Chair for Computation in Engineering
Fakultät für Bauingenieur- und Vermessungswesen
Technische Universität München

High-order finite elements for material and geometric nonlinear finite strain problems

Ulrich Heisserer

Vollständiger Abdruck der von der Fakultät für Bauingenieur- und Vermessungswesen der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. Dr.-Ing. habil. G. H. Müller

Prüfer der Dissertation:

1. Univ.-Prof. Dr.rer.nat. E. Rank
2. Priv.-Doz. Dr.-Ing. St. Hartmann, Universität Kassel

Die Dissertation wurde am 24.09.2007 bei der Technischen Universität München eingereicht und durch die Fakultät für Bauingenieur- und Vermessungswesen am 31.01.2008 angenommen.

Zusammenfassung

Finite Elemente hoher Ordnung (p-Version) werden zur Simulation von geometrisch und materiell nichtlinearen Problemen angewandt. Neben hyperelastischen Materialien wird ein viskoplastisches Modell mit inneren Variablen verwendet. Das Algebro-Differentialgleichungssystem, welches aus der räumlichen Diskretisierung der schwachen Form entsteht, wird mit der Backward-Euler Methode zusammen mit dem Mehrebenen-Newton-Verfahren gelöst. Um den Prozess des kalt-isostatischen Pressens effizient abzubilden, werden ein axialsymmetrisches Element für große Dehnungen, Reaktionskräfte und Folgelasten für die p-Version abgeleitet. Analytische Vergleichslösungen zeigen, dass die p-Version volumetrisches Locking auch für große Dehnungen überwindet. Die effiziente Anwendung der entwickelten Methoden auf einaxiales und isostatisches Pressen von Metallpulvern wird demonstriert. Ein komplexes Validierungsbeispiel zeigt gute Übereinstimmung mit dem Experiment.

Abstract

For the simulation of geometric and material nonlinear problems implicit high-order (p-version) displacement-based finite elements are applied. Beside hyperelastic materials a finite strain viscoplasticity model with internal variables is considered. We apply the combination of Backward-Euler integration and Multilevel-Newton algorithm to solve the system of differential-algebraic equations resulting from the space-discretized weak form. For an efficient modeling of the cold isostatic pressing process an axisymmetric finite strain element, reaction forces and follower loads are derived in the p-version context. We demonstrate that the p-version can overcome volumetric locking also in the finite strain case. An adaptive time-stepping algorithm is presented to perform simulations of metal powder compaction. We report applications to die-compaction and isostatic pressing processes, and a complex validation example where a good agreement to experimental data is achieved.

Acknowledgements

I would like to thank all those people who contributed in any way to this dissertation. First of all, I thank my supervisor Prof. Ernst Rank for his continuous support and the opportunity to work in a very stimulating environment on the GIF project. PD Dr.-Ing. Alexander Düster was not only significantly involved in the definition of the project, but also had always an open door for discussions and dared to join the adventures in Israel.

I owe Dominik Scholz very much for introducing me to the world of high-order finite elements and supervising my diploma thesis. Hanne Cornils not only managed to 'run' our chair, she indeed was the heart and soul, I wish her all the best. Especially, I want to thank my officemate Petra Wenisch for good times and her friendship. Thanks go to all colleagues, with whom I share many pleasant memories and look back to some thrilling (table) soccer games.

For a friendly and fruitful cooperation, I would like to thank all colleagues of this GIF project, namely Prof. Moshe P. Dariel, Prof. Nahum Frage, Prof. Stefan Hartmann, Prof. Stefan Holzer, and Prof. Zohar Yosibash as well as Wolfgang Bier, Idit Cohen, Magda Martins-Wagner, and Moti Szanto. I specially want to thank Zohar Yosibash and his group for the hearty hospitality during my stays in Israel.

Furthermore, I want to thank Stefan Hartmann for being the second reviewer of this thesis and his very constructive and detailed remarks. Additionally, I thank Prof. Gerhard Müller for chairing the examination.

With gratitude, I think of all those people who benevolently fostered my development — my family, teachers, and friends in Friedberg, Munich, and beyond. And Andrea, thanks for all the fish.

The support by the German-Israeli Foundation of Scientific Research and Development under grant I-700- 26.10/2001 is gratefully acknowledged.

Contents

Notation

Wherever possible without overloading the notation, it is tried to reflect the different conceptional representations (tensor, column matrix etc.) and discretization stages in the symbolism.

