



# Material parameter identification using finite elements with time-adaptive higher-order time integration and experimental full-field strain information

Stefan Hartmann<sup>1</sup> · Rose Rogin Gilbert<sup>1</sup>

Received: 30 September 2020 / Accepted: 20 February 2021 / Published online: 24 April 2021  
© The Author(s) 2021

## Abstract

In this article, we follow a thorough matrix presentation of material parameter identification using a least-square approach, where the model is given by non-linear finite elements, and the experimental data is provided by both force data as well as full-field strain measurement data based on digital image correlation. First, the rigorous concept of semi-discretization for the direct problem is chosen, where—in the first step—the spatial discretization yields a large system of differential-algebraic equation (DAE-system). This is solved using a time-adaptive, high-order, singly diagonally-implicit Runge–Kutta method. Second, to study the fully analytical versus fully numerical determination of the sensitivities, required in a gradient-based optimization scheme, the force determination using the Lagrange-multiplier method and the strain computation must be provided explicitly. The consideration of the strains is necessary to circumvent the influence of rigid body motions occurring in the experimental data. This is done by applying an external strain determination tool which is based on the nodal displacements of the finite element program. Third, we apply the concept of local identifiability on the entire parameter identification procedure and show its influence on the choice of the parameters of the rate-type constitutive model. As a test example, a finite strain viscoelasticity model and biaxial tensile tests applied to a rubber-like material are chosen.

**Keywords** Material parameter identification · Least-square method · Finite elements · Evolution equations · Digital image correlation · DIRK-method · DAE-system

## 1 Introduction

In solid mechanics, the mechanical behavior of a material under external loads is represented by constitutive equations. These material models depend on parameters called *material parameters*. It is necessary to calibrate the model to defined experiments in order to predict the structural behavior of components or even structures. In former times, tensile tests were carried out to ensure that more or less homogeneous stress and strain states were obtained. In this case, either the external forces of the testing machine's force gauge and the displacements of the testing machine's clamps are used to

determine stresses and strains within the specimen under consideration. Alternatively, strain gauges were used to locally determine the strains in certain regions of the specimen. Today, in cases where optical access to the specimen's surface is given, digital image correlation systems (DIC) serve as an adequate tool to detect the surface displacements in a sub-region of the entire specimen, and, accordingly, the strain field in that region—see, for example [76] or [17], and the literature cited therein. Here, one can find out whether the assumption of a homogeneous deformation state is justified, or whether inhomogeneities dominate the deformation. Moreover, it is possible to directly choose specimens providing inhomogeneous deformations. Then, however, it is necessary to solve the partial differential equations under consideration (equilibrium conditions).

To determine the parameters, which represents an inverse problem, we follow a non-linear least-square method (NLS) minimizing the difference between the model response and the experimental data. Commonly, the solution of boundary-value problems in solid mechanics is computed by means

---

This article is dedicated to the 70th birthday of Tomasz Łodygowski.

✉ Stefan Hartmann  
stefan.hartmann@tu-clausthal.de

<sup>1</sup> Institute of Applied Mechanics, Clausthal University of Technology, Adolph-Roemer-Str. 2a, 38678 Clausthal-Zellerfeld, Germany

of finite elements (FEM). A first approach of such a procedure was published by Schnur and Zabaras [70], where the goal was to detect the place and the Young's moduli of an inclusion in a matrix material. Following this concept of combining NLS and FEM, a first extension is the use of discrete force-displacement curves, see, for example [30,57,75]. An enhancement of such an approach is the application of optical information either by a grating on the specimen's surface [51], or by Moirè-patterns [47]. In regard to discrete displacement information, very few points on the specimen's surface or the contour data can serve for identification purposes as well [29]. The entire material parameter identification process using finite elements and full-field measurements data was mainly driven by the works of [2,53,54]. For a comprehensive overview see [52] as well. There, a non-linear least-square method minimizes the residual between the displacement field of the DIC-data (or manufactured simulation data) and the displacements on a sub-region of the FE-surface. Based on this approach, which employs gradient-based optimization schemes or BFGS-type approaches, further publications on different applications followed, see, for example [6,44–46,63,66].

An alternative approach to identify material parameters is provided by the virtual field method, see, for example [59], where the NLS is circumvented. In this case, the virtual displacements are chosen in such a manner that a resulting system to determine the material parameters is obtained. For an overview of different methods to identify material parameters see [3]. In [3] the procedure to apply FEM and displacement data is called *FEM-U*. A further alternative approach is discussed in [65], where the parameters are obtained by a direct approach to the DIC-data and which is compared to the NLS-approach. Apart from gradient-based methods, further gradient-free schemes are applied, for example, in [37,38] using neural networks—or [29] with a direct search method.

We call the schemes using a NLS-method together with full-field measurement by a DIC-system and simulations by means of finite elements the *NLS-FEM-DIC approach*. The fundamental difference in the applications of the NLS-FEM-DIC approach lies in the use of the material models, the calculation of the sensitivities as well as which measurement data are included in the identification. The starting point for constitutive models are either linear elastic [34], or non-linear elastic material properties. In the case of inelastic material properties, the mathematical problem changes. In this case, there has been no debate on how the NLS-FEM-DIC approach is connected to the schemes in Numerical Mathematics. Here, a connection can be drawn on to the methods compiled in [69]. This was addressed on a theoretical level in [24] by comparing the three approaches *simultaneous simulation equations* (SSE), *internal* (IND), and *external numerical differentiation* (END). In this article, we extend

the discussion in [24] to high-order time integration and a real application. For this purpose, we draw on a particular interpretation of the finite element method using material models of evolutionary-type. The computation of finite elements on the basis of constitutive equations of evolutionary-type such as models of viscoelasticity, rate-independent plasticity, and viscoplasticity turned out to be interpretable as the solution of differential-algebraic equations (DAEs) after applying the spatial discretization using finite elements, see [15,16,77]. Drawing on this interpretation, consistent algorithms to solve a system of DAEs can be applied. In [15] this is discussed for von Mises-type plasticity, and in [22] for the case of viscoelasticity, where singly diagonal-implicit Runge–Kutta (SDIRK) methods are applied. These schemes have the advantage that time-adaptivity (step-size control) is provided without any considerable additional numerical costs, and the fact that Backward–Euler like implementations to integrate the constitutive equations are embedded in the scheme ensuring that no additional work has to be done. In addition, it was found that the procedures are very efficient for long-term processes such as relaxation or creep processes. For further reading on similar approaches, see [8,39,64,67].

If the NLS-FEM-DIC approach is followed, where a gradient-based method is applied, END (numerical differentiation) is the first choice of most publications [9,55,60,65]. Although there is a large variability and flexibility of this application, the computational costs are higher than for providing analytical derivatives. Using the analytical approach to calculate the sensitivities (IND), only a very publications use this approach [40,53,66]. Here, we draw on Acegen to provide the codes containing the derivatives with respect to the material parameters [41–43]. It will be shown that this essentially reduces the computational times.

Regarding to the evaluation of the experimental data, it is very common only to evaluate the displacement data, which has the disadvantage that rigid-body motions in the experiments cannot be represented by the finite element simulation. Thus, the surface strains have to be compared, which requires knowledge about the strain computations in the FEM-simulation and the DIC-evaluation. In the case of plane problems, the strains can directly be obtained by the finite element program [9]. However, if the surface is curvilinear, most of the finite element programs cannot provide the in-plane surface strains so that a comparison to the 3D-DIC data is not possible. Here, we draw on the strain determination scheme proposed in [28]. This is applied not only on the motion of the DIC-coordinates but to the nodal coordinates of the FE surface as well. The nodes are treated in the same manner as for the DIC-data. With regard to the sensitivity computations, the derivatives of the equations with respect to the material parameters are consequently also provided.

Furthermore, it is very common not to consider reaction forces (or torques), although they can be determined

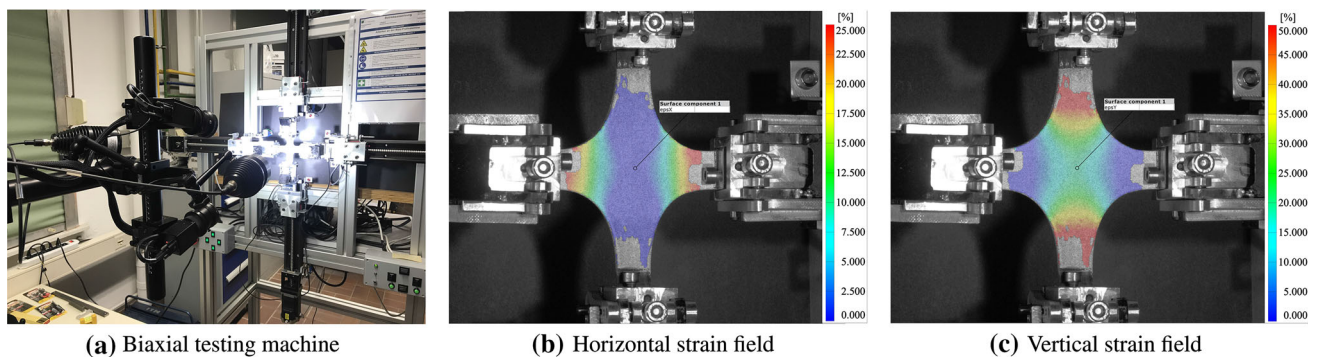


Fig. 1 Experimental setup and typical strain distributions in horizontal and vertical directions

by finite elements as well. An approach using force data is discussed in [60], where END is drawn on. However, the use of IND requires a clear description of the analytical sensitivities. This gap was filled in [24] using the Lagrange-multiplier method within the DAE-interpretation of finite elements. Although the computation of the reaction forces is frequently performed by a node-wise formulation of the equilibrium conditions, we follow the Lagrange-multiplier procedure [31]. This is necessary if other time integration schemes are applied, for example, Rosenbrock-type methods [26]. Furthermore, it is a clear variational concept for displacement control and reaction force computation, and is helpful in step-size control for considering local error estimation of the forces as well.

Moreover, most contributions do not discuss quality measures describing the resulting material parameters, such as the confidence interval, or the correlation between the parameters. Additionally, we follow the concept of local identifiability proposed in [4,7] which was studied in [25,71] in the field of solid mechanics.

If all these concepts are applied, the entire NLS-FEM-DIC concept becomes obvious. This will be demonstrated for biaxial tensile tests in this article, where, additionally, the comparison of numerical differentiation (equivalent to END) to analytical derivatives (IND) is provided. Specifically, the connection to the Multilevel-Newton algorithm—commonly applied in finite elements—is discussed [23]. Rubber is chosen as the test material.

The notation in use is defined in the following manner: geometrical vectors are symbolized by  $\vec{a}$ , second-order tensors **A** by bold-faced Roman letters, and calligraphic letters  $\mathcal{A}$  define fourth order tensors. Furthermore, we introduce matrices at global level symbolized by bold-faced italic letters **A** and matrices on local level (Gauss-point level) using bold-faced Roman letters **A**.

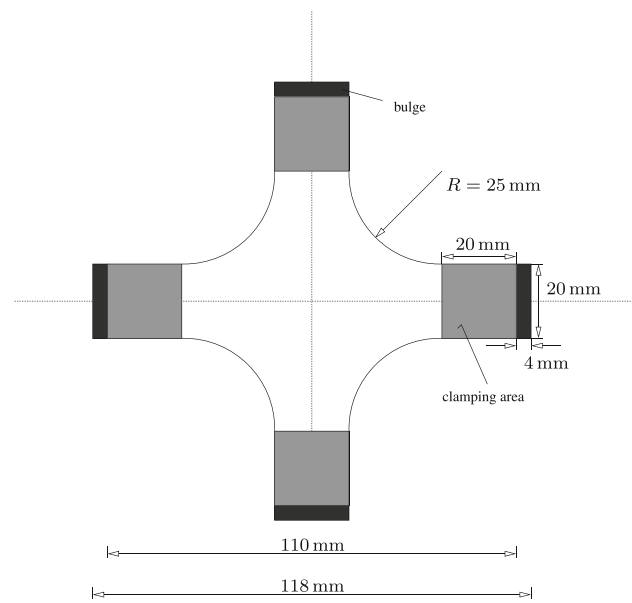


Fig. 2 Geometry of the cross-like specimen (thickness 6 mm)

## 2 Experimental data

First, we carried out experiments using a biaxial testing device, see [33] for more details and references. We chose a natural rubber according to [5]. In view of the identifiability of all material parameters, we have to circumvent equibiaxial testing paths. Fig. 1a shows the testing device, and Fig. 1b,c represent typical strain distributions that are determined by a digital image correlation system. Here, we draw on the 3D-DIC system Aramis of the company GOM, Brunswick, Germany. It turned out that rigid body motions resulting from the stiffness of the entire testing machine have significant influence on the parameter identification concept. Thus, principal strains (or stretches) are determined using the displacement data of the system. In our tests, the clamp displacements are prescribed and the resulting forces are measured and compared. The specimens under consideration are shown in Fig. 2. The bulge serves for a better fixation in the clamps.

Since we are applying a model of overstress-type viscoelasticity, particular loading paths are required to obtain the basic properties of the equilibrium stress state by the termination points of relaxation, see [35] for a discussion. We prescribe rate-dependent loading paths, see Fig. 3a, and a multi-step relaxation path, see Fig. 3b.

