_{\delta,r}^1$ .

#### 5.2 Adaptive discretization of the deformation-gradient space

The mesh  $\mathcal{N}_{\delta,r}$  was constructed in (4.1) as an equidistant discretization, to be more precise, a hypercube with edge length  $2r$  and resolution  $\delta$  of the space  $\mathbb{R}^{d \times d}$ . The mesh size parameter  $\delta$  and the radius of the hypercube were applied to all of the components of the matrix space. Letting  $N = \frac{2r}{\delta}$  denote the number of grid points in one matrix component, i.e. unit direction of the space  $\mathbb{R}^{d \times d}$ , the mesh consists of  $N^{d^2}$  points. To reduce the number of mesh points, we now treat each component individually. Instead of setting up a hypercube of edge length  $2r$  we consider points in the cuboid that is characterized by  $F_{ij}^{\min} \leq F_{ij}^{\max}$  for  $i, j = 1, \dots, d$ . Furthermore, the mesh-width parameter is adjusted for each dimension, i.e.  $\delta_{ij} \in \mathbb{R}$ . Overall, we utilize the discrete set of points

$$\mathcal{N}_{\delta,r} = \left\{ \mathbf{F} \in \mathbb{R}^{d \times d} \mid F_{ij} \in \delta_{ij} \mathbb{Z}, F_{ij}^{\min} \leq F_{ij} \leq F_{ij}^{\max} \right\}$$

as our deformation gradient grid. For simplicity, we use the shorthand notation  $\mathcal{N}$  instead of  $\mathcal{N}_{\delta,r}$  for the deformation gradient mesh as long as there is no confusion about parameters. To be more precise, depending on the boundary value problems and the expected occurrence of deformations, the bounds of each entry in the discretization are set accordingly and the shear components are not fully resolved in the mesh  $\mathcal{N}$  but only represented by a coarse discretization.

Together with the reduction of the discretized rank-one direction set this discretization approach turns out reasonable in our numerical experiments. Complexity in (other) applications may also be reduced by other (adaptive) deformation gradient space discretizations while using larger rank-one sets simultaneously. The complexity reduction by considering smaller rank-one discretized sets or the possibility

to reduce complexity via the deformation gradient mesh-size should also incorporate the possibility of parallelization, i.e. parallelization over the directions only allows the usage of number of directions as threads, whereas the parallelization over mesh points and larger computational resources offers the possibility for major computational gain.

### 5.3 Reliable computation of derivative information

To perform a descent method or a (quasi-)Newton scheme, first-order and second-order derivative information of the incremental stress potential is required to set up the first and second variation of the energy functional. However, the constructed piecewise linear rank-one convex envelope is only  $C^0$  and its derivative is not well defined, e.g. in the mesh points. With the regularity results of [7] in mind, our consistent computation of derivatives utilizes the two-folded decomposition of the approximated rank-one convex hull  $W_{\delta,r}^k(\mathbf{F})$  in its  $\mathcal{H}_k$  sequence and, if necessary, the weighted sum of the linear interpolation to formulate a derivative in dependence on the smoother function  $W(\mathbf{F})$ , i.e.

$$\partial_{\mathbf{F}} W_{\delta,r}^k(\mathbf{F}) = \begin{cases} \xi \partial_{\mathbf{F}} W_{\delta,r}^{k-1}(\mathbf{F}^+) + (1 - \xi) \partial_{\mathbf{F}} W_{\delta,r}^{k-1}(\mathbf{F}^-) & \mathbf{F} \in \mathcal{N} \\ \sum_i^{2^{d \times d}} w_i \partial_{\mathbf{F}} W^k(\mathbf{F}_i) & \mathbf{F} \notin \mathcal{N}. \end{cases} \quad (5.2)$$

Note that the index  $k$  only refers to the lamination/global iteration. The branching in the tree due to interpolation is only encoded in the sum of current iteration ( $k$ ) laminates. In the case  $\mathbf{F} \in \mathcal{N}$ , the matrices  $\mathbf{F}^+$  and  $\mathbf{F}^-$  correspond to the minimizing arguments of the optimization problem (4.3). Since the linear interpolation (as handled in the case  $\mathbf{F} \notin \mathcal{N}$ ) can have in the two-dimensional case a maximum of  $2^{2 \times 2} = 16$  nodes per cell, the tree can branch into a maximum of sixteen different children where the volume fractions correspond to the interpolation weights, i.e. the  $w_i \in [0, 1]$  for the mesh points  $\mathbf{F}_i$ . Further, if a given deformation gradient  $\mathbf{F}$  is in the set  $\mathcal{N}$ , the branching is bounded by two children. However, the resulting tree is unbalanced and not structured at all, which renders a sophisticated construction task. Thus, we present a stack (queue) based algorithm to construct the two-folded decomposition tree. Algorithm 2 presents the procedure and works as follows: first, the root node is constructed and the queue is initialized. The queue consists of a tuple containing the node and its parent. The first candidates are queued from line 6 to 14 and if the given deformation gradient  $\mathbf{F}$  is not part of the convexification grid, it is decomposed into the weights and points that contribute to the interpolation sum. Afterwards a two staged while loop begins. In the first stage, the children are pushed into the associated parent node container, and in the second stage, the queue is filled with new potential lamination candidates. This while loop ends as soon as the queue is empty, returning

the root node, which is connected to all its descendants that are needed to construct the recursive derivative.

---

#### Algorithm 2 `buildtree`( $\mathbf{F}, \mathcal{N}$ , lamination forrest, startdepth)

---

```

1: Input:  $\mathbf{F}, \mathcal{N}$ , laminationforrest, startdepth
2: Output: root
3: ▷ Initialize tree with empty children array and  $\xi = 1.0$ 
4: root = Tree( $\mathbf{F}, 1.0$ , startdepth+1, [])
5: initialize queue
6: if  $\mathbf{F} \in \mathcal{N}$  then
7:   laminate = getlamine( $\mathbf{F}$ , laminationforrest, startdepth)
8:   push(queue, (laminate, root))
9: else
10:  points, weights = decompose( $\mathbf{F}, \mathcal{N}$ )
11:  for (weight, point)  $\in$  (weights, points) do
12:    depth = highestlamineorder
      (point, laminationforrest)
13:    candidate = Tree(point, weight, depth, [])
14:    push(queue, (candidate, root))
15: while queue  $\neq \emptyset$  do
16:  (lc, parent) = pop(queue)
17:  if lc is a laminate then
18:     $\xi = (\|\text{parent}.\mathbf{F} - \text{lc}.\mathbf{F}^-\|_1) / (\|\text{lc}.\mathbf{F}^+ - \text{lc}.\mathbf{F}^-\|_1)$ 
19:    push(parent.children, Tree(node. $\mathbf{F}^-$ , (1 -  $\xi$ ), lc.k, []))
20:    push(parent.children, Tree(node. $\mathbf{F}^+$ ,  $\xi$ , lc.k, []))
21:  else
22:    push(parent.children, lc)
23:  for child  $\in$  parent.children do
24:    depth = child.k - 1
25:    if child. $\mathbf{F} \in \mathcal{N}$  then
26:      laminate = getlamine
      (child. $\mathbf{F}$ , laminationforrest, depth)
27:      push(queue, (laminate, child))
28:    else
29:      points, weights = decompose(child. $\mathbf{F}, \mathcal{N}$ )
30:      for (weight, point)  $\in$  (weights, points) do
31:        candidate = Tree(point, weight,
      depth+1, [])
32:        push(queue, (candidate, child))
33: return root

```

---

With the tree at hand, the derivative can be recursively evaluated according to (5.2). A pseudocode of this procedure is given in Algorithm 3. The constructed tree is traversed until a leaf node is found and, in this case, the original non-convex function and its derivatives can be evaluated. Afterwards, the values are returned and multiplied by the convex combination or weighting coefficient, respectively.

---

**Algorithm 3** `eval(tree, material, history)`:  
 Recursive function and derivative value construction

---

```

1: Input: tree, material, history
2: Output: W, P, A
3: W = 0, P = 0, A = 0
4: n = 1
5: if tree.children = ∅ then ▷ tree is a leafnode
6:   W += W(tree.F), P += ∂W(tree.F), A += ∂2W(tree.F)
7: else
8:   for child ∈ tree.children do
9:     Wc, Pc, Ac = eval(child, material, history) ▷ recursion call
10:    W += child.ξ Wc, P += child.ξ Pc, A += child.ξ Ac
11: return W, P, A
    
```

---

An alternative approach for the construction of derivative information is to consider a representative subdifferential of the constructed multidimensional linear interpolation of the rank-one convex envelope, whenever an entry of the deformation gradient lies on a cell boundary. Multiple subdifferentials in the approximate neighborhood need to be evaluated and averaged. However, at points where change of signs in the subdifferential occurs as it is the case at the transition from compression to tension, a special treatment is required. Since the compression and tension regime are not of the same size, symmetry is not given and, thus, the subdifferential average is distorted. Hence, the subdifferential is set to zero in this case.

### 5.4 Overall procedure

For a given boundary value problem, in each pseudo-time iterate a nonlinear solver is utilized where the rank-one convexification is performed in each quadrature point. The construction of the derivatives was already outlined and hence a steepest descent method or a Newton scheme can be applied with a line search or Armijo-Goldstein stepsize criterion. For steepest descent and Armijo-Goldstein, see [4, Algorithm 9.1]; for Newton, see [4, Algorithm 4.2]. The rank-one convexification is performed with a direction set (5.1) or (4.2) according to the optimization problem (4.3). From a computational point of view, the direction set  $\mathcal{R}$  is obtained by creating all vectors  $\mathbf{a}$ ,  $\mathbf{b}$  whose outer products characterize the direction set.