Second order tensors are written as capital roman letters, for example the Green strain tensor $\mathbf{E}$. For computer implementation, the components of second order tensors are collected in column matrices denoted by sans-serif bold letters. Work conjugate symmetric tensors like the Green strain tensor $\mathbf{E}$ and the second Piola-Kirchhoff stress tensor $\tilde{\mathbf{T}}$ are treated according to the Voigt notation (compare Footnote 11 on p. 61) and rendered $\boldsymbol{E}$, $\tilde{\boldsymbol{T}}$. For non-symmetric tensors like the displacement gradient $\mathbf{G} = \operatorname{Grad} \vec{u}$, Eq. (3.8), the order of terms in the column matrix representation is stipulated, cf. in Eq. (4.49). In view of the finite element method, we distinguish global (assembled) quantities written in italic sans-serif letters like $\boldsymbol{E}$ and their restriction to an element Ω^{e} – denoted by upright sans-serif capitals like E^e. If a quantity is 'living' in the vector space of Ansatz functions (FE-space) it is denoted by a superscript h as $\boldsymbol{E}^h$. Geometric vectors in the reference configuration, cf. Fig. 2.1, are written as capital letters, e.g. $\vec{X}$, in the current configuration as small letters, like $\vec{x}$. In an analogous way the column matrix representation is $\boldsymbol{X}$ and $\boldsymbol{x}$. The displacement field $\vec{u} = \vec{X} - \vec{x}$ is approximated in the FE-space as $\vec{u} \approx \vec{u}^{\,h}$ and in column matrix notation referred as $\boldsymbol{u}^h$.

For later reference, some tables are compiled to show how the notation reflects the discretization process from the continuous tensor term to the spatially discretized expression in column matrix representation. It is helpful to see at a glance that the gradient-displacement matrix $\boldsymbol{G}$ for example results from discretizing the gradient field $\mathbf{G} = \operatorname{Grad} \vec{u}$ and is written in column matrix notation $\boldsymbol{\Theta}^h$ (FE-space). Or compare how the strain-displacement matrix $\boldsymbol{B}$ is used to compute the discrete variation of the Green-Lagrange $\delta\boldsymbol{E}$ strain tensor in vector form. Tab. 2 provides this information.

The commutative diagram in Tab. 1 shows, that we arrive at equal results whether one discretizes first and then carries out linearization of the nonlinear expressions, or if the linearization is done at intermediate stages. In this work the linearization at intermediate stages is also given. Finally, the List of Symbols starting on page 163 gives access to further in-depth information.