Horizontally, a displacement is applied which is four times larger than in the vertical direction (in order to ensure identifiability of the parameters, see the discussion in [33]). The displacement rates of the clamps are horizontally  $\dot{u}_h(t) = 0.004, 0.01, 0.1, 1 \text{ mm s}^{-1}$ . Rates slower than  $0.004 \text{ mm s}^{-1}$  were not possible by the testing machine’s capabilities. As a result, the vertical displacement rates are  $\dot{u}_v(t) = \dot{u}_h(t)/4$ . The force-displacement (displacements of the holder) results are shown in Fig. 7 together with the calibrated model.

### 3 Method of vertical lines

As discussed in the introduction, we follow the method of vertical lines, which represents a semi-discretization of solving partial differential equations (PDEs), first in space and subsequently in time, see [18,68]. In our case these are the coupled partial differential equations stemming from the local balance of linear momentum (quasi-static case) and the stress-strain relation coupled with the evolution equations for internal variables,

$$\begin{aligned} \text{Div}(\mathbf{F}(\vec{X}, t)\tilde{\mathbf{T}}(\vec{X}, t)) + \rho_R(\vec{x})\vec{k} &= \vec{0}, \\ \dot{\tilde{\mathbf{T}}}(\vec{X}, t) &= \tilde{\mathbf{h}}(\mathbf{C}(\vec{X}, t), \mathbf{q}(\vec{X}, t)), \\ \dot{\mathbf{q}}(\vec{X}, t) &= \tilde{\mathbf{r}}(\mathbf{C}(\vec{X}, t), \mathbf{q}(\vec{X}, t)). \end{aligned} \tag{1}$$

This is accompanied by initial and boundary conditions for the displacements, the tractions, and the internal variables.  $\mathbf{F}(\vec{X}, t) = \text{Grad } \vec{\chi}_R(\vec{X}, t)$  represents the deformation gradient of the motion  $\vec{\chi} = \vec{\chi}_R(\vec{X}, t)$ , where the material point with the position vector  $\vec{X}$  in the initial configuration occupies the place  $\vec{x}$  at time  $t$ .  $\mathbf{C} = \mathbf{F}^T \mathbf{F}$  is the right Cauchy-Green tensor, whereas  $\tilde{\mathbf{T}} = (\det \mathbf{F})\mathbf{F}^{-1} \mathbf{T} \mathbf{F}^{-T}$  represents the second Piola-Kirchhoff stress tensor, and  $\mathbf{T}$  the Cauchy stress tensor.  $\rho_R$  defines the density in the reference configuration, and  $\vec{k}$  is the acceleration of gravity.  $\text{Div } \mathbf{A} = \partial A_{ij} / \partial X_j \vec{e}_i$  is the divergence operator applied to a second-order tensor  $\mathbf{A}$  using the partial derivatives with respect to the coordinates in the reference configuration. Here, we draw on a model of finite strain viscoelasticity, see [35,50,62], with particular modifications. Eq. (1)<sub>2,3</sub> are dependent on internal variables  $\mathbf{q} \in \mathbb{R}^{n_q}$ , which can be scalar- or tensor-valued. In the entire procedure here, we assume that the time derivative Eq. (1)<sub>3</sub> are done on quantities operating relative to the reference configuration. Thus, variables with relative derivatives such as Oldroyd- or Jaumann derivatives have to be transformed back

to referential quantities. In the presentation here, the vector  $\mathbf{q}$  is composed of the six independent components of the (symmetric) viscous right Cauchy–Green tensor  $\mathbf{q} \leftarrow \mathbf{C}_v$ , i.e. we have  $n_q = 6$ .  $\mathbf{C}_v = \mathbf{F}^T_v \mathbf{F}_v$  stems from the multiplicative decomposition of the deformation gradient into an elastic and a viscous part,  $\mathbf{F} = \mathbf{F}_e \mathbf{F}_v$ . The second Piola–Kirchhoff tensor decomposes into three parts,

$$\begin{aligned} \tilde{\mathbf{T}} &= J \rho_R U'(J) \mathbf{C}^{-1} + \tilde{\mathbf{T}}_{\text{eq}}^{\text{iso}} + \tilde{\mathbf{T}}_{\text{ov}} \\ \text{with } U'(J) &= \frac{K}{10} (J^4 - J^{-6}). \end{aligned} \tag{2}$$

The first part is connected to the volumetric deformation  $J := \det \mathbf{F}$ , the second part represents the isochoric, equilibrium stress state, and the last term defines the overstress part. The specific strain energy function  $U(J) = K/50(J^5 + J^{-5} - 2)$  is chosen to avoid nonphysical behavior in tension and compression as it is common in most models, see the discussion in [13,27]. The isochoric equilibrium stress part reads

$$\tilde{\mathbf{T}}_{\text{eq}}^{\text{iso}} = \varphi_1 \mathbf{I} + \varphi_2 \bar{\mathbf{C}} + \varphi_3 \bar{\mathbf{C}}^{-1} \tag{3}$$

with

$$\begin{aligned} \varphi_1 &= \frac{2\rho_R}{(\det \bar{\mathbf{C}})^{1/3}} (w_1 + w_2 \text{I}_{\bar{\mathbf{C}}}), \quad \varphi_2 = -\frac{2\rho_R}{(\det \bar{\mathbf{C}})^{1/3}} w_2, \\ \varphi_3 &= -\frac{2\rho_R}{3(\det \bar{\mathbf{C}})^{1/3}} (w_1 \text{I}_{\bar{\mathbf{C}}} + 2w_2 \text{II}_{\bar{\mathbf{C}}}) \end{aligned} \tag{4}$$

and

$$w_1 = \frac{\partial w_{\text{eq}}}{\partial \text{I}_{\bar{\mathbf{C}}}} = c_{10}, \quad w_2 = \frac{\partial w_{\text{eq}}}{\partial \text{II}_{\bar{\mathbf{C}}}} = c_{01} \frac{3}{2} \text{II}_{\bar{\mathbf{C}}}^{1/2} \tag{5}$$

for

$$w_{\text{eq}}(\text{I}_{\bar{\mathbf{C}}}, \text{II}_{\bar{\mathbf{C}}}) = c_{10}(\text{I}_{\bar{\mathbf{C}}} - 3) + c_{01}(\text{II}_{\bar{\mathbf{C}}}^{3/2} - 3\sqrt{3}). \tag{6}$$

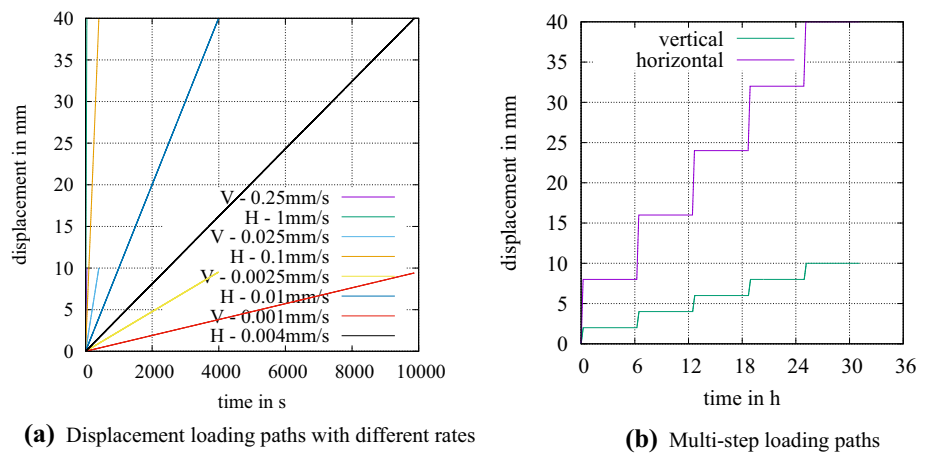
In a certain sense, this is the extension of the Mooney-Rivlin model to polyconvexity, see [27].  $\bar{\mathbf{C}} = \bar{\mathbf{F}}^T \bar{\mathbf{F}}$  defines the unimodular right Cauchy–Green tensor depending on  $\bar{\mathbf{F}} = (\det \mathbf{F})^{-1/3} \mathbf{F}$  with  $\det \bar{\mathbf{F}} = 1$ . Here, we have the invariants  $\text{I}_{\bar{\mathbf{C}}} = \text{tr } \bar{\mathbf{C}}$  and  $\text{II}_{\bar{\mathbf{C}}} = ((\text{tr } \bar{\mathbf{C}})^2 - \text{tr } \bar{\mathbf{C}}^2)/2$ . The overstress part is given by

$$\tilde{\mathbf{T}}_{\text{ov}} = 2\rho_R \mu_0 \frac{(\det \mathbf{C}_v)^{1/3}}{(\det \mathbf{C})^{1/3}} \left( \mathbf{C}^{-1}_v - \frac{1}{3} (\mathbf{C} \cdot \mathbf{C}_v^{-1}) \mathbf{C}^{-1} \right) \tag{7}$$

depending on the evolving viscous right Cauchy–Green tensor

$$\dot{\mathbf{C}}_v = \frac{4\rho_R \mu_0}{\eta} \frac{(\det \mathbf{C}_v)^{1/3}}{(\det \mathbf{C})^{1/3}} \left( \mathbf{C} - \frac{1}{3} (\mathbf{C} \cdot \mathbf{C}_v^{-1}) \mathbf{C}_v \right). \tag{8}$$

**Fig. 3** Displacement loading paths in the biaxial tensile testing machine



Particularly, a process-dependent viscosity

$$\eta = \eta_0 \exp\left(-s_0 \sqrt{\mathbf{C}\tilde{\mathbf{T}}_{ov} \cdot \tilde{\mathbf{T}}_{ov}\mathbf{C}}\right) \tag{9}$$

is necessary to reflect rate dependence. In some situations, we express the function depending on the Green-strain tensor  $\mathbf{E} = (\mathbf{C} - \mathbf{I})/2$ , which does not essentially affect the principal equations.

In this model, we have six material parameters, which are determined step-wise. The parameter vector  $\kappa^T_{eq} = \{K, c_{10}, c_{01}\}$  determines the equilibrium stress state, and  $\kappa^T_{ov} = \{\mu_0, \eta_0, s_0\}$  controls the overstress behavior. Here, we follow the common procedure of connecting the density  $\rho_R$  with the material parameters  $K \leftarrow K\rho_R, c_{10} \leftarrow c_{10}\rho_R, c_{01} \leftarrow c_{01}\rho_R$ , and  $\mu_0 \leftarrow \mu_0\rho_R$ . Obviously, Eq. (2) represents the concrete formulation of Eq. (1)<sub>2</sub>, and Eq. (8) the evolution Eq. (1)<sub>3</sub>.

Equation (1)<sub>1</sub> is transferred into a weak form represented by either the classical principle of virtual displacements or by a mixed formulation, for example a three-field formulation as proposed in [74]. Both weak formulations have the disadvantage that the calculation of the reaction forces at those degrees of freedom, where displacements are prescribed, is not possible (they do not produce a virtual work). Of course, local equilibrium formulations intuitively lead to results, but it is not embedded in the theory itself. Furthermore, no consistent mathematical description is provided. This can be circumvented by analytical considerations regarding the Lagrange multiplier method, see [31], leading to the DAE-system

$$\mathbf{F}(t, \mathbf{y}(t), \dot{\mathbf{y}}(t)) := \begin{Bmatrix} \mathbf{g}_a(t, \mathbf{u}_a(t), \mathbf{q}(t)) + \mathbf{M}\boldsymbol{\lambda}(t) \\ \mathbf{C}_c(t, \mathbf{u}_a(t)) \\ \dot{\mathbf{q}}(t) - \mathbf{r}(\mathbf{u}_a(t), \mathbf{q}(t)) \end{Bmatrix} = \mathbf{0}. \tag{10}$$

Here, we draw on the abbreviation of the unknowns  $\mathbf{y}^T(t) = \{\mathbf{u}_a^T(t), \boldsymbol{\lambda}^T(t), \mathbf{q}^T(t)\}$  and their initial conditions

$$\mathbf{y}(t_0) := \begin{Bmatrix} \mathbf{u}_a(t_0) \\ \boldsymbol{\lambda}(t_0) \\ \mathbf{q}(t_0) \end{Bmatrix} = \begin{Bmatrix} \mathbf{u}_{a0} \\ \boldsymbol{\lambda}_0 \\ \mathbf{q}_0 \end{Bmatrix} := \mathbf{y}_0. \tag{11}$$

$\mathbf{u}_a \in \mathbb{R}^{n_u+n_p}$  consists of the unknown nodal displacements  $\mathbf{u} \in \mathbb{R}^{n_u}$ , where forces (or surface tractions) are applied, and of the degrees of freedom  $\hat{\mathbf{u}}$ , where the known displacements  $\bar{\mathbf{u}} \in \mathbb{R}^{n_p}$  are given. In the Lagrange multiplier approach to obtain the reaction forces, it is assumed that all nodal displacements  $\mathbf{u}_a^T = \{\mathbf{u}^T, \hat{\mathbf{u}}^T\}$  are unknown. This implies the constraint equation

$$\begin{aligned} \mathbf{C}_c(\mathbf{u}_a) = \hat{\mathbf{u}} - \bar{\mathbf{u}} = \mathbf{M}^T \mathbf{u}_a - \bar{\mathbf{u}} = \mathbf{0}, \quad \text{with} \\ \mathbf{M} = \begin{bmatrix} \mathbf{0}_{n_u \times n_p} \\ \mathbf{1}_{n_p} \end{bmatrix}. \end{aligned} \tag{12}$$