### 5.5 Implementation details

Implementation details play a crucial role to get into a range of computational efficiency where a concurrent computation of the rank-one convex envelope is feasible. Thus, this subsection discusses implementation decisions. Our implementation in Julia [8] enables a significant gain in compu-

tational efficiency by parallelization of the one-dimensional convexifications. Each deformation gradient and, thus, the convolution over the deformation gradient grid is parallelized. A crucial step for the parallelization is the use of *lazy datastructures*, cf. [50]. Here, the notion of *lazy datastructures* refers to a datastructure whose underlying data generation is postponed to the very moment it is queried. In particular, this means for the deformation gradient grid that each grid point is computed ad hoc and not saved in a large multidimensional array. With an explicit data structure, which buffers the full deformation grid in an array, we observed significant performance degradation. Another important part is the parallelization strategy, which is like the concept of *workstreams*, cf. [60]. Each deformation gradient point in the deformation gradient grid formulates a task and for each thread an associated buffer is assigned, such that the allocation is thread-wise minimal. The only difference to the *workstreams* approach is that no coloring is required since the convexification grid points can be treated independently. The tensorial representation of the deformation gradient, stresses and tangent moduli is realized by the continuum mechanics tensors library `Tensors.jl` [13]. The multi-dimensional linear interpolation of `Interpolations.jl` [1] is used for the construction of  $W_{\delta,r}^k$  and boundary value problems are assembled with the finite element toolbox `Ferrite.jl` [14]. The code for the numerical rank-one convexification can be found in the following git repository <https://github.com/koehlerson/NumericalRelaxation.jl>.

## 6 Numerical experiments

This section demonstrates the functionality of the proposed rank-one convexification algorithm. First, the convexification procedure is numerically analyzed when applied to sample energy densities. Then, the efficiency of the proposed algorithmic extensions is illustrated in a series of numerical experiments (Subsections 6.1 – 6.4). In the Subsections 6.5 – 6.8 we proceed with the examination of several boundary value problems in terms of mesh insensitivity and their ability to capture strain softening. Finally, we conclude with a microstructural interpretation of the relaxation process in Subsection 6.9.

The presented experiments are based on the effective strain energy densities  $\psi^0$  of the St. Venant–Kirchhoff (STVK) model (2.2) and the Neo-Hooke (NH) model (2.3). For the initial examples in two spatial dimensions, we consider a mesh  $\mathcal{N}$  that is parameterized by  $F_{ij}^{\min} \leq F_{ij} \leq F_{ij}^{\max}$  and  $\delta_{ij}$  for  $i, j = 1, \dots, d$ . Hence, every matrix component is discretized individually as discussed in Sect. 5.2. We use the same discretization for both diagonal elements represented by the parameters  $F_{ii}^{\min}$ ,  $F_{ii}^{\max}$  and  $\delta_{ii}$ . A different discretiza-

tion represented by the parameters  $F_{ij}^{\min}$ ,  $F_{ij}^{\max}$  and  $\delta_{ij}$  for  $i \neq j$  is utilized for both off-diagonal entries.

As a stopping criterion for the global loop of the convexification procedure, we apply  $\max_{F \in \mathcal{N}} |W_{\delta,r}^{k+1} - W_{\delta,r}^k|_{\infty} \leq 10^{-4}$  while at the same time guaranteeing that a total of  $k_{\max} = 20$  iterations is not exceeded. The number  $k_{\max}$  corresponds to the maximal lamination depth of the microstructure evolution.

### 6.1 Convergence for the different direction sets

First we analyze the distance between two consecutive iterates to illustrate the convergence behavior of the algorithm. We compare the obtained approximations of the rank-one convex envelopes when using the reduced rank-one set  $\mathcal{R}_1^1$  and the full direction set  $\mathcal{R}_{\delta,r}^1$ , the latter one can be interpreted as reference solution. For this purpose, a material point study for the incremental potential  $W$  (as in (2.9)) is carried out for both, an NH and an STVK effective strain energy density  $\psi^0$ . The material parameters and the convexification grid parameters used are listed in Table 1. Note that we used the same discretization stepsize  $\delta_{ij}$  for all matrix components.

The tables presented in Fig. 3 show the convergence of the algorithm for an incremental stress potential  $W$  as in (2.9) based on the given parameter set in Table 1. In the case of the STVK model, 1000 was set as the starting value which was chosen instead of setting it to  $\infty$ . The maximal decrease

$$\max_{F \in \mathcal{N}} W_{\delta,r}^k(F) - W_{\delta,r}^{k+1}(F) \tag{6.1}$$

in iteration  $k$  is compared for both, the full direction set (4.2) and the reduced direction set (5.1). The results show that the reduced set of directions is enough and information of decrease in function values of the mesh points is sufficiently propagated by the smaller set of directions. The observed increase in iterations and hence slower convergence of the reduced directions is manageable since the overall costs massively outperform the full rank-one set. The number of elements in the set  $\mathcal{R}_1^1$  is sixteen, which is significantly lower than the number of elements in the  $\mathcal{R}_{\delta,r}^1$  set, which consists of 93925 and 122199 elements for the NH and SVKT parameters given in Table 1, respectively. The quality of the converged hull is compared in the next subsection.

### 6.2 Comparison of reduced and full rank-one direction sets

Figures 4 and 5 show the relative error of the approximations of the two direction sets, i.e.,

$k$	$\mathcal{R}_{\delta,r}^1$	$\mathcal{R}_1^1$	$k$	$\mathcal{R}_{\delta,r}^1$	$\mathcal{R}_1^1$
1	1000.00	1000.00	1	2.84696	2.84696
2	1000.04	1000.04	2	1.68770	1.62022
3	0.18478	0.18478	3	1.14215	1.14215
4	0.02101	0.07939	4	0.01503	0.09311
5	0.00278	0.01625	5	0.00453	0.01674
6	0.00105	0.00237	6	0.00174	0.00401
7	0.00062	0.00132	7	0.00072	0.00182
8	0.00043	0.00075	8	0.00044	0.00084
9	0.00013	0.00045	9	0.00026	0.00056
10	-	0.00037	10	0.00012	0.00042
11	-	0.00014	11	-	0.00031
12	-	-	12	-	0.00027
13	-	-	13	-	0.00020
14	-	-	14	-	0.00013
15	-	-	15	-	0.00011

(A)
(B)

Fig. 3 Distance of consecutive iterates (6.1) for the models based on the strain energy densities  $\psi^0$  for the STVK (left) and NH (right) model, stopping tolerance  $10^{-4}$

$$\mathcal{E}^{\text{rel}}(\mathbf{F}) = \frac{|W_{\delta,r,\mathcal{R}_{\delta,r}^1}^{\hat{k}}(\mathbf{F}) - W_{\delta,r,\mathcal{R}_1^1}^k(\mathbf{F})|}{\gamma + |W_{\delta,r,\mathcal{R}_{\delta,r}^1}^{\hat{k}}(\mathbf{F})|}$$

for  $\mathbf{F} \in \mathcal{N}$ . Here,  $\hat{k}$  corresponds to the converged iteration index of the reference approximation computed by the full rank-one direction set and  $k$  corresponds to the iteration for the converged approximation computed by the reduced direction set. The stabilization parameter  $\gamma$  is introduced, since both functions exhibit a value of zero multiple times and, thus, the division by zero produces undefined values. The stabilization parameter has been set to  $\gamma = 10^{-8}$ .

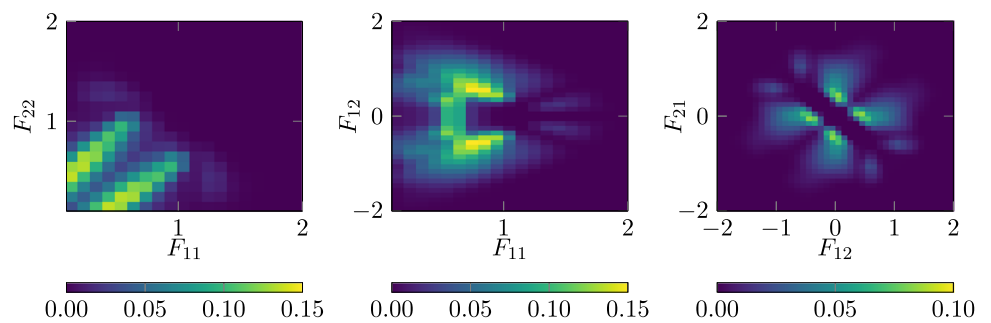
For all plotted planes in Figs. 4 and 5 the fixed axes correspond to initial, undeformed configuration deformation values  $\mathbf{F}_0 = \mathbf{I}$ . In other words, all plots show slices of the mesh hypercuboid that contain the identity matrix, i.e. the point  $\mathbf{I}$ . Compared to the absolute function values of  $W_{\delta,r}^k$ , the difference in the two possible approximations is quite small from a quantitative perspective. Hence, replacing the full rank-one direction set  $\mathcal{R}_{\delta,r}^1$  by  $\mathcal{R}_1^1$  seems reasonable in this representative model problem.