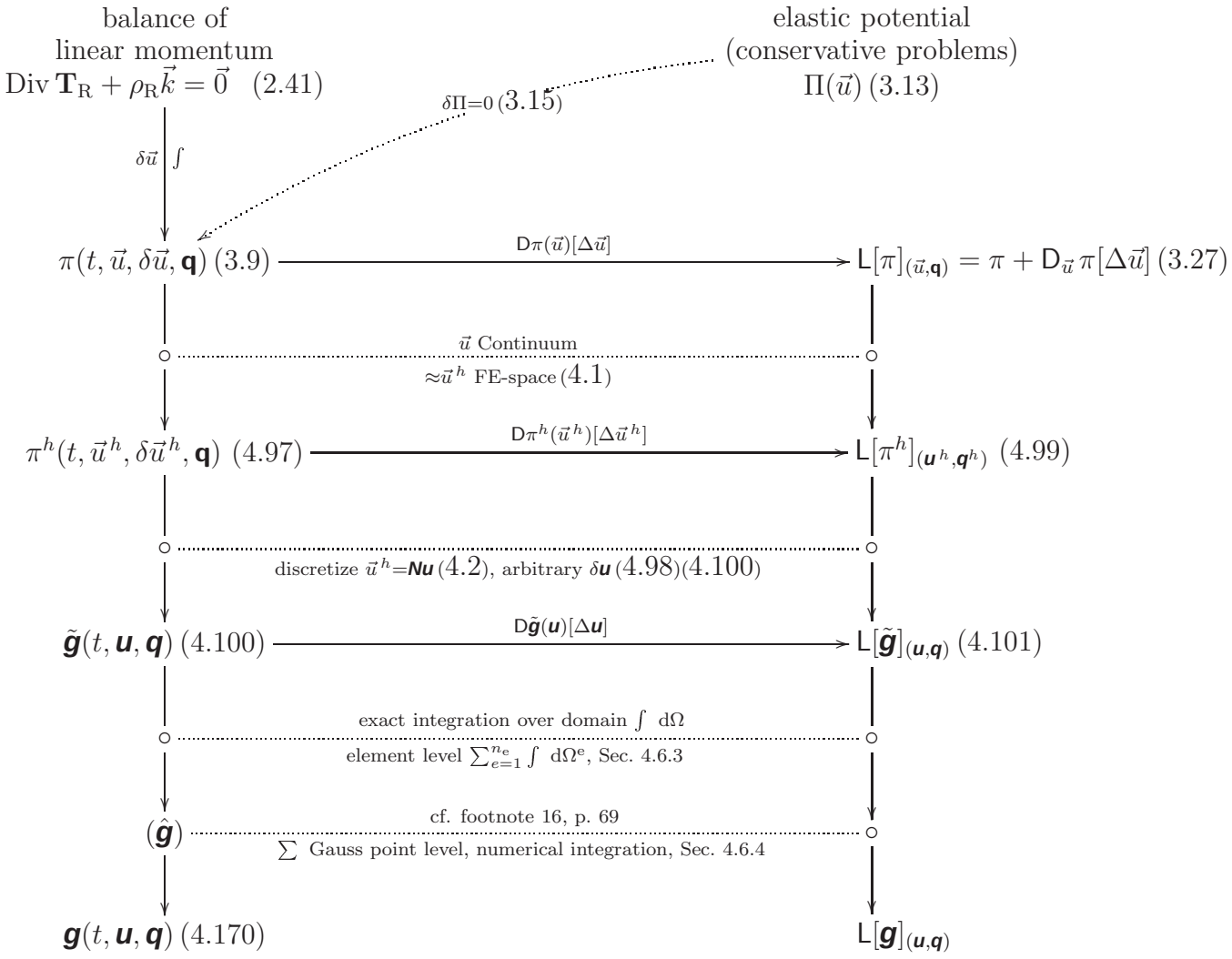


Table 1: Notation weak forms

tensor/vector	FE-space	column matrix	discretized (global level)
$\vec{u}$	$\vec{u}^h$	$\boldsymbol{u}^h\,(4.1)$	$\boldsymbol{Nu}\,(4.2)$
$\vec{n}\,(3.42)$	$\vec{n}^h$	$\boldsymbol{n}\,(4.111)$	(4.115)
$\vec{X}$		$\boldsymbol{X}$	
$\vec{x}=\vec{X}+\vec{u}$	$\vec{x}^h=\vec{X}+\vec{u}^h$	$\boldsymbol{x}$	$\boldsymbol{x}=\boldsymbol{X}+\boldsymbol{Nu}$
$\mathrm{Grad}\,\vec{u}=\mathbf{G}\,(3.8)$	$\mathrm{Grad}\,\vec{u}^h$	$\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t))\,(4.49)$	$\boldsymbol{G}\,\boldsymbol{u}\,(4.74)$
$\mathbf{F}=\mathrm{Grad}\,\vec{x}=\frac{\partial\vec{x}}{\partial\vec{X}}$	$\mathbf{F}^h$	$\boldsymbol{F}^h$	$\boldsymbol{F}$
$\mathbf{C}$	$\mathbf{C}^h$	$\boldsymbol{C}^h\,(4.63)$	$\boldsymbol{C}$
$\mathbf{E}_\ell$	$\mathbf{E}_\ell^h$	$\boldsymbol{E}_\ell^h=\boldsymbol{H}\boldsymbol{\Theta}^h\,(4.51)$	$\boldsymbol{E}_{n\ell}=\boldsymbol{HG}\,\boldsymbol{u}\,(4.82)_2$ $\boldsymbol{E}_{n\ell}=\boldsymbol{B}_\ell\,\boldsymbol{u}\,(4.83)$
$\mathbf{E}_{n\ell}$	$\mathbf{E}_{n\ell}^h$	$\boldsymbol{E}_{n\ell}^h=\frac{1}{2}\boldsymbol{M}_\Theta(\boldsymbol{\Theta}^h)\boldsymbol{\Theta}^h\,(4.51)$	$\boldsymbol{E}_{n\ell}=\frac{1}{2}\left[\boldsymbol{M}_\Theta\left(\boldsymbol{G}(\boldsymbol{u})\right)\boldsymbol{G}\right]\,\boldsymbol{u}\,(4.82)_3$ $\boldsymbol{E}_{n\ell}=\frac{1}{2}\boldsymbol{B}_{n\ell}\boldsymbol{u}\,(4.84)$
$\mathbf{E}=\mathbf{E}_\ell+\mathbf{E}_{n\ell}$	$\mathbf{E}^h$	$\boldsymbol{E}^h\,(4.46)=\left[\boldsymbol{H}+\frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h\right)\right]\,\boldsymbol{\Theta}^h\,(4.51)$	$\boldsymbol{E}=\left[\boldsymbol{B}_\ell+\frac{1}{2}\boldsymbol{B}_{n\ell}\right]\,\boldsymbol{u}=\boldsymbol{B}^*\boldsymbol{u}\,(4.85)$
$\downarrow\tilde{\boldsymbol{\Phi}}(\mathbf{C},\mathbf{q})\,(2.43)$		$\downarrow\tilde{\boldsymbol{\Phi}}^h(\boldsymbol{C}^h,\boldsymbol{q}^h)\,(4.62)$	
$\tilde{\mathbf{T}}$	$\tilde{\mathbf{T}}^h$	$\tilde{\boldsymbol{T}}^h\,(4.59)$	$\widehat{\tilde{\boldsymbol{T}}}\,(4.123)$
$\delta\mathbf{E}\,(3.7)$	$\delta\mathbf{E}^h$	$\delta\boldsymbol{E}^h\,(4.55)=\left[\boldsymbol{H}+\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h)\right)\right]\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h)\,(4.57)$	$\delta\boldsymbol{E}=\left[\boldsymbol{B}_\ell+\boldsymbol{B}_{n\ell}\right]\boldsymbol{u}=\boldsymbol{Bu}\,(4.92)$
$\Delta\mathbf{E}\,(3.33)$	$\Delta\mathbf{E}^h$	$\Delta\boldsymbol{E}^h=\left[\boldsymbol{H}+\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h)\right)\right]\boldsymbol{\Theta}^h(\Delta\boldsymbol{u}^h)\,(4.58)$	$\Delta\boldsymbol{E}=\left[\boldsymbol{B}_\ell+\boldsymbol{B}_{n\ell}\right]\boldsymbol{u}=\boldsymbol{B}\Delta\boldsymbol{u}\,(4.93)$