The displacements  $\bar{\mathbf{u}}$  are assigned by the incidence matrix  $\mathbf{M} \in \mathbb{R}^{(n_u+n_p) \times n_p}$  to the unknown nodal displacements  $\mathbf{u}_a$ . Further,  $\boldsymbol{\lambda} \in \mathbb{R}^{n_p}$  represents the Lagrange multiplier vector and is interpreted as the vector containing the reaction forces, which are necessary to enforce the prescribed displacements  $\bar{\mathbf{u}}$ .

$$\begin{aligned} \mathbf{q}(t) &= \sum_{e=1}^{n_{el}} \sum_{j=1}^{n_{GP}^e} \mathbf{z}_q^{e(j)T} \mathbf{q}^{e(j)}(t), \\ \text{or } \mathbf{q}^{e(j)}(t) &= \mathbf{z}_q^{e(j)} \mathbf{q}(t) \end{aligned} \tag{13}$$

contains all internal variables,  $\mathbf{q} \in \mathbb{R}^{n_Q}$ , evaluated at all  $n_G$  spatial integration points, which are usually the Gauss-points.  $n_{el}$  is the number of elements, and  $n_{GP}^e$  represents the number of Gauss-points within element  $e$ .  $n_Q = (\sum_{e=1}^{n_{el}} n_{GP}^e) \times n_q$  defines all internal variables of the entire mesh. In this sense,

the “global” ODEs

$$r(t, \mathbf{u}_a, \mathbf{q}) = \sum_{e=1}^{n_{el}} \sum_{j=1}^{n_{GP}^e} \mathbf{Z}_q^{e(j)T} \mathbf{r}(\mathbf{c}^{e(j)}, \mathbf{q}^{e(j)}) \tag{14}$$

hold as well. The vector  $\mathbf{g}_a \in \mathbb{R}^{n_u+n_p}$  contains all equations resulting from the weak formulation

$$\mathbf{g}_a(t, \mathbf{u}_a, \mathbf{q}) = \begin{Bmatrix} \mathbf{g}(\mathbf{y}) \\ \bar{\mathbf{g}}(\mathbf{y}) \end{Bmatrix} = \begin{Bmatrix} \sum_{e=1}^{n_{el}} \mathbf{Z}^e T \left\{ \sum_{j=1}^{n_{GP}^e} w^{e(j)} \tilde{\mathbf{B}}^{e(j)T} \tilde{\mathbf{h}}(\mathbf{c}^{e(j)}, \mathbf{q}^{e(j)}) \det \mathbf{J}^{e(j)} \right\} - \bar{\mathbf{p}}(t) \\ \sum_{e=1}^{n_{el}} \bar{\mathbf{Z}}^e T \left\{ \sum_{j=1}^{n_{GP}^e} w^{e(j)} \tilde{\mathbf{B}}^{e(j)T} \tilde{\mathbf{h}}(\mathbf{c}^{e(j)}, \mathbf{q}^{e(j)}) \det \mathbf{J}^{e(j)} \right\} \end{Bmatrix} \tag{15}$$

with the incidence matrices  $\mathbf{Z}^e \in \mathbb{R}^{n_u^e \times n_u}$  and  $\bar{\mathbf{Z}}^e \in \mathbb{R}^{n_p^e \times n_p}$  assembling all element contributions into a large system of equations. They are not explicitly programmed, and they represent the assembling procedure. They contain only the numbers 1 and 0.  $\mathbf{c}^{e(j)} \in \mathbb{R}^6$  represents the column vector representation of the symmetric part of the right Cauchy–Green tensor (Voigt-notation) which depends non-linearly on the element nodal displacements  $\mathbf{u}^e \in \mathbb{R}^{n_u^e}$ ,  $\mathbf{c}^{e(j)} = \hat{\mathbf{C}}^{e(j)}(\mathbf{u}^e)$ , with  $\mathbf{u}^e = \mathbf{Z}^e \mathbf{u} + \bar{\mathbf{Z}}^e \hat{\mathbf{u}}$ .  $n_u^e$  is the number of element nodal displacement degrees of freedom,  $w^{e(j)}$  the weighting factors of the Gauss-integration in an element,  $n_{GP}^e$  are the number of Gauss-points within one element, and  $\tilde{\mathbf{B}}^{e(j)} \in \mathbb{R}^{6 \times n_u^e}$  defines the strain-displacement matrix of element  $e$  evaluated at the  $j$ -th Gauss-point,  $j = 1, \dots, n_{GP}^e$ , which depends on the element nodal displacements  $\mathbf{u}^e$  as well. The Gauss points have the local coordinates  $\xi^{(j)} \in \mathbb{R}^3$  (we only consider three-dimensional continua). Furthermore,  $\mathbf{J}^{e(j)} \in \mathbb{R}^{3 \times 3}$  symbolizes the Jacobi-matrix of the coordinate transformation between reference element coordinates and global coordinates, and  $\bar{\mathbf{p}}(t) \in \mathbb{R}^{n_u}$  defines the given equivalent nodal force vector. The symmetric part of the stress tensor (1)<sub>2</sub> is transferred into a vector  $\tilde{\mathbf{T}} \in \mathbb{R}^6$ ,  $\tilde{\mathbf{T}} = \tilde{\mathbf{h}}(\mathbf{c}^{e(j)}, \mathbf{q}^{e(j)})$ , which is evaluated at Gauss-point  $\xi^{(j)}$ , and depends due to the right Cauchy–Green tensor on the displacements and the internal variables at that Gauss point.

In the case of elasticity, where the equations have to be evaluated in the absence of internal variables, the DAE-system (10) degenerates to the system of non-linear equations

$$\mathbf{F}(t, \mathbf{y}(t)) := \begin{Bmatrix} \mathbf{g}_a(t, \mathbf{u}_a(t)) + \mathbf{M}\lambda(t) \\ \mathbf{C}_c(t, \mathbf{u}_a(t)) \end{Bmatrix} = \mathbf{0}, \tag{16}$$

where  $t$  stands for a parameter similar to a generalized continuation method. Here, the non-linear system (16) has to be fulfilled at each loading step  $t$ .

In implicit finite element approaches, the DAE-system (10) is often solved using a Backward–Euler method. This yields the non-linear system

$$\begin{aligned} \mathbf{g}(t_{n+1}, \mathbf{u}_{n+1}, \hat{\mathbf{u}}_{n+1}, \mathbf{q}_{n+1}) &= \mathbf{0}, \\ \bar{\mathbf{g}}(\mathbf{u}_{n+1}, \hat{\mathbf{u}}_{n+1}, \mathbf{q}_{n+1}) + \lambda_{n+1} &= \mathbf{0}, \\ \mathbf{C}_c(t_{n+1}, \hat{\mathbf{u}}_{n+1}) &= \mathbf{0}, \\ \mathbf{l}(\mathbf{u}_{n+1}, \hat{\mathbf{u}}_{n+1}, \mathbf{q}_{n+1}) &= \mathbf{0}, \end{aligned} \tag{17}$$

with

$$\begin{aligned} \mathbf{l}(\mathbf{u}_{n+1}, \hat{\mathbf{u}}_{n+1}, \mathbf{q}_{n+1}) \\ = \mathbf{q}_{n+1} - \mathbf{q}_n - \Delta t_n \mathbf{r}(\mathbf{u}_{n+1}, \hat{\mathbf{u}}_{n+1}, \mathbf{q}_{n+1}) \end{aligned} \tag{18}$$

at each point in time  $t_{n+1}$ ,  $0 \leq t_{n+1} \leq T$ ,  $t_{n+1} = t_n + \Delta t_n$ , where  $\Delta t_n$  is the step-size. In the final iterated solution,  $\hat{\mathbf{u}} = \bar{\mathbf{u}}$  is reached—which is not necessarily the case in other integration schemes, see [26]. In this case, the system (17) can be written as ( $\hat{\mathbf{u}}$  is assumed to be known)

$$\begin{aligned} \mathbf{g}(t_{n+1}, \mathbf{u}_{n+1}, \mathbf{q}_{n+1}) &= \mathbf{0}, \\ \bar{\mathbf{g}}(\mathbf{u}_{n+1}, \mathbf{q}_{n+1}) + \lambda_{n+1} &= \mathbf{0}, \\ \mathbf{l}(\mathbf{u}_{n+1}, \mathbf{q}_{n+1}) &= \mathbf{0}. \end{aligned} \tag{19}$$

Equations (19)<sub>1,3</sub> are solved using the Multilevel-Newton algorithm, and Eq. (19)<sub>2</sub> represents a downstream step since it is explicit in  $\lambda_{n+1}$ . Regarding the Multilevel-Newton algorithm, see [61] for the original scheme, and [15,23] for the discussion in finite elements. Since we assume a DAE-system with consistent initial conditions, the non-linear system (19) is fulfilled for the initial conditions at time  $t_0$ .

The same algorithmic structure is given by diagonal-implicit Runge–Kutta methods (DIRK-methods), where at each stage  $T_{ni} = t_n + c_i \Delta t_n$ ,  $\Delta t_n = t_{n+1} - t_n$ , the non-linear system

$$\begin{aligned} \mathbf{g}(T_{ni}, \mathbf{u}_{ni}, \hat{\mathbf{u}}_{ni}, \mathbf{q}_{ni}) &= \mathbf{0}, \\ \bar{\mathbf{g}}(\mathbf{u}_{ni}, \hat{\mathbf{u}}_{ni}, \mathbf{q}_{ni}) + \lambda_{ni} &= \mathbf{0}, \\ \mathbf{C}_c(T_{ni}, \hat{\mathbf{u}}_{ni}) &= \mathbf{0}, \\ \mathbf{l}(\mathbf{u}_{ni}, \hat{\mathbf{u}}_{ni}, \mathbf{q}_{ni}) &= \mathbf{0}, \end{aligned} \tag{20}$$

with

$$\mathbf{l}(\mathbf{u}_{ni}, \hat{\mathbf{u}}_{ni}, \mathbf{q}_{ni}) = \mathbf{q}_{ni} - \mathbf{S}_{ni}^q - \Delta t_n a_{ii} \mathbf{r}(\mathbf{u}_{ni}, \hat{\mathbf{u}}_{ni}, \mathbf{q}_{ni}) \tag{21}$$

has to be solved.  $c_i$ ,  $i = 1, \dots, s$ , and  $a_{ij}$  ( $a_{ij} = 0$  for  $j < i$ ), are the coefficients of the Butcher array containing the given weighting factors of the method under consideration, see [19,20]. In the final solution, condition (20)<sub>3</sub> is fulfilled

so that we can simplify system (20) to

$$\begin{aligned} \mathbf{g}(T_{ni}, \mathbf{U}_{ni}, \mathbf{Q}_{ni}) &= \mathbf{0}, \\ \bar{\mathbf{g}}(T_{ni}, \mathbf{U}_{ni}, \mathbf{Q}_{ni}) + \mathbf{\Lambda}_{ni} &= \mathbf{0}, \\ \mathbf{l}(T_{ni}, \mathbf{U}_{ni}, \mathbf{Q}_{ni}) &= \mathbf{0}. \end{aligned} \tag{22}$$

The stage quantities are used to determine the stage derivatives

$$\begin{aligned} \dot{\mathbf{U}}_{ni} &= \frac{\mathbf{U}_{ni} - \mathbf{S}_{ni}^u}{\Delta t_n a_{ii}}, & \dot{\mathbf{Q}}_{ni} &= \frac{\mathbf{Q}_{ni} - \mathbf{S}_{ni}^q}{\Delta t_n a_{ii}}, \\ \dot{\mathbf{\Lambda}}_{ni} &= \frac{\mathbf{\Lambda}_{ni} - \mathbf{S}_{ni}^\lambda}{\Delta t_n a_{ii}}, \end{aligned} \tag{23}$$

required for the starting vectors

$$\begin{aligned} \mathbf{S}_{ni}^u &= \mathbf{u}_n + \Delta t_n \sum_{j=1}^{i-1} a_{ij} \dot{\mathbf{U}}_{nj}, \\ \mathbf{S}_{ni}^q &= \mathbf{q}_n + \Delta t_n \sum_{j=1}^{i-1} a_{ij} \dot{\mathbf{Q}}_{nj}, \\ \mathbf{S}_{ni}^\lambda &= \mathbf{\Lambda}_n + \Delta t_n \sum_{j=1}^{i-1} a_{ij} \dot{\mathbf{\Lambda}}_{nj}. \end{aligned} \tag{24}$$

The unknown stage quantities are the nodal displacements  $\mathbf{U}_{ni}$ , the reaction forces  $\mathbf{\Lambda}_{ni}$  (negative Lagrange multipliers), and all internal variables  $\mathbf{Q}_{ni}$  from all Gauss-points in the structure. Alternatively, it is possible to apply a rate formulation in which the stage-derivatives are the unknowns. We choose the first version to obtain the same implementation as in “classical” implicit finite elements.