Notice that due to the asymmetry of the compression ( $0 \leq F_{ii} \leq 1$ ) and tension ( $1 \leq F_{ii}$ ) regime in the  $\mathbb{R}^{d \times d}$  space, the equidistant discretization favors the tension regime in terms of error distribution. This is due to the smaller size of the compression region and therefore coarser approximation, while the same discretization step size in the tension range is able to capture the behavior of the incremental stress potential slightly better because of the slower function growth for  $F_{ii} \rightarrow \infty$ . For an overall better approximation in the compression zone, one should prefer a finer discretization in the deformation gradient space instead of taking a larger rank-

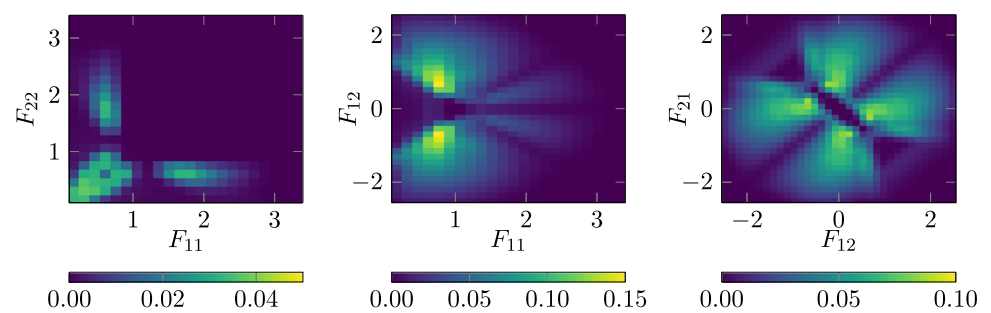
**Table 1** Material and convexification parameters for the  $2 \times 2 = 4$  fully resolved deformation grid material point study

$\psi^0$	$\lambda$	$\mu$	$D_0$	$D_\infty$	$\beta_k$	$\delta_{ij}$	$F_{ii}^{\min}$	$F_{ii}^{\max}$	$F_{ij}^{\min} \forall i \neq j$	$F_{ij}^{\max} \forall i \neq j$	$ \mathcal{N} $
NH	0.5	1.0	0.3	0.9	0.06	0.15	0.1	3.4	-2.55	2.55	648025
STVK	0.5	1.0	0.3	0.9	0.07	0.1	0.1	2.0	-2.0	2.0	672400

**Fig. 4** Relative error  $\mathcal{E}^{\text{rel}}$  for  $W_{\delta,r, \mathcal{R}_{\delta,r}^1}^9$  and  $W_{\delta,r, \mathcal{R}_1^1}^{11}$  obtained by the direction sets  $\mathcal{R}_1^1$  and  $\mathcal{R}_{\delta,r}^1$  based on the STVK model for  $\psi^0$



**Fig. 5** Relative error  $\mathcal{E}^{\text{rel}}$  for  $W_{\delta,r, \mathcal{R}_{\delta,r}^1}^{10}$  and  $W_{\delta,r, \mathcal{R}_1^1}^{15}$  obtained by the direction sets  $\mathcal{R}_1^1$  and  $\mathcal{R}_{\delta,r}^1$  based on the NH model for  $\psi^0$



one direction set. The information of ‘function lowering’ due to convexification seems to be propagated by the reduced direction sufficiently (requiring a higher number of global iterations) and a refinement in terms of directions seems to raise no substantial gain. Hence, with regard to refinement at the cost of higher workload, the spatial resolution of the deformation gradient should be improved instead of increasing the number of rank-one directions.

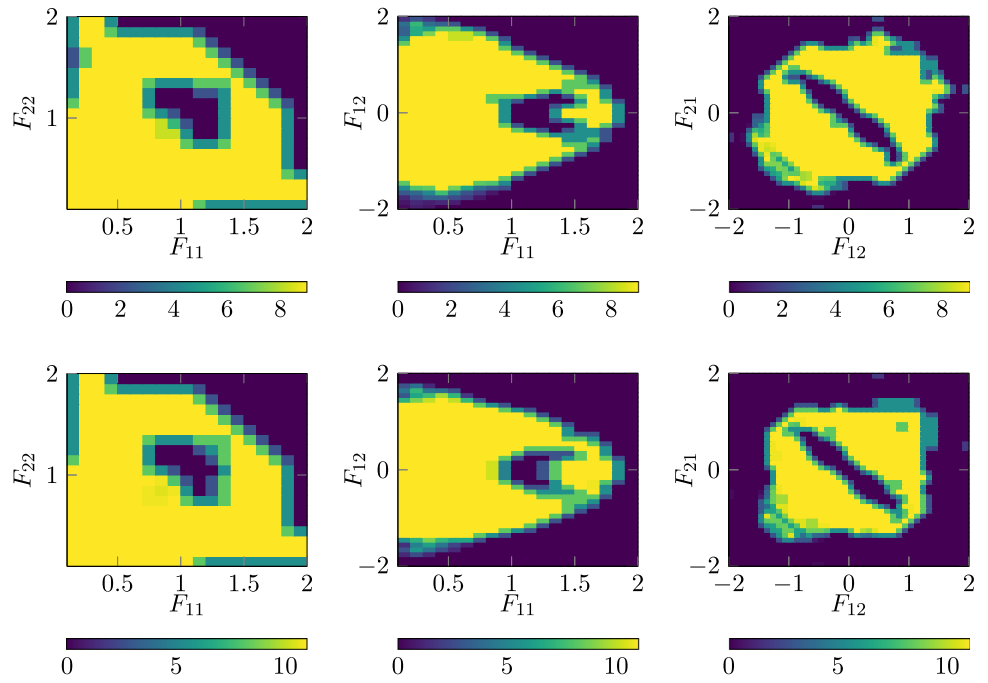
### 6.3 Lamination matrix

The Figs. 6 and 7 show slices of the computed states corresponding to the last row of the tables in Fig. 3 for the STVK and NH model, respectively. The pictures show lamination values on slices of the mesh hypercuboid through the point  $\mathbf{I}$  as described in the previous subsection. For two given matrix component values, the lamination order for the full (4.2) and reduced directions (5.1) are illustrated. Again, the higher lamination order of the reduced directions is observable while the qualitative behavior is similar. Also the convex (lamination order 0) regime around the identity matrix  $\mathbf{I}$  is visible. This illustration also motivates the replacement of  $\mathcal{R}_{\delta,r}^1$  by  $\mathcal{R}_1^1$ .

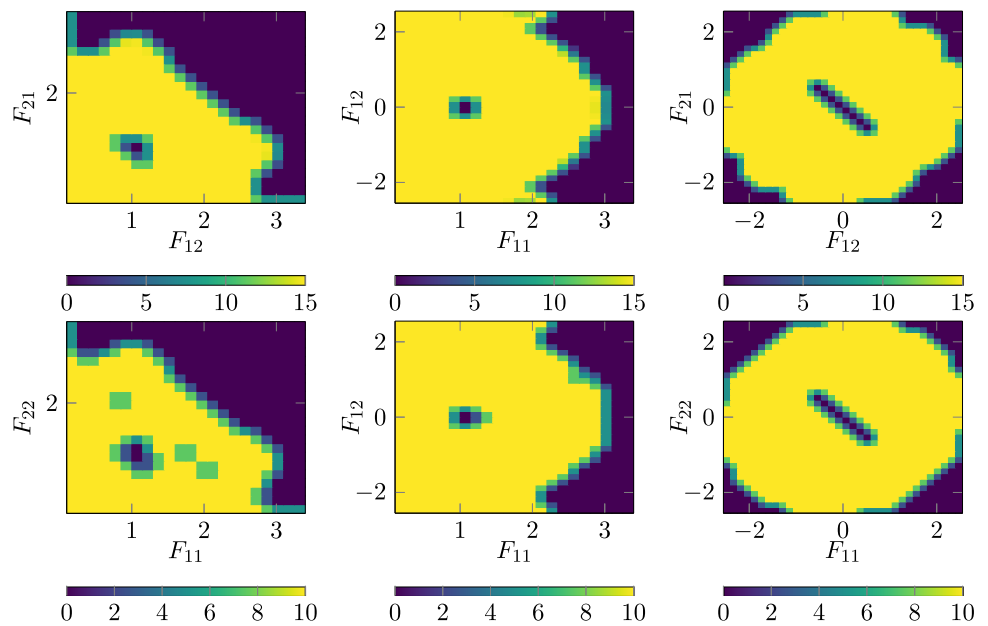
### 6.4 Strong scaling of the rank-one convexification

The parallelized version of the convexification algorithm is tested in a strong scaling setting on a single cluster node with an increasing number of threads with a fixed amount of total work, i.e. convexification grid size. The scaling study was done without building the decomposition tree. If the tree is needed either each thread has its own partial lamination forest which needs to be merged after the parallel part of the code or a parallel safe data structure for the lamination forest can be used. In our implementation, each thread has its own partial lamination forest that is merged after the parallel part of the code. The implementation’s performance can be seen in Fig. 8 which is close to the logarithmic perfect scaling. Julia’s garbage collector is realized by a *stop the world* implementation, cf. [48, Section 7]. Therefore, the gain of having the deformation grid not buffered in the memory comes with a tradeoff. While, initially, there is a gain in performance due to the avoided paging faults, cache misses and similar effects, the performance degrades from perfect scaling since there are more threads whose local memory needs to be cleaned in a serialized way after finishing the threaded convexification. However, since convexification meshes scale drastically in memory, we still advocate for lazy convexification grid representation.