Table 2: Quantities in different discretization stages

Chapter 1

Introduction

Numerical methods have become indispensable tools both for engineering applications and research. An important case is the simulation of a system's response by finite element methods. It can provide further insight and in many cases minimize the number of experiments. The finite element method started as a tool for applied engineers but the mathematical analysis catched up in the mean time, deepening understanding and also spurring new developments. One of those contributions from the mathematical side goes by the name *high order finite elements* or *p-version*. The monograph by Barna Szabó and Ivo Babuška (Szabó and Babuška, 1991) on the p-version unifies the knowledge of high-order finite element methods at the beginning of the nineties of the 20th century. The p-version is known to be robust against element distortion and to perform very well in the small strain setting, also for nonlinear materials. Recently it was shown that these benefits also carry over to the finite strain case, see for example (Düster et al., 2003; Heisserer et al., 2007; Krause et al., 1995; Yosibash et al., 2007). In this thesis, both the locking-free property of the p-version for *finite strain* hyperelasticity and the application of p-FEM to a complex pressure dependent finite strain inelastic material is shown in the context of cold isostatic pressing of powder materials. This work reflects insights gained in a project funded by GIF[1]. There, the focus was on simulating cold isostatic pressing covering the full range from experiments over constitutive modeling, numerical calculation with the p-version, and finally validation of the results, compare Fig. 1.1.

1.1 The process of cold isostatic pressing (CIP)

Cold isostatic pressing is a technology used to compact powder materials, compare (German, 1998), (Koizumi and Nishihara, 1991), (Price, 1998). For a historical review compare (Papen, 1977). The specimen are often sintered after the compaction stage. Cold isostatic pressing is used for a wide range of materials, however most prevalent is the compaction of metal powder and ceramic, cf. (Richerson, 2006). The process of manufacturing suitable metal powder is an art by itself[2] as the shape of the powder particles has significant influence on the handling and properties of the pressed material.

[1]The support of the German-Israeli Foundation of Scientific Research and Development grant I-700-26.10/2001 is gratefully acknowledged.

[2]In the ASM Handbook of Powder Metal Technologies and Application, (ASM, 1998), over 170 pages are devoted to the different powder production methods for a wide varieties of metals ranging from iron over copper and titanium to tungsten and molybdenum.