The final values at time  $t_{n+1}$  are

$$\begin{aligned} \mathbf{u}_{n+1} &= \mathbf{u}_n + \Delta t_n \sum_{i=1}^s b_i \dot{\mathbf{U}}_{ni}, \\ \mathbf{q}_{n+1} &= \mathbf{q}_n + \Delta t_n \sum_{i=1}^s b_i \dot{\mathbf{Q}}_{ni}, \\ \mathbf{\lambda}_{n+1} &= \mathbf{\lambda}_n + \Delta t_n \sum_{i=1}^s b_i \dot{\mathbf{\Lambda}}_{ni}, \end{aligned} \tag{25}$$

where the  $b_i, i = 1, \dots, s$ , are additional weighting factors of the Butcher array. For stiffly accurate methods, the condition  $a_{sj} = b_j$  holds—so that the stage quantities at the  $s$ -stage are identical to the final result,  $\mathbf{u}_{n+1} = \mathbf{u}_{ns}$ ,  $\mathbf{q}_{n+1} = \mathbf{q}_{ns}$ , and  $\mathbf{\lambda}_{n+1} = \mathbf{\lambda}_{ns}$ .

For  $s = 1$  (one stage),  $c_1 = b_1 = a_{11} = 1$ , the Backward–Euler approach is embedded in the more general DIRK-methods. The advantage of the DIRK-methods is that step-size selection is obtained for very few extra

computations, see [15], which is necessary for physical problems with different local time-scales. This has clear advantages with regard to computational, especially in relaxation or creep dominated problems. The higher order of the methods yields larger time steps for the same accuracy as in the Backward–Euler approach. This has been demonstrated for problems in crystal plasticity, viscoplasticity, diffusion-driven mechanics, thermo-mechanical or electro-thermo-mechanical coupling, curing processes, and thermal fluid-structure interaction. According to our experience, a two-stage, second-order method is sufficient for more accurate solutions and time-adaptivity.

In the case of non-linear elastic problems (absence of ordinary-differential part), the entire approach can be interpreted as a continuation method, i.e. the Newton-Raphson scheme is applied to

$$\begin{aligned} \mathbf{g}(t_{n+1}, \mathbf{u}_{n+1}) &= \mathbf{0} \\ \boldsymbol{\lambda}_{n+1} &= -\bar{\mathbf{g}}(t_{n+1}, \mathbf{u}_{n+1}) \end{aligned} \tag{26}$$

which has to be fulfilled at each time  $t_{n+1}$ , see Eq. (16). The starting (guess) vector of the Newton-Raphson method is commonly chosen to be the previous solution  $\mathbf{u}_{n+1}$ , or some estimation, see [32]. The latter estimation is strongly recommended (also for the problem (20) for some parts of the non-linear system) since it essentially stabilizes the computations.

## 4 Least-square approach

In the following, the non-linear least-square problem is discussed in detail. First, the basic NLS is recapped. Then the fully analytical computation of the sensitivity matrices is explained, and the numerical differentiation technique is summarized. Since we are interested in investigating quality measures, which is also connected to the concept of local identifiability for the identified material parameters, this is explained afterwards.

### 4.1 Basic problem

Least-square methods minimize the square of the distance between the model and the experimental data. Before this is explained in more detail, the experimental data has to be discussed. In the case of full-field measurement, we obtain from each experiment  $E, E = 1, \dots, n_{\text{exp}}$ , a data vector  $\mathbf{d}^{(E)} \in \mathbb{R}^{n_{\text{exp}}^{(E)}}$ . For  $n_N^{(E)}$  (temporal) load-steps, each load-step consists of  $n_d^{(E)}$  entries (discrete spatially distributed displacements or stretches, forces concerned, ...), i.e.  $n_{\text{exp}}^{(E)} = n_N^{(E)} n_d^{(E)}$ . The experimental evaluation times are  $t_m, m = 1, \dots, n_N^{(E)}$ . These data vectors are assembled

into the vector  $\mathbf{d}^{(E)T} = \{\mathbf{d}_0^{(E)T}, \mathbf{d}_1^{(E)T}, \dots, \mathbf{d}_{n_N^{(E)}}^{(E)T}\}$ , where  $\mathbf{d}_m^{(E)} \in \mathbb{R}^{n_d^{(E)}}$  contains the information from one experiment. For all tests under consideration, this leads to the entire data vector  $\mathbf{d}^T = \{\mathbf{d}^{(1)T}, \mathbf{d}^{(2)T}, \dots, \mathbf{d}^{(n_{\text{exp}})T}\}$ ,  $\mathbf{d} \in \mathbb{R}^{n_D}$ , with  $n_D = \sum_{E=1}^{n_{\text{exp}}} n_{\text{exp}}^{(E)}$ .

For each experiment  $E$ , the finite element model provides displacements ( $\mathbf{u}_n^{(E)}$  and  $\bar{\mathbf{u}}_n^{(E)}$ ), and reaction forces  $\lambda_n^{(E)}$  for the temporal points  $t_n^{(E)}$ ,  $n = 1, \dots, N^{(E)}$ . Both the temporal and the spatial points of the experiment and the finite element model do not coincide. Thus, a temporal and a spatial interpolation of the model data to the experimental data have to be carried out. We choose the (temporal) linear interpolation of the experimental data to the model time data since the sensitivities of the simulations cannot be interpolated. In this sense, the size of the data vectors  $\mathbf{d}$  adapts to the information from the finite element computations. In the spatially distributed data, there are several aspects to be considered. Firstly, a DIC-system describes only surface data on a sub-region of the finite element mesh. Thus, only a subset of model data has to be compared. Secondly, a (commercial) DIC-system can provide both the motion of material points, accordingly the displacements of the material points, or, by a black-box interpolation of the DIC-evaluation program, strain data at these points. This statement holds for commercial finite element programs as well. It has turned out in several applications that rigid-body motions in the experimental tests—resulting from the compliance of the testing machine—essentially influence the comparability of both data sets. Although there are algorithmic possibilities to minimize the rigid-body motions in the DIC-data, this does not help in all cases. Thus, the strain data evaluation is—to our experience—of greater interest. In this case, however (and, in view of a fully analytical sensitivity analysis, see Sect. 4.2), the strain evaluation procedure has to be known. Thus, we follow the interpolation concept proposed in [28] based on triangulation, see [36,58] for a similar approach. This concept can be applied to both the DIC-data as well as to the finite element nodal displacement data. Thus, the same interpolation scheme and strain evaluation is applied for both systems (DIC and FEM). Moreover, we can compute at any point of the DIC-region and the finite element model displacements and strains lead to a considerable flexibility in the evaluation. Finally, the forces, which are commonly recorded as well, should be considered in the identification process since we are interested in adapting stress-strain models requiring some force information.

As we discussed before, only a subset  $\tilde{\mathbf{u}}_n^{(E)} = \tilde{\mathbf{M}}^{(E)} \mathbf{u}_n^{(E)}$ ,  $\tilde{\mathbf{u}}_n^{(E)} \in \mathbb{R}^{n_u^{(E)}}$  of the finite element nodal displacements  $\mathbf{u}_n^{(E)} \in \mathbb{R}^{n_u^{(E)}}$  can be compared to the DIC-data. If in-plane strains or stretches are considered, an interpolation scheme is required. Instead of the common letter  $\lambda$  for a stretch, we

take  $\nu$  to circumvent a misinterpretation with the Lagrange-multiplier  $\lambda$ . In this case the simulation component

$$s_k(\boldsymbol{\kappa}) = \nu_k(\tilde{\mathbf{u}}_n(\boldsymbol{\kappa})) \quad (27)$$

is indirectly dependent on the parameter set  $\boldsymbol{\kappa}$ . In the case of a principal strain measure, it reads  $\varepsilon_k(\nu_k(\tilde{\mathbf{u}}_n(\boldsymbol{\kappa})))$ .

In order to extract the required nodal displacements, the incidence matrix  $\tilde{\mathbf{M}}^{(E)}$  is introduced. On the side of the forces—in our application of a biaxial tensile test we have two,  $F_{\text{FEM}1n}^{(E)} = -\tilde{\mathbf{M}}_1^{(E)T} \lambda_n$  and  $F_{\text{FEM}2n}^{(E)} = -\tilde{\mathbf{M}}_2^{(E)T} \lambda_n$ . The vectors  $\tilde{\mathbf{M}}_k^{(E)} \in \mathbb{R}^{n_p^{(k)}}$ ,  $k = 1, 2$ , are chosen to extract the nodal reaction forces required to determine the scalar values. The indices 1, 2 indicate the horizontal and the vertical directions.

In conclusion, we have the entire vector  $\mathbf{d}$  of all experimental data with different physical properties (e.g. forces, displacements, strains, ...), and the finite element result  $\mathbf{s}(\boldsymbol{\kappa})$  depending on the parameter set  $\boldsymbol{\kappa} \in \mathbb{R}^{n_\kappa}$ . Since different physical quantities and different amounts of data are available (few forces and a huge amount of displacement-based data), a weighting technique  $\tilde{\mathbf{r}}(\boldsymbol{\kappa}) = \mathbf{W}\mathbf{r}(\boldsymbol{\kappa}) = \mathbf{W}\{\mathbf{s}(\boldsymbol{\kappa}) - \mathbf{d}\}$  is applied to the residual  $\mathbf{r}(\boldsymbol{\kappa}) = \mathbf{s}(\boldsymbol{\kappa}) - \mathbf{d}$ . Here, we draw on the concepts proposed in [21]. The entire weighting matrix reads

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}^{(1)} & & \\ & \ddots & \\ & & \mathbf{W}^{(n_{\text{exp}})} \end{bmatrix} \quad \text{with} \quad \mathbf{W}^{(E)} = \begin{bmatrix} \mathbf{W}_0^{(E)} & & \\ & \ddots & \\ & & \mathbf{W}_{n_N^{(E)}}^{(E)} \end{bmatrix} \quad (28)$$

Within the weighting matrix

$$\mathbf{W}_n^{(E)} = \begin{bmatrix} w_F^{(E)} & & \\ & w_F^{(E)} & \\ & & w_\varepsilon^{(E)} \mathbf{I} \end{bmatrix}, \quad (29)$$

we have the same weighting factors

$$w_F^{(E)} = \frac{1}{\max_{n=1, \dots, n_N^{(E)}} |F_n^{(E)}|} \quad (30)$$

for the force data in vertical and horizontal direction of the biaxial tensile tests. The weighting factors of the strain data



are defined by

$$w_\varepsilon^{(E)} = \frac{1}{\left( \|\boldsymbol{\varepsilon}_n^{(E)}\|_{\max_{n=1,\dots,n_N^{(E)}}} \right)^{n_d^{(E)}}} \tag{31}$$

Now, the NLS-problem reads

$$\begin{aligned} f(\boldsymbol{\kappa}) &= \frac{1}{2} \{ \mathbf{W}r(\boldsymbol{\kappa}) \}^T \mathbf{W}r(\boldsymbol{\kappa}) \\ &= \frac{1}{2} \{ \mathbf{W}\{\mathbf{s}(\boldsymbol{\kappa}) - \mathbf{d}\} \}^T \{ \mathbf{W}\{\mathbf{s}(\boldsymbol{\kappa}) - \mathbf{d}\} \} \rightarrow \min \end{aligned} \tag{32}$$

i.e. the square of the weighted residual should be a minimum, and the necessary condition for the minimum in the solution  $\boldsymbol{\kappa} = \boldsymbol{\kappa}^*$  is given by

$$\mathbf{F}(\boldsymbol{\kappa}) = \frac{df}{d\boldsymbol{\kappa}} = \mathbf{D}^T(\boldsymbol{\kappa}) \mathbf{W}^T \mathbf{W} \{ \mathbf{s}(\boldsymbol{\kappa}) - \mathbf{d} \} = \mathbf{0}, \rightarrow \boldsymbol{\kappa}^* \tag{33}$$

representing a system of non-linear equations to determine the material parameters  $\boldsymbol{\kappa}^*$ . No inequalities were necessary in the applications of this article.