**Fig. 6** Matrix of lamination order for St. Venant–Kirchhoff  $\psi^0$ . The color corresponds to the iteration  $k$  for which the last time a laminate was built. (top) shows the obtained lamination matrix for  $\mathcal{R}_{\delta,r}^1$  and (bottom) in the case of  $\mathcal{R}_1^1$ . Note that the convex regime, i.e. the area, where either a fully dissipated configuration ( $D = D_\infty$ ) is reached or ellipticity is not lost yet ( $F \approx I$ ), does not need lamination



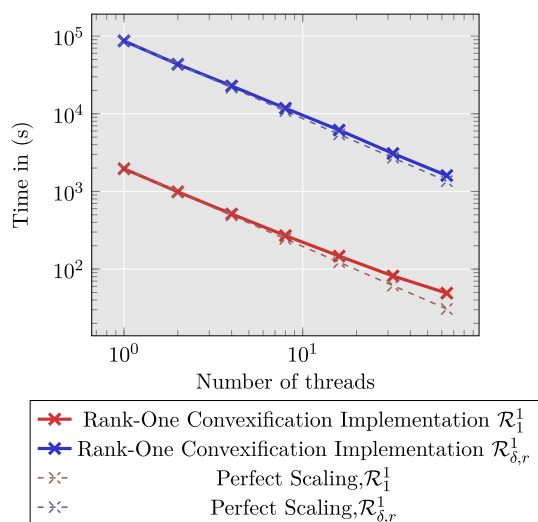
**Fig. 7** Lamination matrix for Neo-Hooke effective strain energy density. Again, the qualitative behavior is similar between the (top):  $\mathcal{R}_{\delta,r}$  direction set and (bottom):  $\mathcal{R}_1^1$  approach



Until now, the focus was on the pure application of the convexification algorithm, its convergence and lamination properties using different direction sets. In the next subsections, we test the obtained rank-one convex hull approximations in terms of mesh dependence under different loading and material conditions.

### 6.5 Uniaxial mesh independence test

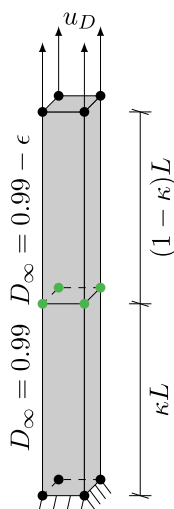
As a first mesh independence test, we consider the two element perturbation test of [30] and [9]. However, here bilinear quadrilateral elements are considered. The boundary value problem is depicted in Fig. 9a and the associated force-



**Fig. 8** Strong scaling of Neo-Hooke example with a convexification mesh consisting of 43681 points on a cluster node with an Intel Xeon Phi 7210 processor. The scaling consists of fixing the total workload while varying the number of threads by 1, 2, 4, 8, 16, 32, and 64

displacement curve in Fig. 9b. Within this boundary value problem, a pure uniaxial deformation is exhibited. Therefore, only the diagonal entries of the deformation gradient are resolved by the convexification grid. In order to introduce inhomogeneity to the problem, the asymptotic damage limit  $D_\infty$  in one of the two finite elements is perturbed by a small, physically meaningless value  $\epsilon = 10^{-5}$ , such that  $D_\infty$  is replaced by  $D_\infty - \epsilon$ . The total length  $L = 1$  is fixed; however, the individual element size of both elements is varied by the parameter  $\kappa$ . The derivatives were constructed with the subdifferential approach in this case. Then, only the first variation of the potential is known, and therefore, the descent method of [4, Algorithm 9.1] with Armijo–Goldstein based

**Fig. 9** The same uniaxial perturbation test as in [9, 30] for a Neo-Hooke effective strain energy density. Due to the exhibited rank-one path of the deformation, the same convex response as in the aforementioned contributions can be seen for the relaxed model. The unrelaxed model shows mesh dependence, where the descent method snaps back into different minima for different values of  $\kappa$

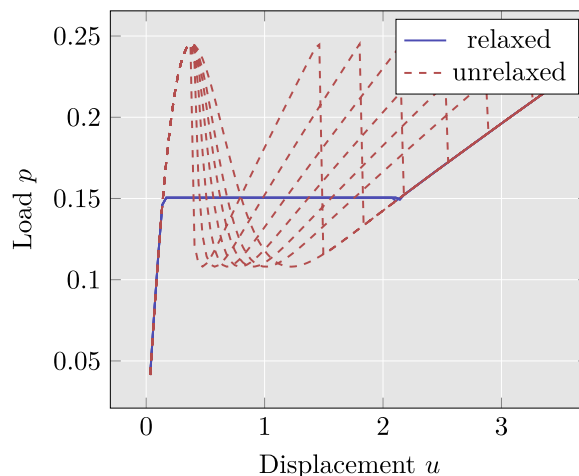


(A) Uniaxial boundary value problem

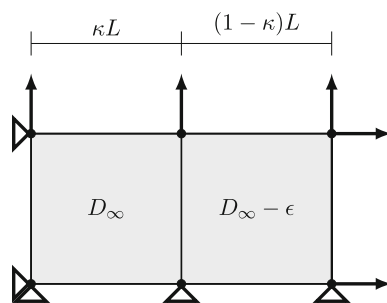
linesearch is used. The parameters of the Armijo–Goldstein linesearch were set to  $\alpha = 0.5$  and  $\hat{\mu} = 0.01$ , respectively. As material parameters, the following settings were chosen: The Neo-Hookean effective strain energy density  $\psi_{NH}^0$  is used with  $\lambda = 0.5, \mu = 1.0, D_0 = 0.3, D_\infty = 0.9$  and the convexification grid was spanned with a stepsize of 0.15 from 1.0 to 3.4 for both diagonal entries of the deformation gradient. From the force–displacement curve, an analogous behaviour as in [9] can be seen. As soon as the non-convex regime is entered, the relaxed model is activated and a constant stress response is observed. Since the deformation path can be parameterized by a rank-one path, the approximated rank-one convex envelope is convex along the path, thus the constant stresses. Furthermore, a mesh-independent response can be observed since different discretizations  $\kappa$  do not lead to a different response. This contrasts the unrelaxed basic model, which shows heavy mesh sensitivity.

### 6.6 Biaxial mesh independence test

The biaxial tension boundary value problem illustrated in Fig. 10 is used for a further mesh sensitivity tests. The damage limit parameter in the right element is again disturbed by a small parameter  $\epsilon = 10^{-5}$  and the discretization of the individual elements is varied by the parameter  $\kappa \in \{0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1.0\}$ . The disturbance parameter  $\epsilon$  is applied for varying  $\kappa$  to the right element. Material and convexification parameters can be found in Table 2. In contrast to the material point study, the relaxation is performed in each non-convex step until the convexified regime is reached. After that, the approximated rank-one convex envelope is fixed and reused in subsequent incremental time steps. The Figs. 11a and 12a show the unrelaxed and the Figs. 11b and 12b the relaxed (and then fixed) incremental



(B) Uniaxial mesh sensitivity in load displacement diagram



**Fig. 10** Boundary value problem for the biaxial mesh sensitivity test with disturbance  $\varepsilon = 10^{-5}$  in the asymptotic damage limit parameter in the right element. The parameter  $\kappa$  characterizes length of the elements

stress potential along rank-one and rank-two lines which are given by the paths

$$\mathbf{F}_s^{r1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + s \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \mathbf{F}_s^{r2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + s \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (6.2)$$

for varying  $s$ . Along the rank-two line (which corresponds to the evolution of the deformation of this boundary value problem), non-convexity of the deformation path is visible. Indeed, the non-convexity along the paths is required to reflect some strain softening (since a decreasing slope is required) as illustrated in Fig. 13.

This illustration shows the resulting material responses of the biaxial test using STVK and NH effective strain energy densities. In both cases, the derivative was computed with the novel tree decomposition procedure. It was not possible to obtain a convergent finite element solver by using the subdifferential description of the derivative. Since the tree decomposition approach is also capable of describing tangent moduli (which may vanish across rank-one lines), a Newton scheme with Armijo–Goldstein based line search was applied for the St. Venant–Kirchhoff case. As pointed out in [4, Remark 9.6], the second derivative can vanish for semi-convex envelopes; however, in this special case rank-two deformation paths are expected and, thus, the second derivative does not vanish. In the Neo-Hooke effective strain energy density case, the descent method with line search as in the uniaxial example was used again.

The unrelaxed model shows strain softening, but also considerably strong mesh sensitivity. Despite the observed non-convexity along the rank-two line, the response is mesh independent. Note that although some strain softening is observed along specific lines, the relaxation approach presented here with fixed convex envelopes does not represent a suitable model for the description of a realistic strain softening. The reason is two-fold: (i) strain-softening cannot be modeled specifically, and (ii) strain-softening cannot be obtained along rank-one lines, which, however, may be par-

ticularly relevant in practice. If specifically strain-softening needs to be modeled, the concept of the approach in [36] can be considered, where the convex hull is recomputed in each time step.

## 6.7 Lamination depth convergence study

As already mentioned, the number of global iterations of the convexification algorithm can be interpreted as the number of successive rank-one laminations, and hence the number of laminates that can occur in the microstructure. In Fig. 14, the response of the biaxial boundary value problem of the previous section is shown in relation to the lamination depth (or number of global lamination iterations in the convexification algorithm). While so far, only criteria with respect to the function values were discussed, we discuss in this section the convergence in terms of the first Piola–Kirchhoff stresses  $\mathbf{P}$ . The stresses serve as a better convergence indicator, since their value will be used in the finite element solver. Only in special cases, e.g. line searches, the rank-one convex envelope function value is needed. Since the hull is fixed, the evolution path of the deformation gradient in the biaxial setting somehow crosses some left non-convexities which could be an explanation for the buckling. Interestingly, at some points, the approximation coincides with the converged response. It is recognizable that 5 global iterations already deliver a suitable convex hull since the higher lamination depths do not produce a notable change in material response. In other words, the convexification procedure delivered a meaningful material response after only five iterations.

## 6.8 Triaxial test

We consider a generalization of the boundary value problem of the previous section to three spatial dimensions and show that the proposed convexification procedure is also applicable in this higher, nine-dimensional setting. However, this has currently been limited to a single increment in a single material point and, thus, corresponds to a single determination of a three-dimensional microstructure. Again, the off-diagonal entries, the six shear components, have been represented in the mesh  $\mathcal{N}$  by a very coarse discretization. All parameters related to energy densities and convexification are listed in Table 3.