Here, the functional matrix

$$\mathbf{D}(\boldsymbol{\kappa}) = \frac{d\tilde{\mathbf{r}}(\boldsymbol{\kappa})}{d\boldsymbol{\kappa}} = \frac{d\mathbf{s}(\boldsymbol{\kappa})}{d\boldsymbol{\kappa}}, \tag{34}$$

$\mathbf{D} \in \mathbb{R}^{n_D \times n_\kappa}$ , is required, which is frequently called the *sensitivity matrix* (or, in short, *sensitivity*). This holds for Gauss-Newton-like algorithms as well [48,56,69]. In this respect, we need the derivatives of both the resulting forces as well as the displacements with respect to the parameter vector  $\boldsymbol{\kappa}$ ,

$$\begin{aligned} \frac{dF_{\text{FEM}n}^{(E)}(\boldsymbol{\kappa})}{d\boldsymbol{\kappa}} &= -\tilde{\mathbf{M}}^{(E)T} \frac{d\boldsymbol{\lambda}_n^{(E)}(\boldsymbol{\kappa})}{d\boldsymbol{\kappa}}, \\ \frac{d\tilde{\mathbf{u}}_n^{(E)}(\boldsymbol{\kappa})}{d\boldsymbol{\kappa}} &= \tilde{\mathbf{M}}^{(E)} \frac{d\mathbf{u}_n^{(E)}(\boldsymbol{\kappa})}{d\boldsymbol{\kappa}}. \end{aligned} \tag{35}$$

The derivatives of the principal strains (or stretches) with respect to the parameters  $\boldsymbol{\kappa}$  are obtained using the chain-rule, e.g.

$$\frac{d\varepsilon_k}{d\boldsymbol{\kappa}} = \frac{d\varepsilon_{kn}}{dv_{kn}} \left\{ \frac{dv_{kn}}{d\tilde{\mathbf{u}}_n} \right\}^T \left[ \frac{d\tilde{\mathbf{u}}_n}{d\boldsymbol{\kappa}} \right], \tag{36}$$

see Eq. (35)<sub>2</sub>, see ‘‘Appendix A’’. For the case of overstress-type models, the equilibrium stress part is identified first, i.e. we need the sensitivities of the parameterized non-linear system (26). Since the derivatives are defined implicitly, as

the solution of the non-linear system (26) depends on the parameters  $\boldsymbol{\kappa}$ , the system reads

$$\begin{aligned} \mathbf{g}(t_{n+1}, \mathbf{u}_{n+1}(\boldsymbol{\kappa}), \boldsymbol{\kappa}) &= \mathbf{0} \\ \boldsymbol{\lambda}_{n+1}(\boldsymbol{\kappa}) &= \bar{\mathbf{g}}(t_{n+1}, \mathbf{u}_{n+1}(\boldsymbol{\kappa}), \boldsymbol{\kappa}) \end{aligned} \tag{37}$$

The sensitivities are obtained by differentiating Eq. (37)<sub>1</sub> with respect to  $\boldsymbol{\kappa}$  (the indices  $n + 1$  and  $(E)$  are omitted for brevity in the following)

$$\frac{\partial \mathbf{g}}{\partial \boldsymbol{\kappa}} + \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \frac{d\mathbf{u}}{d\boldsymbol{\kappa}} = \mathbf{0} \Rightarrow \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \frac{d\mathbf{u}}{d\boldsymbol{\kappa}} = -\frac{\partial \mathbf{g}}{\partial \boldsymbol{\kappa}}. \tag{38}$$

In other words, a system of linear equations with  $n_\kappa$  right-hand sides has to be computed after each load step  $n$ . The matrix  $\partial \mathbf{g} / \partial \mathbf{u}$  represents the tangential stiffness matrix. For moderate systems and direct solution schemes, one LU-decomposition is required, and  $n_\kappa$  back-substitutions (commonly, the LU-decomposition has already been done which was required for solving the system (37)<sub>1</sub>). Then, the sensitivities of the Lagrange multipliers read

$$\frac{d\boldsymbol{\lambda}}{d\boldsymbol{\kappa}} = -\frac{\partial \bar{\mathbf{g}}}{\partial \boldsymbol{\kappa}} - \frac{\partial \bar{\mathbf{g}}}{\partial \mathbf{u}} \frac{d\mathbf{u}}{d\boldsymbol{\kappa}} \tag{39}$$

requiring the solution of Eq. (38). This implies only a matrix-matrix product. The matrix  $\partial \bar{\mathbf{g}} / \partial \mathbf{u}$  is also a part of the entire tangential stiffness matrix.

In the case of DAEs, we follow the concepts summarized in [69]. The sensitivities can be determined either on the level of the DAE-system 10, yielding the so-called *simultaneous simulation of sensitivities*, see [12,49], or on the time-discretized level—leading to two approaches namely *internal numerical differentiation* (IND) and *external numerical differentiation* (END). The resulting simultaneous sensitivity equations (SSE) can be shown to be equivalent to IND under special conditions (same integrator to obtain the sensitivities). Since the implementation of the SSE yields—in the general case—systems that are too large for our applications, it is not recommended. We follow the sensitivity computation using IND and END as discussed in the following.

### 4.2 Internal numerical differentiation

In the case of internal numerical differentiation, the DAE-system (10) is discretized in time first. All sensitivities are provided by analytical computations. The solution (25)

depends on the parameters  $\kappa$ ,

$$\begin{aligned} \mathbf{u}_{n+1}(\kappa) &= \mathbf{u}_n(\kappa) + \Delta t_n \sum_{i=1}^s b_i \dot{\mathbf{U}}_{ni}(\kappa), \\ \mathbf{q}_{n+1}(\kappa) &= \mathbf{q}_n(\kappa) + \Delta t_n \sum_{i=1}^s b_i \dot{\mathbf{Q}}_{ni}(\kappa), \\ \boldsymbol{\lambda}_{n+1}(\kappa) &= \boldsymbol{\lambda}_n(\kappa) + \Delta t_n \sum_{i=1}^s b_i \dot{\boldsymbol{\Lambda}}_{ni}(\kappa). \end{aligned} \tag{40}$$

Since we draw on stiffly accurate diagonally-implicit Runge–Kutta methods, the final solution at each time  $t_{n+1}$  is directly provided by the non-linear system,  $\mathbf{u}_{n+1}(\kappa) = \mathbf{U}_{ns}(\kappa)$ ,  $\mathbf{q}_{n+1}(\kappa) = \mathbf{Q}_{ns}(\kappa)$ ,  $\boldsymbol{\lambda}_{n+1}(\kappa) = \boldsymbol{\Lambda}_{ns}(\kappa)$ . Thus, we need the derivatives of the stage derivatives and the starting values with respect to  $\kappa$ ,

$$\begin{aligned} \frac{d\dot{\mathbf{U}}_{ni}}{d\kappa} &= \frac{1}{\Delta t_n a_{ii}} \left[ \frac{d\mathbf{U}_{ni}}{d\kappa} - \frac{d\mathbf{S}_{ni}^u}{d\kappa} \right], \\ \frac{d\dot{\mathbf{Q}}_{ni}}{d\kappa} &= \frac{1}{\Delta t_n a_{ii}} \left[ \frac{d\mathbf{Q}_{ni}}{d\kappa} - \frac{d\mathbf{S}_{ni}^q}{d\kappa} \right], \end{aligned} \tag{41}$$

with

$$\begin{aligned} \frac{d\mathbf{S}_{ni}^u}{d\kappa} &= \frac{d\mathbf{u}_n}{d\kappa} + \Delta t_n \sum_{j=1}^{i-1} \frac{d\dot{\mathbf{U}}_{nj}}{d\kappa}, \\ \frac{d\mathbf{S}_{ni}^q}{d\kappa} &= \frac{d\mathbf{q}_n}{d\kappa} + \Delta t_n \sum_{j=1}^{i-1} \frac{d\dot{\mathbf{Q}}_{nj}}{d\kappa}. \end{aligned} \tag{42}$$

Thus, the derivatives (41) must be stored additionally, and we obtain the dependencies

$$\begin{aligned} \mathbf{g}(T_{ni}, \mathbf{U}_{ni}(\kappa), \mathbf{Q}_{ni}(\kappa), \kappa) &= \mathbf{0}, \\ \mathbf{l}(T_{ni}, \mathbf{U}_{ni}(\kappa), \mathbf{Q}_{ni}(\kappa), \mathbf{S}_{ni}^q(\kappa), \kappa) &= \mathbf{0}, \\ \bar{\mathbf{g}}(T_{ni}, \mathbf{U}_{ni}(\kappa), \mathbf{Q}_{ni}(\kappa), \kappa) + \boldsymbol{\Lambda}_{ni}(\kappa) &= \mathbf{0}. \end{aligned} \tag{43}$$

We seek for the derivatives  $d\mathbf{U}/d\kappa$  and  $d\mathbf{Q}/d\kappa$ , which can be—similarly to the Multilevel-Newton algorithm—be determined by assuming the implicit function theorem (here, we briefly recap the theory in [24]). We assume that the conditions of the implicit function theorem are fulfilled, i.e. the stage quantities of the internal variables can be represented by a function  $\mathbf{Q}_{ni} = \hat{\mathbf{Q}}(\mathbf{U}_{ni}(\kappa), \kappa)$ . This function is inserted into Eq. (43)<sub>1</sub>,

$$\mathbf{g}(T_{ni}, \mathbf{U}_{ni}(\kappa), \hat{\mathbf{Q}}(\mathbf{U}_{ni}(\kappa), \kappa), \kappa) = \mathbf{0}, \tag{44}$$

and calculate the derivative with respect to  $\kappa$  (we omit again the temporal index  $ni$ )

$$\left[ \frac{\partial \mathbf{g}}{\partial \mathbf{U}} + \frac{\partial \mathbf{g}}{\partial \mathbf{Q}} \frac{d\hat{\mathbf{Q}}}{d\mathbf{U}} \right] \frac{d\mathbf{U}}{d\kappa} = -\frac{\partial \mathbf{g}}{\partial \kappa} - \frac{\partial \mathbf{g}}{\partial \mathbf{Q}} \frac{\partial \hat{\mathbf{Q}}}{\partial \kappa} \tag{45}$$

On the left, the coefficient matrix is again the tangential stiffness matrix of the non-linear solver in the FE-program. The matrix  $\partial \hat{\mathbf{Q}}/\partial \kappa$  is obtained from the second equation (integration step for the internal variables)

$$\mathbf{l}(T_{ni}, \mathbf{U}_{ni}(\kappa), \hat{\mathbf{Q}}(\mathbf{U}_{ni}(\kappa), \kappa), \mathbf{S}_{ni}^q(\kappa), \kappa) = \mathbf{0}, \tag{46}$$

by applying again the chain rule, see Eq. (21) for fulfilled constraint (20)<sub>3</sub>,

$$\begin{aligned} \left[ \frac{\partial \mathbf{l}}{\partial \mathbf{U}} + \frac{\partial \mathbf{l}}{\partial \mathbf{Q}} \frac{d\hat{\mathbf{Q}}}{d\mathbf{U}} \right] \frac{d\mathbf{U}}{d\kappa} + \frac{\partial \mathbf{l}}{\partial \mathbf{Q}} \frac{\partial \hat{\mathbf{Q}}}{\partial \kappa} \\ + \frac{\partial \mathbf{l}}{\partial \mathbf{S}^q} \frac{d\hat{\mathbf{S}}^q}{d\kappa} + \frac{\partial \mathbf{l}}{\partial \kappa} = \mathbf{0}, \end{aligned} \tag{47}$$

where the first matrix on the left (term within the brackets) vanishes in the Multilevel-Newton algorithm, see [24, Eq.(23)]. With  $\partial \mathbf{l}/\partial \hat{\mathbf{S}}^q = -\mathbf{1}$ , the linear system

$$\frac{\partial \mathbf{l}}{\partial \mathbf{Q}} \frac{\partial \hat{\mathbf{Q}}}{\partial \kappa} = \frac{d\hat{\mathbf{S}}^q}{d\kappa} - \frac{\partial \mathbf{l}}{\partial \kappa} \tag{48}$$

has to be computed in each stage. Since we assembled formally these equations, see Eqs.(13)-(14), they can be decoupled in this step. In other words, after each stage small linear systems have to solved on Gauss-point level (in our case with 6 internal variables,  $6 \times n_\kappa$ ). This result is inserted into Eq. (45).

Since the Lagrange-multipliers are explicitly determinable,

$$\boldsymbol{\Lambda}_{ni}(\kappa) = -\bar{\mathbf{g}}(T_{ni}, \mathbf{U}_{ni}(\kappa), \hat{\mathbf{Q}}(\mathbf{U}_{ni}(\kappa), \kappa), \kappa) \tag{49}$$

the sensitivity is based only on quantities derived before,

$$\frac{d\boldsymbol{\Lambda}}{d\kappa} = - \left[ \frac{\partial \bar{\mathbf{g}}}{\partial \mathbf{U}} + \frac{\partial \bar{\mathbf{g}}}{\partial \mathbf{Q}} \frac{\partial \hat{\mathbf{Q}}}{\partial \mathbf{U}} \right] \frac{d\mathbf{U}}{d\kappa} - \frac{\partial \bar{\mathbf{g}}}{\partial \mathbf{Q}} \frac{\partial \hat{\mathbf{Q}}}{\partial \kappa} - \frac{\partial \bar{\mathbf{g}}}{\partial \kappa}. \tag{50}$$

The “classical” derivatives within the Multilevel-Newton algorithm are provided in an analytical form in [22], whereas the sensitivities required for the numerical optimizer are computed by means of the program Acegen [41–43].

### 4.3 External numerical differentiation

External numerical differentiation, i.e. the computation of the sensitivities (34) by means of numerical differentiation is

the usual procedure in the NLS-FEM-DIC approach. In this case, the derivatives are approximated by

$$\frac{d\tilde{\mathbf{u}}_{n+1}}{d\boldsymbol{\kappa}} \approx \sum_{j=1}^{n_\kappa} \frac{\tilde{\mathbf{u}}_{n+1}(\boldsymbol{\kappa} + \Delta\kappa_j \bar{\mathbf{e}}_j) - \tilde{\mathbf{u}}_{n+1}(\boldsymbol{\kappa})}{\Delta\kappa_j} \bar{\mathbf{e}}_j^T, \tag{51}$$

with the vectors  $\bar{\mathbf{e}}_j \in \mathbb{R}^{n_\kappa}$  (all entries are zero except for one having a 1 in row  $j$ ). This holds similarly for the Lagrange multipliers,  $d\boldsymbol{\lambda}_{n+1}/d\boldsymbol{\kappa}$ , as well. To determine these derivatives, the finite element solver, i.e. the DAE-solution (22), is computed with  $\boldsymbol{\kappa}$  and  $\boldsymbol{\kappa} + \Delta\kappa_j \bar{\mathbf{e}}_j$ . The derivatives  $d\mathbf{u}_{n+1}/d\boldsymbol{\kappa}$  and  $d\boldsymbol{\lambda}_{n+1}/d\boldsymbol{\kappa}$  from all time steps  $t_{n+1}$  must be stored. END has an essential advantage if the finite element program is treated as a black-box solver, for example as it is the case for commercial programs. A disadvantage is to be seen in the computational time required and the unawareness of the exact implementation.

### 4.4 Quality measures

Today, there are many tools to identify material parameters using a least-square approach. They are programmed very stable. The  $R^2$ -value,  $R^2 = 1 - (\sum_{i=1}^{n_d} (d_i - s_i)^2) / (\sum_{i=1}^{n_d} (d_i - \bar{d})^2)$  with  $\bar{d} = (1/n_d) \sum_{i=1}^{n_d} d_i$ ,  $R^2 \leq 1$ , gives a hint on how the model reflects the course of the experimental data. However, it does not say anything about the quality of the parameters found. Either one stops with the iterative scheme in a local minimum of the non-linear least-square problem, or a subset of the material parameters is not uniquely determinable. In this case, it is possible to obtain infinite combinations of the parameters—with the same quality regarding the experimental curves. In this context, the physical meaning of the parameters becomes questionable. This was mentioned in [4,7] for the general case in parameter identification, and studied in solid mechanics in [25,33,71]. Since this is only a local concept in the solution found by some optimizer, it is merely an indicator (so far, however, it appears to be quite a useful indicator). In its theoretical development, an expansion is done in the solution, and the curvature is investigated to find out whether a local “valley” of the goal function exists. In this case, the Hessian

$$\begin{aligned} \mathbf{H}(\boldsymbol{\kappa}) &= \frac{d^2 f(\boldsymbol{\kappa})}{d\boldsymbol{\kappa} d\boldsymbol{\kappa}} = \left[ \frac{\partial^2 f(\boldsymbol{\kappa})}{\partial \kappa_i \partial \kappa_j} \right] \\ &= \left[ \sum_{k=1}^{n_d} w_{kk}^2 \left( \frac{\partial^2 s_k(\boldsymbol{\kappa})}{\partial \kappa_i \partial \kappa_j} (s_k(\boldsymbol{\kappa}) - d_k) + \frac{\partial s_k(\boldsymbol{\kappa})}{\partial \kappa_i} \frac{\partial s_k(\boldsymbol{\kappa})}{\partial \kappa_j} \right) \right] \end{aligned} \tag{52}$$

is exploited, or its approximation

$$\mathbf{H} \approx \mathbf{D}^T \mathbf{D} = \left[ w_{kk}^2 \frac{\partial s_k(\boldsymbol{\kappa})}{\partial \kappa_i} \frac{\partial s_k(\boldsymbol{\kappa})}{\partial \kappa_j} \right]. \tag{53}$$

We check the condition  $\det \mathbf{H} \neq 0$  to verify whether there might be a unique solution, see for details [4,7].

To our experience, the investigation of identifiability should be done using a re-identification procedure. This means that we perform a direct computation with the parameter found by the optimizer, with the same path and boundary conditions of the experiment. Then, we try to re-identify the parameters and evaluate the Hessian. The reason for this procedure is that scattering of real experiments can essentially influence the value of the determinant so that misinterpretations regarding the results can occur. A certain drawback of this indicator is that in real applications  $\det \mathbf{H}$  is never really zero so that the question occurs how small should be this measure so that the results become questionable. Unfortunately, this essentially depends on the problem under consideration (size of  $\mathbf{s}$ , number of parameters, influence of weighting factors, ...). The smaller it become the more sensitive the parameter identification process becomes. Thus, it is a experienced-based quantity.

A measure of the quality of the parameters is also provided by the confidence interval—here, we draw on the real experimental data

$$\boldsymbol{\kappa}_{\text{conf}} = \boldsymbol{\kappa}^* \pm \Delta\boldsymbol{\kappa}, \tag{54}$$

which is based on the diagonal components of the covariance matrix,

$$\Delta\kappa_i = \sqrt{P_{ii}}, \quad i = 1, \dots, n_d, \tag{55}$$

with

$$\mathbf{P} = s^2 \mathbf{H}^{-1}(\boldsymbol{\kappa}^*). \tag{56}$$

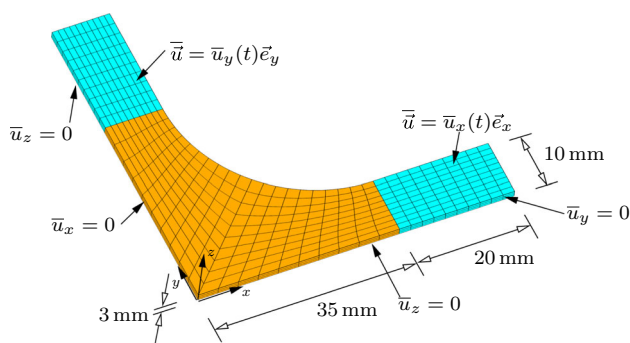
Here, we have the standard deviation

$$s^2 = \frac{1}{n_D - 1} \mathbf{r}^T(\boldsymbol{\kappa}^*) \mathbf{r}(\boldsymbol{\kappa}^*). \tag{57}$$

These investigations can be done after determining a parameter set that is totally independent of the parameter identification method. In view of analytical and numerical parameter identification studies, we would like to refer to [34]. A result of the covariance matrix is given by the correlation matrix

$$\mathbf{C} = [c_{ij}] \quad \text{with} \quad c_{ij} = \frac{P_{ij}}{\sqrt{P_{ii}} \sqrt{P_{jj}}}, \quad i, j = 1, \dots, n_\kappa. \tag{58}$$

This matrix provides a hint whether parameters are correlated, see, for example [11].



**Fig. 4** Mesh using 20-noded Q2P1-elements and the boundary conditions concerned ( $n_u = 8472$ )

### 5 Identification at biaxial tensile test

To identify the material parameters  $\kappa_{eq}$  and  $\kappa_{ov}$  we proceed as follows. In a first step, only the termination points of relaxation resulting from the loading process in Fig. 3b are chosen to determine  $\kappa_{eq}$ . In the second identification step, we draw on the loading with four different displacement rates, see Fig. 3a, and the multi-step relaxation loading path shown in Fig. 3b. These tests are required to determine the material parameter set  $\kappa_{ov}$ .

The geometry of the biaxial tensile specimens are shown in Fig. 2. We exploit triple symmetry, i.e. only one-eighth of the specimen is discretized, and draw on the 20-noded mixed hexahedral elements (Q2P1-elements of [74], see also [22] for details. Figure 4 shows the mesh, and the boundary conditions concerned. The triangulated surface regions of both the DIC-data points as well as the finite element nodal points are compared in Fig. 5. Here, we apply the triangulation tool proposed in [72,73].

#### 5.1 Identification of equilibrium stress part

With regard to the findings of a unique identification of the material parameters in biaxial experiments from [33], one approach is to apply a much larger deformation in one direction than in the other direction. We define a four times larger displacement in vertical direction, see Fig. 3b. Since we have an overstress-type model, the material parameters  $\kappa_{eq}^T = \{K, c_{10}, c_{01}\}$  of the equilibrium stress part are determined by the termination points of relaxation in a first step,  $\tilde{\mathbf{T}}_{ov} = \mathbf{0}$  in Eq. (2). We store the maximum and minimum strain state of the DIC-data as well as the vertical and horizontal forces of the testing machine’s force gauge at the termination points of relaxation. These data-points are compared with the finite element solution in a least-square sense. In this case, there are no internal variables so that we have only a NLS-problem applied to systems of non-linear equations, see Eq. (37), considering either IND, see Eqs.(38)- (39) for the functional matrices required by the Matlab tool

lsqnonlin.m, or END, as discussed in Sect. 4.3. Here, we compare both the surface strains of the strain measure  $\mathbf{E}^{(1)} = \mathbf{U} - \mathbf{1}$ , where  $\mathbf{U}$  is the right stretch tensor of the polar decomposition of the deformation gradient  $\mathbf{F} = \mathbf{R}\mathbf{U}$ , and the force-displacement curves provided by the termination points of relaxation.

The resulting material parameters are compiled in Table 1. We obtain the results in Fig. 6a with an  $R^2$ -value of 0.994. The computational strain distribution at the final point is compared with the experimental strain distribution. Here, we consider the maximum and minimum principal strains. Figure 6b shows the relative error of the principal maximum strains

$$e = \frac{|\varepsilon_{\max, \text{exp}} - \varepsilon_{\max, \text{sim}}|}{|\varepsilon_{\max, \text{exp}}|} \times 100. \tag{59}$$

Here, the relative error is around 20% in the center region of the specimen. The confidence interval of the material parameter indicates that only  $c_{10}$  is sensitive, which is due to the fact that the middle region of the sample is primarily considered in the identification, not so much the heavily loaded sample arms. It is well-known that the first and second invariant  $I_{\bar{\mathbf{C}}}$  and  $II_{\bar{\mathbf{C}}}$  are strongly coupled in pure tensile tests. In the biaxial tensile test this is less pronounced, but visible in the correlation matrix:

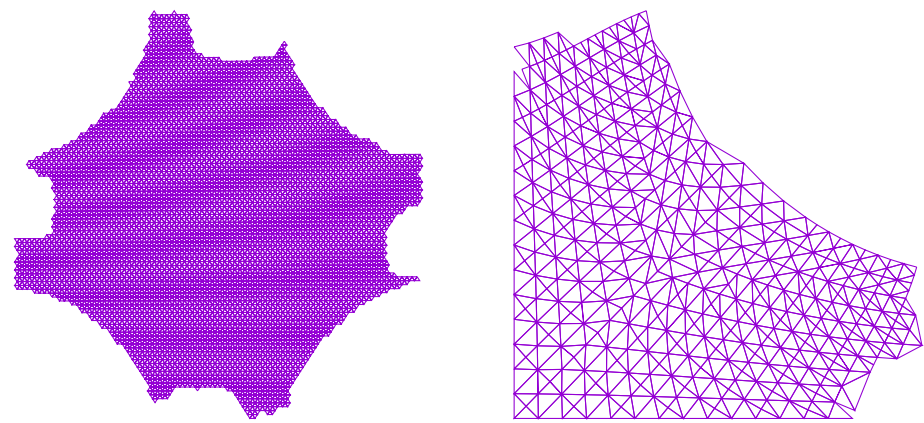
$$\mathbf{C} = \begin{bmatrix} 1.0 & 0.107 & -0.623 \\ & 1.0 & -0.841 \\ & & 1.0 \end{bmatrix} \tag{60}$$

There is a strong correlation between the parameters  $c_{10}$  and  $c_{01}$ , see Eq. (58), which is known since the invariants  $I_{\bar{\mathbf{C}}}$  and  $II_{\bar{\mathbf{C}}}$  are not independent of each other in the small strain range. Here, we have a square of the standard deviation of  $s^2 = 0.053$ .

The computations require 25 iterations with a starting vector  $\kappa_{eq}^{(0)} = \{100, 0.264, 0.5\}$ . In the case of END, we have 104 function calls, and it requires a factor of the computational time of approximately 3.4 more than IND. It should be noted that the number of iterations with the starting vectors can be more or less. However, only a local minimum is met, i.e. there is no statement about a global minimum with the selected entire numerical scheme. In our experience to do so far, the concept of local identifiability provides relatively stable, unique solutions.