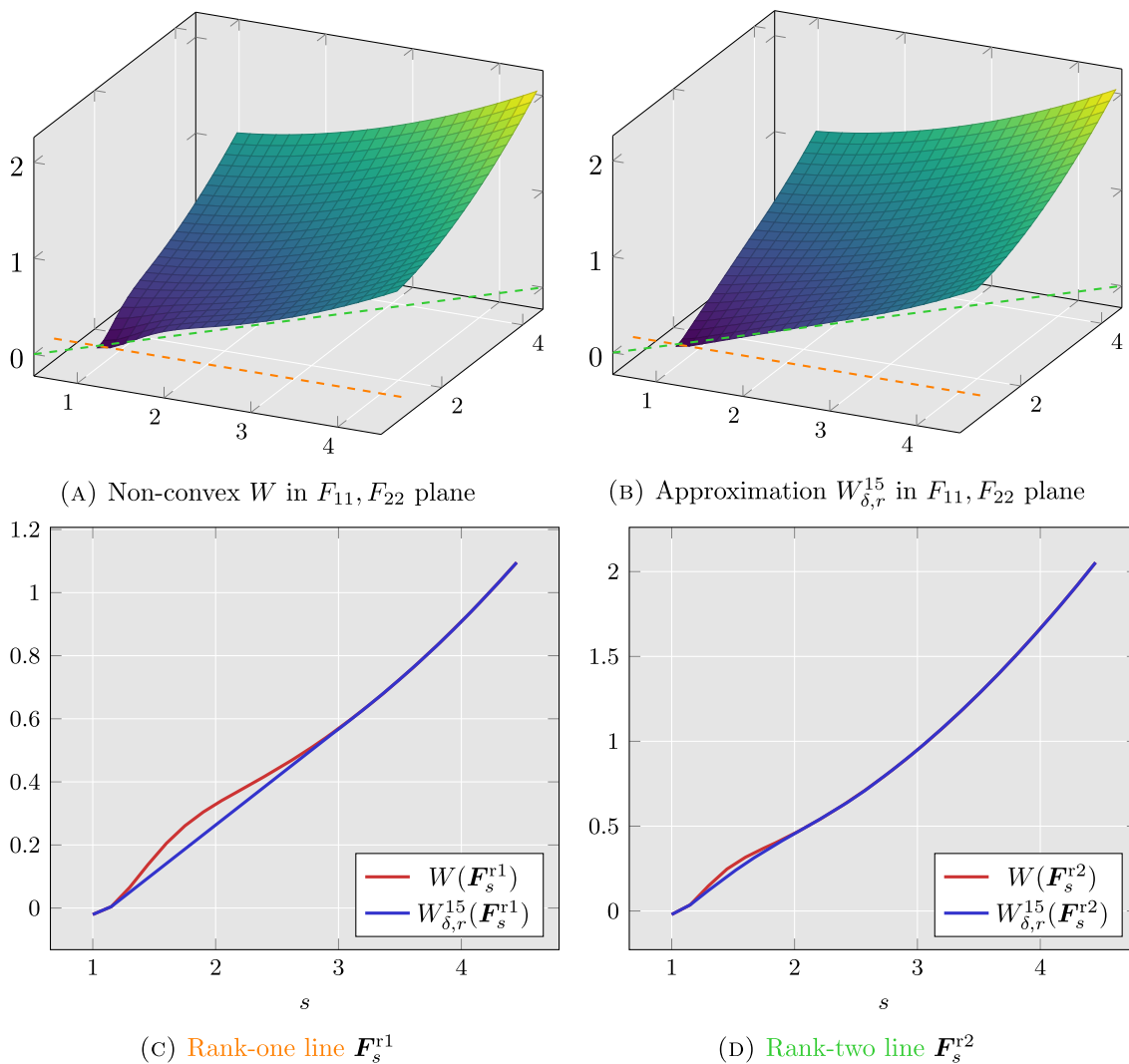
Figure 15 shows slices of the energy densities that correspond to paths of uni-, bi-, and triaxial boundary value problems. These paths are characterized as follows. Let  $\mathbf{I}_d \in \mathbb{R}^{d \times d}$  denote the identity matrix. The plotted rank- $d$  lines are described by

$$\mathbf{F}_s = \mathbf{I}_d + s \mathbf{I}_d, \quad (6.3)$$

**Table 2** Parameter settings for the biaxial perturbation test.

$\psi^0$	$\lambda$	$\mu$	$D_0$	$D_\infty$	$\delta_{ij}$	$F_{ii}^{\min}$	$F_{ii}^{\max}$	$F_{ij}^{\min} \forall i \neq j$	$F_{ij}^{\max} \forall i \neq j$	$ \mathcal{N} $
NH	0.5	1.0	0.3	0.9	0.15	1.0	3.4	-0.15	0.15	2601
STVK	0.5	1.0	0.4	0.99	0.15	1.0	3.1	-0.15	0.15	2025

Here, the convexification grid is specially constructed by coarsely discretizing the off-diagonal components of the deformation gradient

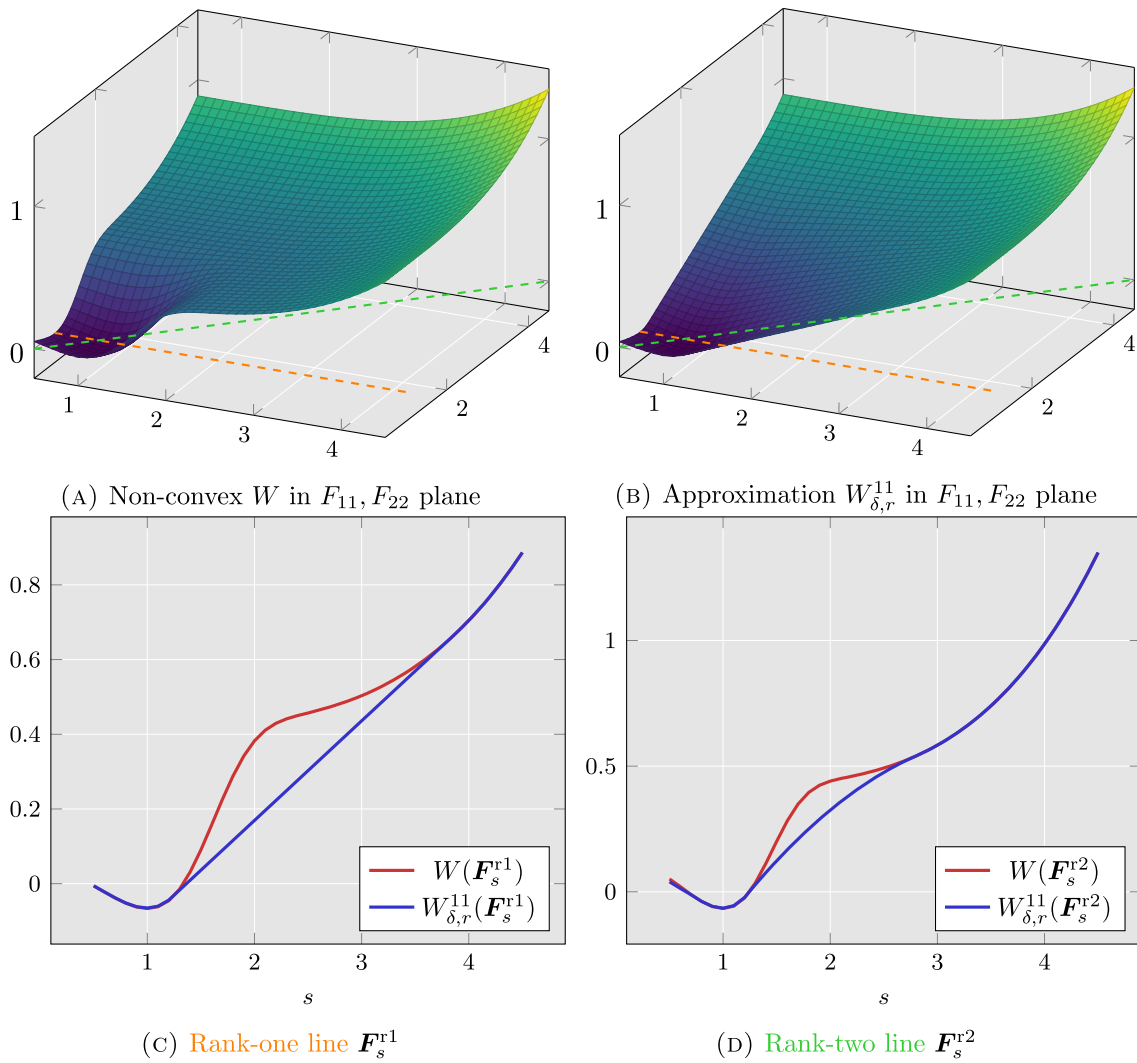


**Fig. 11** Non-convex incremental stress potential **(a)** and the associated rank-one convex envelope approximation  $W_{\delta,r}^k$  **(b)** for a Neo-Hooke effective strain energy density  $\psi^0$ . The value  $s$  corresponds to the line

characterizing equations (6.2), where **c** shows the rank-one path and **d** the rank-two path, respectively. Note that the rank-one convex envelope is convex along the rank-one line but non-convex following the rank-two line

**Table 3** Material and convexification parameters for the  $3 \times 3 = 9$ -dimensional convexification for the uni-, bi-, and triaxial deformation configurations