To find out whether the material parameters are uniquely identifiable, a re-identification concept is applied. In this case, a forward computation is performed with the material parameters found, and the simulation data is chosen as virtual experimental data. Then, the determinant of the Hessian (53) is evaluated. If the parameters are not uniquely identifiable, the determinant is zero. In our case,  $\det \mathbf{H} \approx 1.287 \times 10^{-3}$  holds. Thus, identifiability is guaranteed according to the

**Fig. 5** Triangulation of surface data for strain determination

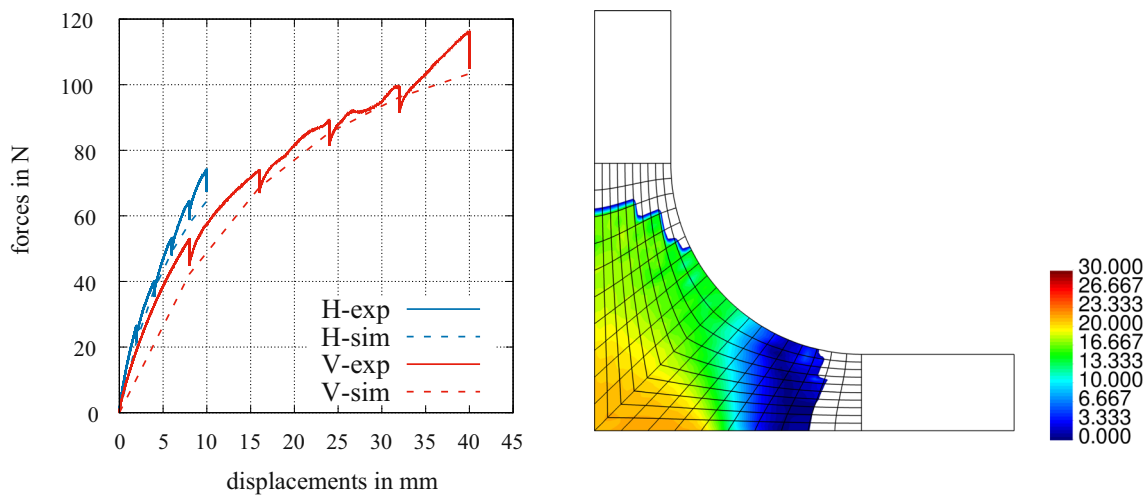


**(a)** Triangulation of the entire DIC-data

**(b)** Triangulation of the FEM-data (only the region which is compared to the DIC-data is shown)

**Table 1** Identified material parameters of equilibrium stress part and overstress part

Equilibrium stress part $\kappa_{eq}$			Overstress part $\kappa_{ov}$		
$K$ $N\ mm^{-2}$	$c_{10}$ $N\ mm^{-2}$	$c_{01}$ $N\ mm^{-2}$	$\mu_0$ $N\ mm^{-2}$	$\eta_0$ $s\ mm\ N^{-1}$	$s_0$ $(N\ mm^{-2})$
$5706.81 \pm 222$	$0.012 \pm 0.007$	$0.33 \pm 0.004$	$0.26 \pm 0.02$	$14113.39 \pm 4405$	$0.00053 \pm 0.006$



**(a)** Calibration of equilibrium stress part at termination points of relaxation

**(b)** Relative error  $e$  of final principal strain, see definition (59)

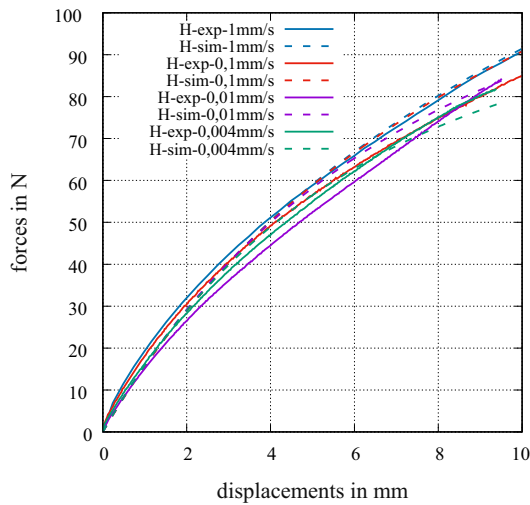
**Fig. 6** Force-displacement identification result and relative error in principal strains

chosen indicator, see discussion in [33], since it is moderately small.

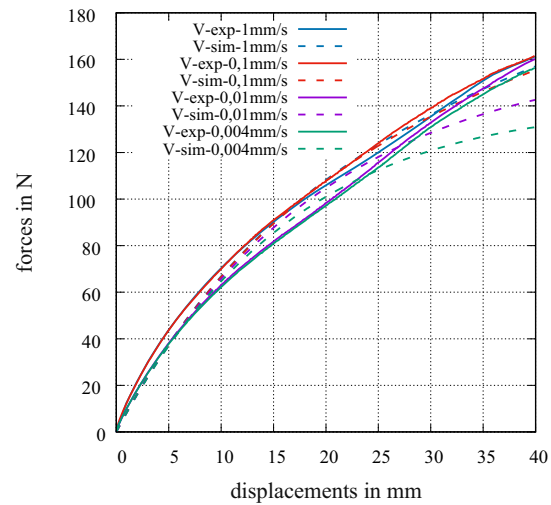
### 5.2 Identification of overstress part

In the viscoelastic case, we have  $n_u = 8472$  unknown nodal displacements,  $n_Q = 194400$  unknown internal variables in the whole structure. Here, we fix the material parameters  $\kappa_{eq}$ ,

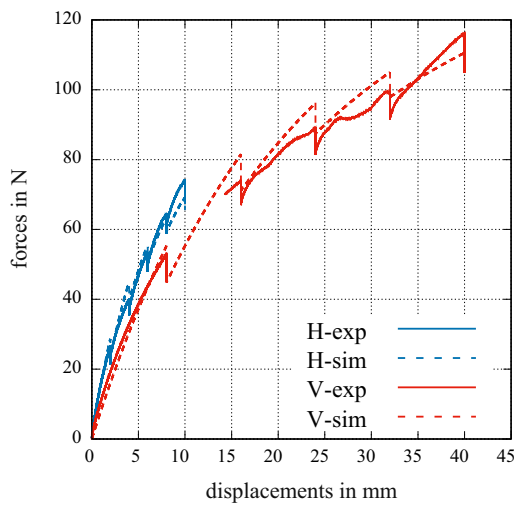
see Table 1, and determine the parameters  $\kappa_{ov}^T = \{\mu_0, \eta_0, s_0\}$  on the basis of the four displacement rates in the loading paths of Fig. 3a and the multi-step relaxation path of Fig. 3b. Once again, we make use of the measured horizontal and vertical force data as well as the maximum and minimum strain data, adding up to significantly more data than for the parameter identification of the equilibrium stress part. The time-adaptive time integrator is the second-order scheme of



(a) Horizontal rate-dependent calibration results



(b) Vertical rate-dependent calibration results



(c) Calibration of overstress-type model at multi-step relaxation path

**Fig. 7** Force-displacement results of identification of rate dependence and relaxation

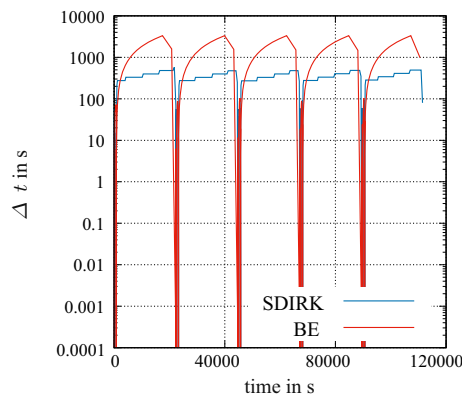
Ellsiepen, see [10,14], which is based on the second-order method of [1]. The result of the identification of the force-displacement curves is shown in Fig. 7 (the strain data is not shown here). We have  $R^2 = 0.9903$ , and the parameters are assembled in Table 1, resulting from the starting values of the iterative scheme  $\kappa_{ov}^{(0)} = \{0.2, 180., 0.001\}$ . Since the confidence interval is larger than the value itself, the parameter  $s_0$  required to incorporate rate-dependence is not of high quality. This is indicated also by the correlation matrix

$$\mathbf{C} = \begin{bmatrix} 1 & -0.075 & 0.015 \\ & 1 & 0.934 \\ & & 1 \end{bmatrix}, \tag{61}$$

where the viscosity  $\eta_0$  and  $s_0$  are strongly correlated. The square of the standard deviation  $s^2 = 0.0573$  is, however, small. The re-identification procedure yields the determinant of the Hessian  $\det \mathbf{H} \approx 1.71 \times 10^{-4}$  indicating identifiability.

The comparison of the computations with analytical derivatives (IND) with numerical differentiation (END) again yields a factor of 3.5, i.e. END requires much more time for the identification process.

A brief comparison of a Backward–Euler computation with the step-size controlled, second-order SDIRK-method of Ellsiepen (both computed adaptively—however, the BE-method requires the number of global Newton-iterations within the Multilevel-Newton algorithm, where we reduce the step-size by a factor of 0.5 if the number of iterations



**Fig. 8** Step-size behavior of Backward–Euler method and second-order SDIRK-scheme for multi-step-relaxation path

exceeds 15, and increase the load-step by a factor of 1.2 if the iteration number is less than 4) yields for one computation of the multi-step relaxation path of Fig. 3b the step-size behavior shown in Fig. 8.

Although the time-steps become larger for the BE-scheme during the relaxation path, it requires a factor of 1.12 only for this process. However, this depends essentially on the error tolerances assumed for the SDIRK time-step estimation. After each change of the process path, the implementation of the step-size control starts with a step-size of  $\Delta t_n = 10^{-4}$  s again.

## 6 Conclusions

In this paper a non-linear least-square method of Matlab, which requires gradients, is applied to the residual between full-field measurement data and finite-element computations. The usual approach of only including displacements has been extended here with respect to the consideration of reaction forces and a three-dimensional surface strain tool applied to the finite element nodal data. This was done in particular both in the derivation of the equations by the method of Lagrange multipliers and the three-dimensional strain calculation tool, which are necessary in the analytical calculation of the sensitivities. Both are required since considering forces improves the identification process, while the strain information circumvents rigid body motions which are commonly inherent in displacement data. In particular, the in-plane strain analysis tool is a simpler approach for curved three-dimensional surfaces, which is difficult to implement in the coding of finite element programs. Furthermore, high-order diagonally-implicit Runge–Kutta method are chosen as a time-integrator for the resulting DAE-system. As a consequence of using a thorough matrix notation and of interpreting the finite element method as the solution of DAE-systems, it is possible to use methods of numerical

mathematics and to discover a general approach. In this case, a comparison of internal numerical differentiation (the sensitivities are computed with analytical expressions provided by the code-generation tool Acegen) and external numerical differentiation is provided. Although the latter is more flexible, it requires more than a factor of 3 of computational time—in our applications (dependent on the number of material parameters). This is demonstrated using data for rubber based on biaxial tensile experiments. Additionally, it is shown that measures such as the confidence interval, correlation matrix, and the concept of identifiability are helpful indicators of the quality resulting from the optimization tool. It turns out that not all parameters of the applied finite strain viscoelasticity model are necessary to reproduce the biaxial tensile test data using digital image correlation.

**Acknowledgements** We gratefully acknowledge Dr. Kröger from the DIK (Deutsches Institut für Kautschuktechnologie e.V., Hannover, Germany for providing us the specimens). Furthermore, we would like to thank Dr. Matthias Grafenhorst for his programming support

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

**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/>.

## A Sensitivities of the surface stretches

In Eq. (36) the derivatives of the stretches  $v_k$  with respect to the nodal displacements are required. In [28] the formulas to calculate the in-plane surface stretches by the displacements of the points are explained in detail. The squares of stretches  $\mu_k = v_k^2$  are computed by

$$\mu_{1,2} = \frac{I_{\hat{C}}}{2} \pm \sqrt{\frac{I_{\hat{C}}^2}{4} - II_{\hat{C}}^2}. \tag{62}$$

with the invariants of the right Cauchy–Green tensor

$$I_{\hat{C}} = \hat{C}^1_1 + \hat{C}^2_2, \quad II_{\hat{C}} = \hat{C}^1_1 \hat{C}^2_2 - \hat{C}^1_2 \hat{C}^2_1, \tag{63}$$

where

$$\hat{C}^\alpha_{\beta}(\kappa) = A^{\alpha\beta} \hat{C}_{\alpha\beta}(\kappa), \quad \alpha = 1, 2, \quad \beta = 1, 2 \tag{64}$$

are the mixed-variant components.  $[A^{\alpha\beta}] = [A_{\alpha\beta}]^{-1}$  can be computed by the metric of the tangent vectors,  $\vec{A}_\alpha$ ,  $\alpha = 1, 2$ , in the reference configuration  $A_{\alpha\beta} = \vec{A}_\alpha \cdot \vec{A}_\beta$ . If the surface is described by finite element similar approach, we have

$$\vec{A}_\alpha = \frac{\partial \vec{X}}{\partial \Theta^\alpha} = \left( \sum_{k=1}^{n_{\text{en}}} N_{k,\alpha} X_{kj} \right) \vec{e}_j, \quad (65)$$

where  $X_{kj}$  are the coordinates in the reference configuration, and  $N_k(\Theta^1, \Theta^2)$  a surface description (similar to the shape functions in finite elements).  $n_{\text{en}}$  represents the number of “element nodes”. For convective coordinates, we have  $\hat{C}_{\alpha\beta} = a_{\alpha\beta}$  with the metric coefficients  $a_{\alpha\beta} = \vec{a}_\alpha \cdot \vec{a}_\beta$ , which are based on the surface tangent vectors in the current configuration. They are approximated by

$$\vec{a}_\alpha(\boldsymbol{\kappa}) = \frac{\partial \vec{x}}{\partial \Theta^\alpha} = \left( \sum_{k=1}^{n_{\text{en}}} N_{k,\alpha} x_{kj}(\boldsymbol{\kappa}, t) \right) \vec{e}_j, \quad (66)$$

with  $x_{kj}(\boldsymbol{\kappa}, t) = X_{kj} + u_{kj}(\boldsymbol{\kappa}, t)$ . In this sense, the derivatives  $\partial \mu_k / \partial \vec{\mathbf{u}}_n$  must be calculated. This is done using Acegen to generate the analytical derivatives [41–43].

## References

- Alexander R (1977) Diagonally implicit Runge–Kutta methods for stiff O.D.E.'s. *SIAM J Numer Anal* 14:1006–1021
- Andresen K, Dannemeyer S, Friebe H, Mahnken R, Ritter R, Stein E (1996) Parameteridentifikation für ein plastisches Stoffgesetz mit FE-Methoden und Rasterverfahren. *Bauingenieur* 71:21–31
- Avril S, Bonnet M, Bretelle AS, Grédiac M, Hild F, Lenny P, Latourte F, Lemosse D, Pagano S, Pagnacco E, Pierron F (2008) Overview of identification methods of mechanical parameters based on full-field measurements. *Exp Mech* 48(4):381–402
- Beck JV, Arnold KJ (1977) *Parameter estimation in engineering and science*. Wiley, New York
- Bellec E (2018) “RE-LAUNCHING” a self-made biaxial machine. Comparison to other biaxial tests and FE simulation for different elastomers. Tech. rep., DIK Deutsches Institut für Kautschuktechnologie e.V., Hannover, Germany
- Benedix U, Görke UJ, Kreißig R, Kretschmar S (1998) Local and global analysis of inhomogeneous displacement fields for the identification of material parameters. In: Hoa SV, De Wilde WP, Blain WR (eds) *Computer methods in composite materials VI (CAD-COMP 98)*, pp 159–168
- Beveridge GSG, Schechter RS (1970) *Optimization: theory and practice*, 1st edn. McGraw-Hill Book Company, New York
- Borgqvist E, Wallin M (2013) Numerical integration of elastoplasticity coupled to damage using a diagonal implicit Runge–Kutta integration scheme. *Int J Damage Mech* 22:68–79
- Cooreman S, Lecompte D, Sol H, Vantomme J, Debruyne D (2007) Elasto-plastic material parameter identification by inverse methods: calculation of the sensitivity matrix. *Int J Solids Struct* 44:4329–4341
- Diebels S, Ellsiesen P, Ehlers W (1999) Error-controlled Runge–Kutta time integration of a viscoplastic hybrid two-phase model. *Tech Mech* 19:19–27
- Draper NR, Smith H (1998) *Applied regression analysis*, 3rd edn. Wiley, New York
- Dunker AM (1984) The decoupled direct method for calculating sensitivity coefficients in chemical kinetics. *J Chem Phys* 81:2385–2393
- Ehlers W, Eipper G (1998) The simple tension problem at large volumetric strains computed from finite hyperelastic material laws. *Acta Mech* 130:17–27
- Ellsiesen P (1999) *Zeit- und ortsadaptive Verfahren angewandt auf Mehrphasenprobleme poröser Medien*. Doctoral thesis, Institute of Mechanics II, University of Stuttgart, report No. II-3
- Ellsiesen P, Hartmann S (2001) Remarks on the interpretation of current non-linear finite-element-analyses as differential-algebraic equations. *Int J Numer Methods Eng* 51:679–707
- Fritzen P (1997) *Numerische Behandlung nichtlinearer Probleme der Elastizitäts- und Plastizitätstheorie*. Doctoral thesis, Department of Mathematics, University of Darmstadt
- Grédiac M, Hild F (eds) (2013) *Full-field measurements and identification in solid mechanics*. Wiley, Hoboken
- Großmann C, Roos H (1994) *Numerik partieller Differentialgleichungen*. Teubner Verlag, Stuttgart
- Hairer E, Wanner G (1996) *Solving ordinary differential equations II*, 2nd edn. Springer, Berlin
- Hairer E, Norsett SP, Wanner G (1993) *Solving ordinary differential equations I*, 2nd edn. Springer, Berlin
- Hartmann S (2001) Numerical studies on the identification of the material parameters of Rivlin’s hyperelasticity using tension-torsion tests. *Acta Mech* 148:129–155
- Hartmann S (2002) Computation in finite strain viscoelasticity: finite elements based on the interpretation as differential-algebraic equations. *Comput Methods Appl Mech Eng* 191(13–14):1439–1470
- Hartmann S (2005) A remark on the application of the Newton–Raphson method in non-linear finite element analysis. *Comput Mech* 36(2):100–116
- Hartmann S (2017) A remark on material parameter identification using finite elements based on constitutive models of evolutionary-type. *Comput Assist Methods Eng Sci* 24:113–126
- Hartmann S, Gilbert RR (2018) Identifiability of material parameters in solid mechanics. *Arch Appl Mech* 88(1):3–26
- Hartmann S, Hamkar AW (2010) Rosenbrock-type methods applied to finite element computations within finite strain viscoelasticity. *Comput Methods Appl Mech Eng* 199(23–24):1455–1470
- Hartmann S, Neff P (2003) Polyconvexity of generalized polynomial-type hyperelastic strain energy functions for near-incompressibility. *Int J Solids Struct* 40(11):2767–2791
- Hartmann S, Rodriguez S (2018) Verification examples for strain and strain-rate determination of digital image correlation systems. In: Altenbach H, Jablonski F, Müller W, Naumenko K, Schneider P (eds) *Advances in mechanics of materials and structural analysis. Advanced structured materials*, no. 80 in advanced structured materials. Springer, Cham, pp 135–174
- Hartmann S, Tschöpe T, Schreiber L, Haupt P (2003) Large deformations of a carbon black-filled rubber. Experiment, optical measurement and parameter identification using finite elements. *Eur J Mech Ser A/Solids* 22:309–324
- Hartmann S, Gibmeier J, Scholtes B (2006) Experiments and material parameter identification using finite elements. Uniaxial tests and validation using instrumented indentation tests. *Exp Mech* 46(1):5–18
- Hartmann S, Quint KJ, Hamkar AW (2008) Displacement control in time-adaptive non-linear finite-element analysis. *ZAMM J Appl Math Mech* 88(5):342–364
- Hartmann S, Duintjer Tebbens J, Quint KJ, Meister A (2009) Iterative solvers within sequences of large linear systems in non-linear structural mechanics. *ZAMM J Appl Math Mech* 89(9):711–728



33. Hartmann S, Gilbert RR, Sguazzo C (2018) Basic studies in biaxial tensile tests. *GAMM-Mitteilungen* 41(e201800):004
34. Hartmann S, Gilbert RR, Kheiri Marghzar A, Leistner C, Dileep PK (2021) Material parameter identification of unidirectional fiber-reinforced composites. *Arch Appl Mech* 97:687–712
35. Haupt P, Sedlan K (2001) Viscoplasticity of elastomeric materials. experimental facts and constitutive modelling. *Arch Appl Mech* 71:89–109
36. Hsu FPK, Schwab C, Rigamonti D, Humphrey JD (1994) Identification of response functions from axisymmetric membrane inflation tests: implications for biomechanics. *Int J Solids Struct* 31:3375–3386
37. Huber N, Tsakmakis C (1999a) Determination of constitutive properties from spherical indentation data using neural networks, part I: plasticity with nonlinear and kinematic hardening. *J Mech Phys Solids* 47:1589–1607
38. Huber N, Tsakmakis C (1999b) Determination of constitutive properties from spherical indentation data using neural networks, part II: the case of pure kinematic hardening in plasticity laws. *J Mech Phys Solids* 47:1569–1588
39. Issa S, Wallin M, Ristinmaa H, Hallberg H (2018) Diagonally implicit Runge–Kutta (DIRK) integration applied to finite strain crystal plasticity modeling. *Comput Mech* 62:1429–1441
40. Kleuter B, Menzel A, Steinmann P (2007) Generalized parameter identification for finite viscoelasticity. *Comput Methods Appl Mech Eng* 196:3315–3334
41. Korelc J (1997) Automatic generation of finite-element code by simultaneous optimization of expressions. *Theor Comput Sci* 187:231–248
42. Korelc J (2002) Multi-language and multi-environment generation of nonlinear finite element codes. *Eng Comput* 18:312–327
43. Korelc J (2009) Automation of primal and sensitivity analysis of transient coupled problems. *Comput Mech* 44(5):631–649
44. Krämer S (2016) Einfluss von Unsicherheiten in Materialparametern auf Finite-Elemente Simulationen. Phd-thesis, report no. 5/2016, Institute of Applied Mechanics, Clausthal University of Technology, Clausthal-Zellerfeld
45. Kreißig R (1998) Auswertung inhomogener Verschiebungsfelder zur Identifikation der Parameter elastisch-plastischer Deformationsgesetze. *Forsch Ingenieurwes* 64:99–109
46. Kreißig R, Benedix U, Goerke UJ (2001) Statistical aspects of the identification of material parameters for elasto-plastic models. *Arch Appl Mech* 71:123–134
47. Kreißig R, Benedix U, Görke UJ, Lindner M (2007) Identification and estimation of constitutive parameters for material laws in elastoplasticity. *GAMM-Mitteilung* 30(2):458–480
48. Lawson CL, Hanson RJ (1995) Solving least squares problems. SIAM Society for Industrial and Applied Mathematics, Philadelphia
49. Leis JR, Kramer MA (1988) The simultaneous solution and sensitivity analysis of systems described by ordinary differential equations. *ACM Trans Math Softw* 14:45–60
50. Lion A (1997) On the large deformation behaviour of reinforced rubber at different temperatures. *J Mech Phys Solids* 45:1805–1834
51. Mahnken R (2000) A comprehensive study of a multiplicative elastoplasticity model coupled to damage including parameter identification. *Comput Struct* 74:179–200
52. Mahnken R (2018) Identification of material parameters for constitutive equations. In: Stein E, De Borst R, Hughes TJR (eds) *Encyclopedia of computational mechanics*, 2nd edn. Wiley, pp 1165–1185
53. Mahnken R, Stein E (1996) A unified approach for parameter identification of inelastic material models in the frame of the finite element method. *Comput Methods Appl Mech Eng* 136:225–258
54. Mahnken R, Stein E (1997) Parameter identification for finite deformation elasto-plasticity in principal directions. *Comput Methods Appl Mech Eng* 147:17–39
55. Meuwissen MHH (1998) A inverse method for the mechanical characterization of metals. Phd-thesis, TU Eindhoven, Eindhoven, The Netherlands
56. Nocedal J, Wright SJ (1999) Numerical optimization. Springer, New York
57. Olberding JE, Francis Suh FK (2006) A dual optimization method for the material parameter identification of a biphasic poroviscoelastic hydrogel: Potential application to hypercompliant soft tissues. *J Biomech* 39:2468–2475
58. Orteu JJ (2009) 3-D computer vision in experimental mechanics. *Opt Lasers Eng* 47:282–291
59. Pierron F, Grédiac M (2012) The virtual fields method: extracting constitutive mechanical parameters from full-field deformation measurements, 1st edn. Springer, New York
60. Pottier T, Toussaint F, Vacher P (2011) Contribution of heterogeneous strain field measurements and boundary conditions modelling in inverse identification of material parameters. *Eur J Mech A/Solids* 30:373–382
61. Rabbat NBG, Sangiovanni-Vincentelli AL, Hsieh HY (1979) A multilevel Newton algorithm with macromodeling and latency for the analysis of large-scale nonlinear circuits in the time domain. *IEEE Trans Circuits Syst* 26:733–740
62. Reese S, Govindjee S (1998) A theory of finite viscoelasticity and numerical aspects. *Int J Solids Struct* 35:3455–3482
63. Rieger A (2005) Zur Parameteridentifikation komplexer Materialmodelle auf der Basis realer und virtueller Testdaten. Phd-thesis, Report No. I-14 (2005), University of Stuttgart (Germany), Institute of Mechanics
64. Rothe S, Hamkar AW, Quint KJ, Hartmann S (2012) Comparison of diagonal-implicit, linear-implicit and half-explicit Runge–Kutta methods in non-linear finite element analyses. *Arch Appl Mech* 82(8):1057–1074
65. Ruybalid AP, Hoefnagels JPM, van der Sluis O, Geers MGD (2016) Comparison of the identification performance of conventional fem updating and integrated DIC. *Int J Numer Methods Eng* 106:298–320
66. Scheday G (2003) Theorie und Numerik der Parameteridentifikation von Materialmodellen der finiten Elastizität und Inelastizität auf der Grundlage optischer Feldmessmethoden. Phd-thesis, Report No. I-11 (2003), University of Stuttgart (Germany), Institute of Mechanics
67. Scherf O, Simeon B (2003) Differential-algebraic equations in elasto-viscoplasticity. In: Hutter K, Baaser H (eds) *Deformation and failure in metallic materials*, vol 10. Lecture notes in applied and computational mechanics. Springer, pp 31–50
68. Schiesser WE (1991) The numerical method of lines: integration of partial differential equations. Academic Press
69. Schittkowski K (2002) Numerical data fitting in dynamical systems. Kluwer Academic Publ, Dordrecht
70. Schnur DS, Zabarav N (1992) An inverse method for determining elastic material properties and a material interface. *Int J Numer Methods Eng* 33:2039–2057
71. Sewerin F (2020) On the local identifiability of constituent stress-strain laws for hyperelastic composite materials. *Comput Mech* 65:853–876
72. Shewchuk JR (1996) Triangle: engineering a 2D quality mesh generator and delaunay triangulator. In: Lin MC, Manocha D (eds) *Applied computational geometry: towards geometric engineering*. Lecture notes in computer science, vol 1148. The first ACM workshop on applied computational geometry. Springer, pp 203–222
73. Shewchuk JR (2002) Delaunay refinement algorithms for triangular mesh generation. *Comput Geom Theory Appl* 22(1–3):21–74

74. Simo JC, Taylor RL (1991) Quasi-incompressible finite elasticity in principal stretches. Continuum basis and numerical algorithms. *Comput Methods Appl Mech Eng* 85:273–310
75. Springmann M, Kuna M (2005) Identification of material parameters of the Gurson–Tvergaard–Needleman model by combined experimental and numerical techniques. *Comput Mater Sci* 33:501–509
76. Sutton MA, Orteu JJ, Schreier HW (2009) *Image correlation for shape, motion and deformation measurements*. Springer, New York
77. Wittekandt J (1991) *Die numerische Lösung von Anfangs-Randwertproblemen zur Beschreibung inelastischen Werkstoffverhaltens*. Doctoral thesis, Department of Mathematics, University of Darmstadt

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