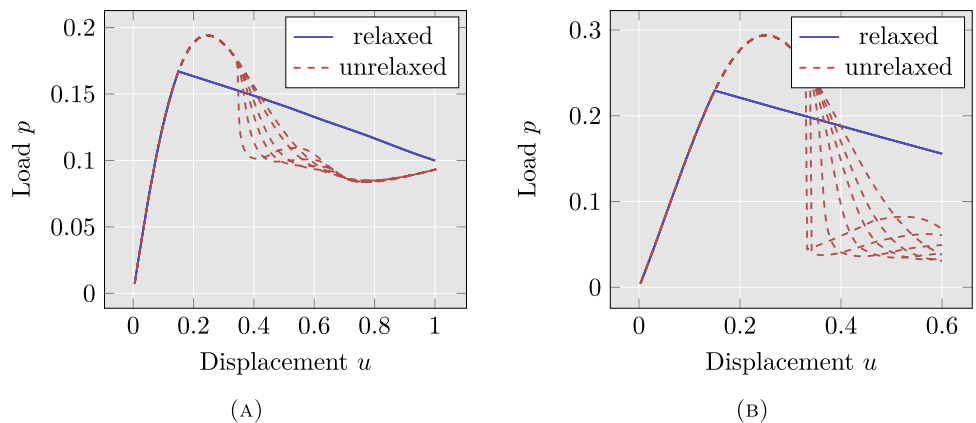
$\psi^0$	$\lambda$	$\mu$	$D_0$	$D_\infty$	$\beta_k$	$\delta_{ij}$	$F_{ii}^{\min}$	$F_{ii}^{\max}$	$F_{ij}^{\min} \forall i \neq j$	$F_{ij}^{\max} \forall i \neq j$	$ \mathcal{N} $
NH	0.5	1.0	0.3	0.9	0.0625	0.15	1.0	3.4	-0.15	0.15	3581577
STVK	0.5	1.0	0.3	0.9	0.07	0.1	0.1	2.0	-0.1	0.1	5832000

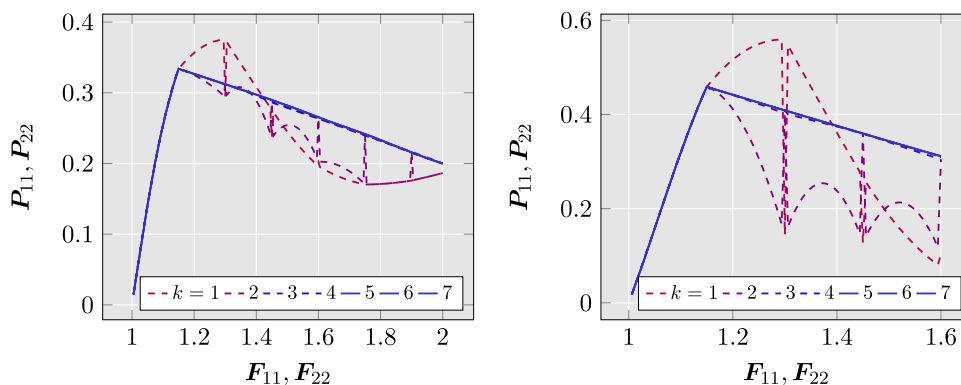


**Fig. 12** The St. Venant–Kirchhoff effective strain energy density and its associated incremental stress potential (a) and obtained rank-one convex envelope approximation (b). Again, the value  $s$  corresponds to the line characterizing equation (6.2) and a similar behavior compared

to Fig. 11 can be observed. Namely, a convex behavior along the rank-one path (depicted in c) while a lower and yet noticeable non-convexity is left along the rank-two path, shown in (d)

**Fig. 13** Biaxial force–displacement response for Neo-Hooke effective strain energy density at the left-hand side and St. Venant–Kirchhoff effective strain energy density at the right-hand side, respectively. The relaxed model shows, despite the strain softening behavior, a mesh independent response; in contrast, the unrelaxed model shows the typical mesh dependence





**Fig. 14** Lamination depth study in terms of the obtained first Piola–Kirchhoff stresses obtained by tree decomposition and evaluation. Here, the same material parameters as well as loading scenarios are evaluated as in the biaxial perturbation test. The left-hand side shows the Neo-

Hooke and the right-hand side the St. Venant–Kirchhoff effective strain energy density, respectively. The stresses converge to a smooth function for a lamination depth  $k \geq 5$  that includes strain softening

for varying  $s$  and  $d = 1, 2, 3$ . The unrelaxed energy density is dashed and the resulting lines after the higher-dimensional rank-one convexification (in two and three dimensions) and the one-dimensional convexification (in one spatial dimension) are indicated by the red, blue and green line, respectively. The qualitative behavior is similar, however the quantitative behavior is different due to the higher energy values of the deformations in higher spatial dimensions originating directly from the analytical formulation of  $W$ . This also explains the smaller non-convex regime for the unrelaxed densities in higher spatial dimensions. It is also worth mentioning that the uniaxial/one-dimensional case results in a convex function, since in this case the underlying path is a rank-one direction. The higher-dimensional hulls still show non-convexity along the rank- $d$  paths.

Altogether, this illustrates that the hull obtained in the three-dimensional case reflects the important properties, and even with the very coarse discretization of the nine-dimensional space, the convexification results in a reasonable approximation.

### 6.9 Microstructure evolution

The rank-one trees obtained in the numerical construction of the rank-one convex envelope can be utilized to give a structural interpretation on the microscale. We consider an energy density  $W$  based on the biaxial test of Sect. 6.6 in two spatial dimensions. For fixed  $F$ , Fig. 16 shows the computed rank-one trees for the first lamination orders  $k = 1, 2, 3$ , which correspond to the first three iterations of the convexification algorithm. Notice, that the tree in iteration  $k = 3$  contains major changes in comparison to the previous iteration. This is in line with the fact that the new rank-one tree is not necessarily an extension of the previous iterate tree. This is due to

the fact that a different minimum can be found when higher lamination orders are allowed.

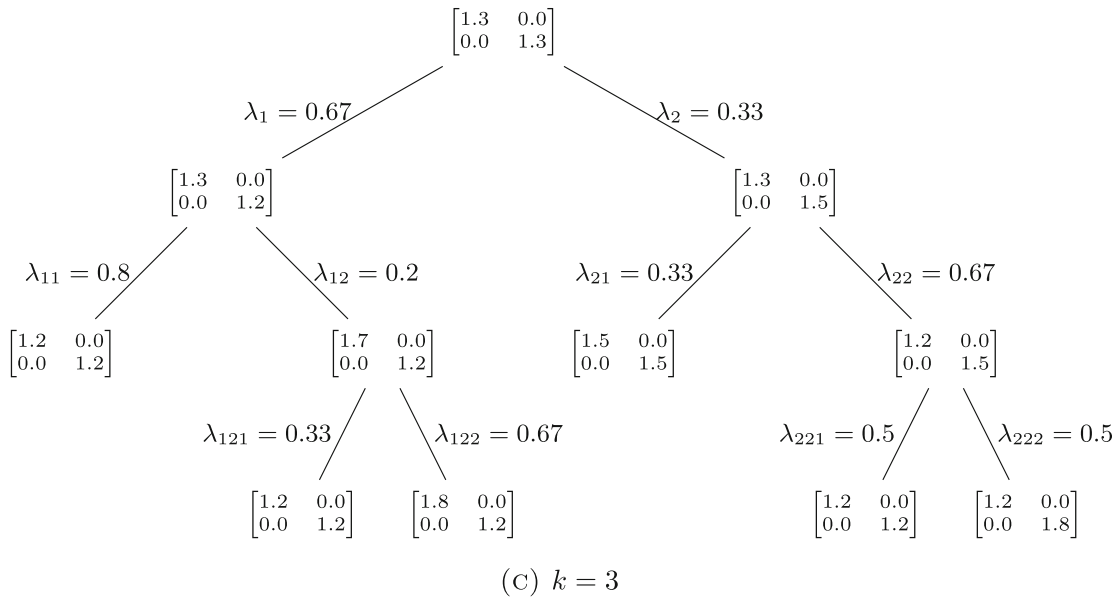
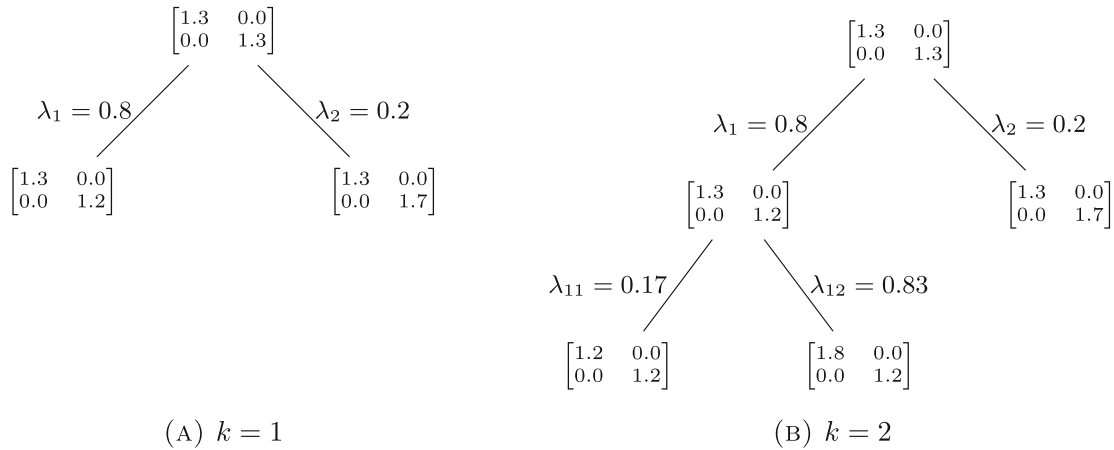
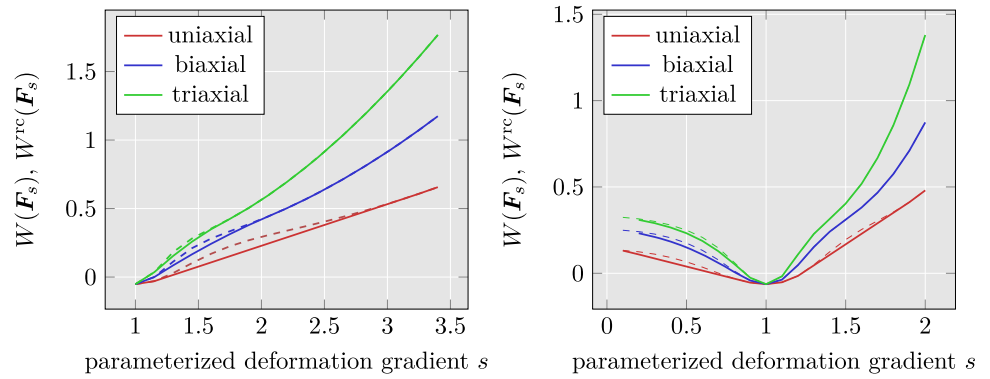
The illustrations in Fig. 16 have been simplified and only contain branching induced by lamination and not due to interpolation. The trees can be utilized to show that the leaves form an  $\mathcal{H}_M$  sequence, where  $M$  equals the number of leaves, i.e.  $M = 2$  for  $k = 2$ ,  $M = 3$  for  $k = 2$ ,  $M = 6$  for  $k = 3$ . In order to show that these trees visualize  $\mathcal{H}_M$  sequences, we have to come up with suitable  $\xi$ 's as introduced in the definition in Sect. 3. These values can be interpreted as volume fractions of the leaves in the microstructure and can be computed by the product of the convex coefficients (the  $\lambda$ 's) along the paths, e.g. in the case  $k = 2$  the volume fraction associated to the leaf node  $\text{diag}[1.2, 1.2]$  is given by  $\xi = \lambda_1 \cdot \lambda_{11} = 0.136$ . The volume fractions associated to the other two leaf nodes are given by  $\lambda_1 \cdot \lambda_{12} = 0.664$  and  $\lambda_2 = 0.2$ , respectively.

We now utilize the information encoded in the tree structure for a microstructural interpretation. For each of the trees in Fig. 16, the associated microstructure is displayed in Fig. 17. Each computed laminate is illustrated as follows. The rank-one direction  $R$ , which is the minimizing argument of the minimization problem (4.3) and connects  $F^+$  and  $F^-$ , characterizes the branching of the two (the weakly and the strongly damaged) phases. The normal vector of each lamination direction is given by the right singular vector corresponding to the only nonzero singular value of the rank-one direction matrix that is associated to the branching.

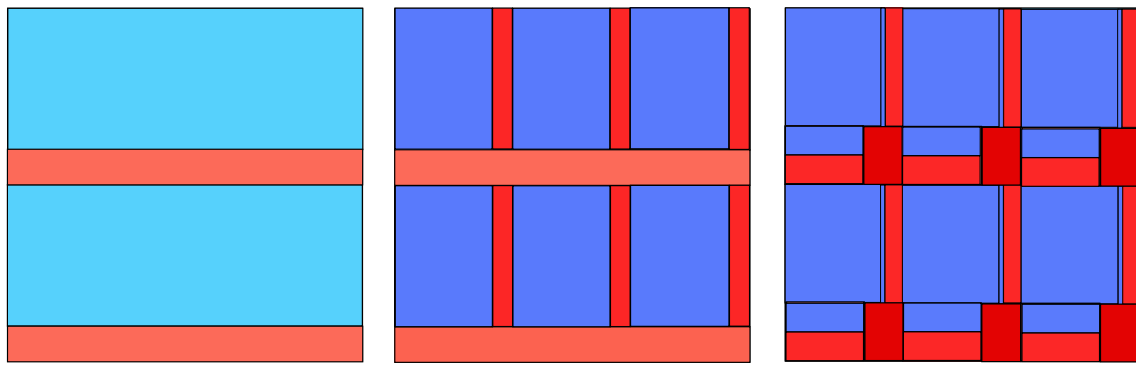
For example, the right singular vector corresponding to the non-zero singular value of the rank-one direction matrix  $\text{diag}[0, 1]$  associated to the only splitting in 16a is given by  $[0, 1]^T$ . The normal vector of this laminate is hence pointing in the  $y$ -direction.

The number of phase oscillations/phase changes in a single laminate can not be determined and is therefore only an

**Fig. 15** Uni-, bi-, triaxial tests based on the undamaged NH (left) and STVK (right) strain energy densities  $\psi^0$ . The value  $s$  refers to the characterization of the rank-one (uniaxial), rank two (biaxial) and rank three (triaxial) deformation gradient paths as parameterized in (6.3). The dashed lines describe the unrelaxed energy density  $W$  while the solid lines represent the computed rank-one hulls



**Fig. 16** Rank-one trees of the deformation gradient  $\mathbf{F} = \text{diag}[1.3, 1.3]$  for the lamination iterations  $k = 1$  (a) and  $k = 2$  (b) and  $k = 3$  (c). Shown is a special example which contains only branching subject to lamination, due to special setting where the interpolation delivers clearly one favoured nearest point



**Fig. 17** Microstructures associated to the trees in Fig. 16 for  $k = 1$  (left),  $k = 2$  (mid),  $k = 3$  (right) and the fixed deformation gradient  $\mathbf{F} = \text{diag}[1.3, 1.3]$ . Note that the microstructures are described by gradient Young measures and, thus, do not encode a length scale or frequency of the phase oscillations, respectively, but instead only the

amplitude and volume fraction of the deformation gradients. Weakly damaged phases resulting from small deformations are coloured in blue, stronger damaged phases resulting from large deformations are coloured red

exemplary realization; however, the volume fraction is given by the convex coefficients  $\lambda$  and  $1 - \lambda$  associated to the branching. Weakly damaged phases (small deformation) are coloured in blue, stronger damaged phases (large deformation) are coloured in red.

Each global iteration  $k$  of the procedure provides the possibility of branching in two phases. For the microstructure in the case  $k = 2$  (middle of Fig. 17), only the weaker phase splits into a new laminate. This time, the normal vector obtained from the non-zero right singularvector to the direction  $\text{diag}[1, 0]$  is perpendicular to the first lamination. The associated convex coefficients to this branching now influence the illustration in the weaker phase in terms of proportion but again the frequency (three phase changes) is only taken for illustration purposes.

The microstructure for the case  $k = 3$  is conceptually different again, as it was the case in Fig. 16 for the tree representation. Here, the first laminate differs in terms of proportion as well as intensity of the phases. Each of those first order laminate phases is then refined by one branching, and a second branching in one of the phases. Again the orientation and relative fraction of those phases can be inferred by the information given in the tree 16c.

Note that the volume fractions  $\xi$  of the leave nodes in each tree can be found as area ratio of the corresponding colour in the illustrated cell in 17. Since the volume fractions of all leave nodes add up to one (cf. Section 3) the mean valued colour of the three cells should match and be associated to the fixed deformation gradient  $\mathbf{F}$ .

Note that for general cases, e.g. a more complex deformation gradient, the laminates can be orientated diagonally and the resulting microstructures could be more complex.

## 7 Conclusion

In this paper, we showed that the numerical multidimensional relaxation via rank-one convexification yields mesh independent solutions for two element perturbation tests. For this purpose, a concurrent relaxation scheme was used, which was embedded into finite element simulations. To this end, efficient parallelized schemes were presented as well as reliable algorithms for the reconstruction of a continuous derivative, which is needed by the finite element solver. In addition to that, some strain softening was observed, which resulted from the non-convexity along paths with rank more than one. Along rank-one paths, the function is convex and thus the stress responses from [9, 56] are obtained. Further, we showed the necessity of high-order laminates, since only laminates of order five or higher show a converged stress response. The extension of the algorithm by [2, 4] with the reduced rank-one direction set of [26, 27] was necessary for the concurrency of the relaxation. The relaxation process allowed interpretation in terms of microstructural damage evolution. However, due to the computational costs, the presented work is still limited in terms of the applicable boundary value problems. Future work needs to reduce algorithmic complexity further, by e.g. adaptivity in mesh and direction discretization and utilization of known properties of the energy density. For the former, pioneering work has been recently presented in [11], where the given macroscopic deformation gradient is the driving refinement criterion. Due to the presented tree decomposition, a different adaptive criterion is needed, which resolves around the laminate supporting points  $\mathbf{F}^+$  and  $\mathbf{F}^-$ .

Despite the model’s capabilities to describe strain softening along paths with rank higher than one, the softening response is rather untypical for brittle materials which exhibit large deformations. Therefore, a generalization of the evolu-

tion of microstructure in the sense of [36] to higher spatial dimensions is needed, such that a realistic strain softening behavior can be captured.

**Acknowledgements** The authors acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG) within the Priority Program 2256 (“Variational Methods for Predicting Complex Phenomena in Engineering Structures and Materials”), Project ID 441154176, reference IDs BA2823/17-1, PE1464/7-1, and PE2143/5-1. Furthermore, the authors acknowledge the free and open source community of the Julia programming language, especially of Ferrite.jl.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

## Declarations

**Conflict of interest** The authors declare that they have no conflicts of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Aschan T, Holy T, Kittisopikul M, and Contributors (2022) Interpolations.jl. Julia Math
- Bartels S (2004) Linear convergence in the approximation of rank-one convex envelopes. *ESAIM Math Model Numer Anal* 38(5):811–820
- Bartels S (2005) Reliable and efficient approximation of polyconvex envelopes. *SIAM J Numer Anal* 43(1):363–385
- Bartels S (2015) Numerical methods for nonlinear partial differential equations, vol 47. Springer series in computational mathematics. Springer International Publishing, Cham
- Bazant ZP (1984) Microplane model for strain-controlled inelastic behavior. *Mech Mater*, pp 45–59
- Bartels S, Carstensen C, Hackl K, Hoppe U (2004) Effective relaxation for microstructure simulations: algorithms and applications. *Comput Methods Appl Mech Eng* 193(48–51):5143–5175
- Ball JM, Kirchheim B, Kristensen J (2000) Regularity of quasi-convex envelopes. *Calc Var Partial Differ Equ* 11(4):333–359
- Bezanson J, Karpinski S, Shah VB, Edelman A (2012) Julia: a fast dynamic language for technical computing. [arXiv:1209.5145](https://arxiv.org/abs/1209.5145)
- Balzani D, Ortiz M (2012) Relaxed incremental variational formulation for damage at large strains with application to fiber-reinforced materials and materials with truss-like microstructures. *Int J Numer Meth Eng* 92(6):551–570
- Carstensen C, Conti S, Orlando A (2008) Mixed analytical–numerical relaxation in finite single-slip crystal plasticity. *Contin Mech Thermodyn* 20(5):275
- Conti S, Dolzmann G (2018) An adaptive relaxation algorithm for multiscale problems and application to nematic elastomers. *J Mech Phys Solids* 113:126–143
- Conti S, Dolzmann G, Kreisbeck C (2015) Variational modeling of slip: from crystal plasticity to geological strata. In: Conti S, Hackl K (eds) *Analysis and computation of microstructure in finite plasticity*. Lecture notes in applied and computational mechanics. Springer International Publishing, Cham, pp 31–62
- Carlsson K, Ekre F (2019) Tensors.jl—tensor computations in Julia. *J Open Res Softw* 7(1):7
- Carlsson K, Ekre F, and Contributors (2021) Ferrite.jl. <https://github.com/Ferrite-FEM/Ferrite.jl>
- Carstensen C, Hackl K, Mielke A (2002) Non-convex potentials and microstructures in finite-strain plasticity. *Proc R Soc Lond Ser A Math Phys Eng Sci* 458(2002):299–317
- Conti S, Hauret P, Ortiz M (2007) Concurrent multiscale computing of deformation microstructure by relaxation and local enrichment with application to single-crystal plasticity. *Multiscale Model Simul* 6(1):135–157
- Ciarlet PG (1998) *Mathematical elasticity*. Number v. 20, 27, 29 in studies in mathematics and its applications. North-Holland, Sole distributors for the U.S.A. and Canada, Elsevier Science Pub. Co, Amsterdam
- Conti S (2006) Relaxation of single-slip single-crystal plasticity with linear hardening, p 6
- Carstensen C, Plecháč P (1997) Numerical solution of the scalar double-well problem allowing microstructure. *Math Comput* 66(219):997–1026
- Conti S, Theil F (2005) Single-slip elastoplastic microstructures. *Arch Ration Mech Anal* 178(1):125–148
- Dacorogna B (2008) *Direct methods in the calculus of variations*, 2nd edn. Number 78 in applied mathematical sciences. Springer, New York
- de Borst R, Crisfield MA, Remmers JJC, Verhoosel CV. *Non-linear finite element analysis of solids and structures*, p 542
- DeSimone A, Dolzmann G (2002) Macroscopic response of nematic elastomers via relaxation of a class of SO(3)-invariant energies. *Arch Ration Mech Anal* 161(3):181–204
- Dimitrijevic BJ, Hackl K (2008) A method for gradient enhancement of continuum damage models. *Tech Mech Eur J Eng Mech* 28(1):43–52
- Dimitrijevic BJ, Hackl K (2011) A regularization framework for damage–plasticity models via gradient enhancement of the free energy. *Int J Numer Methods Biomed Eng* 27(8):1199–1210
- Dolzmann G (1999) Numerical computation of rank-one convex envelopes. *SIAM J Numer Anal* 36(5):1621–1635
- Dolzmann G, Walkington NJ (2000) Estimates for numerical approximations of rank one convex envelopes. *Numer Math* 85(4):647–663
- Freund M, Ihlemann J (2010) Generalization of one-dimensional material models for the finite element method. *ZAMM J Appl Math Mech* 90(5):399–417
- Faria R, Oliver J, Cervera M (1998) A strain-based plastic viscous-damage model for massive concrete structures. *Int J Solids Struct* 35(14):1533–1558
- Gürses E, Miehe C (2011) On evolving deformation microstructures in non-convex partially damaged solids. *J Mech Phys Solids* 59(6):1268–1290
- Graham RL (1972) An efficient algorithm for determining the convex hull of a finite planar set. *Inf Process Lett* 1(4):132–133
- Hackl K (1997) Generalized standard media and variational principles in classical and finite strain elastoplasticity. *J Mech Phys Solids* 45(5):667–688
- Holzappel GA (2000) *Nonlinear solid mechanics: a continuum approach for engineering*. Wiley, Chichester

34. Junker P, Riesselmann J, Balzani D (2022) Efficient and robust numerical treatment of a gradient-enhanced damage model at large deformations. *Int J Numer Meth Eng* 123(3):774–793
35. Kachanov LM (1985) Rupture time under creep conditions. p 8
36. Köhler M, Balzani D (2023) Evolving microstructures in relaxed continuum damage mechanics for the modeling of strain softening. *J Mech Phys Solids*. <https://doi.org/10.1016/j.jmps.2023.105199>
37. Köhler M, Neumeier T, Melchior J, Peter MA, Peterseim D, Balzani D (2022) Adaptive convexification of microsphere-based incremental damage for stress and strain softening at finite strains. *Acta Mech*
38. Kohn RV, Strang G (1986) Optimal design and relaxation of variational problems. II. *Commun Pure Appl Math* 39(2):139–182
39. Kumar S, Vidyasagar A, Kochmann DM (2020) An assessment of numerical techniques to find energy-minimizing microstructures associated with nonconvex potentials. *Int J Numer Meth Eng* 121(7):1595–1628
40. Kiefer B, Waffenschmidt T, Sprave L, Menzel A (2018) A gradient-enhanced damage model coupled to plasticity—multi-surface formulation and algorithmic concepts. *Int J Damage Mech* 27(2):253–295
41. Le Dret H, Raoult A (1995) The quasiconvex envelope of the Saint Venant–Kirchhoff stored energy function. *Proc R Soc Edinb Sect Math* 125(6):1179–1192
42. Langenfeld K, Junker P, Mosler J (2018) Quasi-brittle damage modeling based on incremental energy relaxation combined with a viscous-type regularization. *Contin Mech Thermodyn* 30(5):1125–1144
43. Langenfeld K, Kurzeja P, Mosler J (2022) How regularization concepts interfere with (quasi-)brittle damage: a comparison based on a unified variational framework. *Contin Mech Thermodyn* 34(6):1517–1544
44. Miehe C, Göktepe S, Lulei F (2004) A micro-macro approach to rubber-like materials—part I: the non-affine micro-sphere model of rubber elasticity. *J Mech Phys Solids* 52(11):2617–2660
45. Miehe C (1995) Discontinuous and continuous damage evolution in Ogden-type large-strain elastic materials. *Eur J Mech A Solids*
46. Mielke A, Theil F, Levitas VI (2002) A variational formulation of rate-independent phase transformations using an extremum principle. *Arch Ration Mech Anal* 162(2):137–177
47. Murakami S (2012) Continuum damage mechanics. In: *Solid mechanics and its applications*, vol 185. Springer, Netherlands, Dordrecht
48. Nash J, Bezanson J, Pamnany K (2019) Basic threading examples in JuliaLang v1.3., p 8
49. Needleman A (1988) Material rate dependence and mesh sensitivity in localization problems. *Comput Methods Appl Mech Eng* 67(1):69–85
50. Lazy Evaluation (1998) In: Okasaki C (ed) *Purely functional data structures*. Cambridge University Press, Cambridge, pp 31–38
51. Ortiz M, Repetto E (1999) Nonconvex energy minimization and dislocation structures in ductile single crystals. *J Mech Phys Solids* 47(2):397–462
52. Ortiz M, Stainier L (1999) The variational formulation of viscoplastic constitutive updates. *Comput Methods Appl Mech Eng* 171(3):419–444
53. Peerlings RHH, Geers M, Borst R, Brekelmans WAM (2001) A critical comparison of nonlocal and gradient-enhanced softening continua. *Int J Solids Struct* 38:7723–7746
54. Pandolfi A, Ortiz M (2012) An eigenerosion approach to brittle fracture. *Int J Numer Meth Eng* 92(8):694–714
55. Riesselmann J, Balzani D (2022) A simple and efficient lagrange multiplier based mixed finite element for gradient damage
56. Schmidt T, Balzani D (2016) Relaxed incremental variational approach for the modeling of damage-induced stress hysteresis in arterial walls. *J Mech Behav Biomed Mater* 58:149–162
57. Simo JC (1987) On a fully three-dimensional finite-strain viscoelastic damage model: formulation and computational aspects. *Comput Methods Appl Mech Eng* 60(2):153–173
58. Schwarz S, Junker P, Hackl K (2020) Variational regularization of damage models based on the emulated RVE. *Contin Mech Thermodyn* 33:69–95
59. Suffis A, Lubrecht TAA, Combescure A (2003) Damage model with delay effect: analytical and numerical studies of the evolution of the characteristic damage length. *Int J Solids Struct* 40(13):3463–3476
60. Turcksin B, Kronbichler M, Bangerth W (2016) WorkStream - a design pattern for multicore-enabled finite element computations. *ACM Trans Math Softw* 43(1):1–29
61. Wingender D, Balzani D (2022) Simulation of crack propagation based on eigenerosion in brittle and ductile materials subject to finite strains. *Arch Appl Mech* 92(4):1199–1221

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.