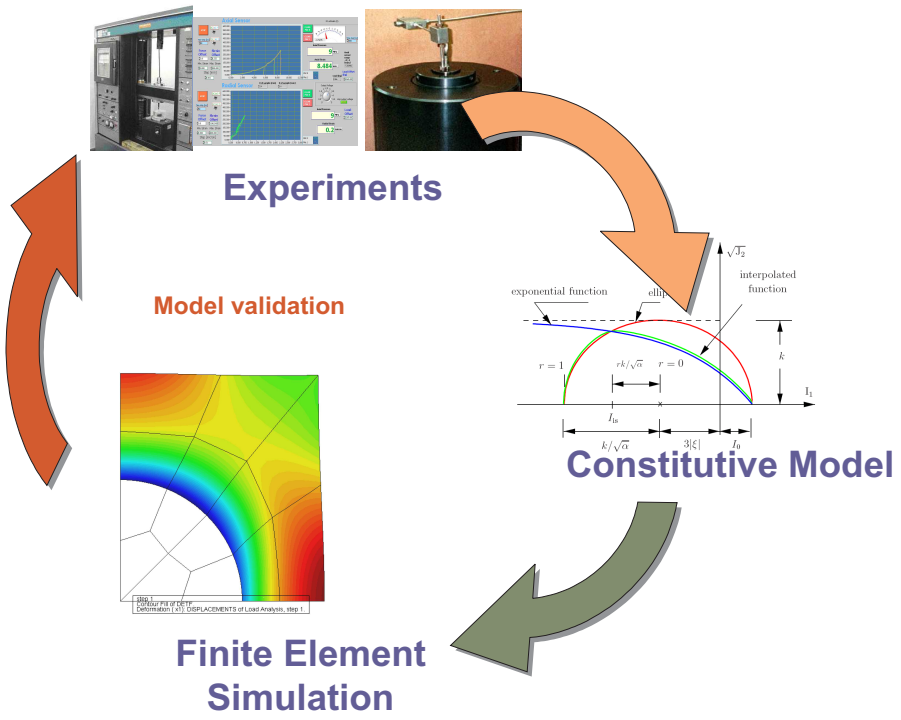


Figure 1.1: Overview of the GIF project modeling the cold isostatic pressing process

The powder is filled in a flexible mold that determines the final shape of the specimen and deforms during the process. At ambient temperature (‘*cold*’) in an *isostatic* manner the *pressure* is applied, in most cases by a liquid medium like water or oil, see Fig. 1.2. Typically the pressure is in the range of $300 - 500$ MPa. Metal powders are, for example, compacted from a relative density of about 0.4 of the full density of the solid materials to a relative density of about 0.9 of the full density by isostatic pressing. During this process the porosity is reduced and the particles are packed denser. Resulting is a brittle body where the bonding is achieved by friction.

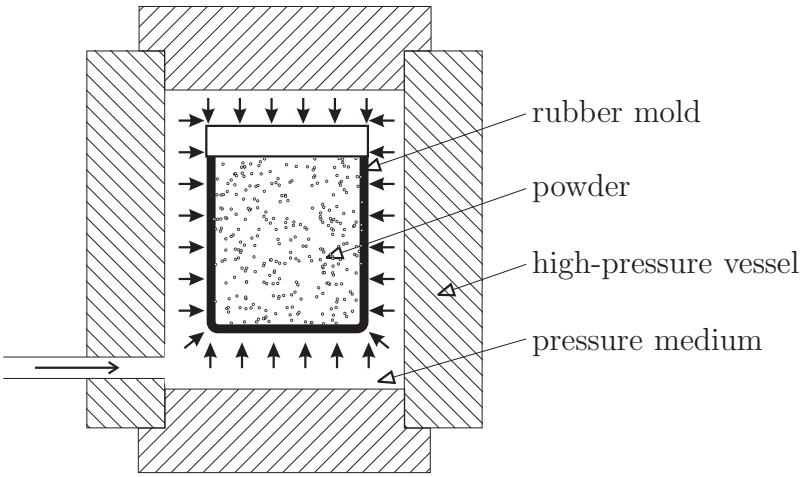


Figure 1.2: Schematic representation of wet-bag isostatic pressing

One advantage of cold isostatic pressing, cf. (Price, 1998), is that even complex shapes with undercuts can be produced. Furthermore, thin-walled sections with large aspect ratios of length to characteristic diameter can be densified, see for example the chamotte tubes used in chimneys in Fig. 1.3. Die wall friction is not an issue because of the elastic behavior of the mold. A very important point is that the density distribution in the pressed sample is very homogenous due to the isostatic pressure. This is especially important for subsequent sintering because regions with different densities are likely to cause cracks by different shrinking. An example of cold isostatically pressed titanium alloy and aluminium alloy powders is shown in Fig. 1.4. Expensive materials like titanium (Lüthering and Williams, 2003, p. 91), tungsten (Lassner and Schubert, 1999, p. 323), molybdenum, and tantalum are candidates for CIP. Also materials difficult to machine like tungsten carbide, titanium carbide, and tool steels. In the GIF-project copper powder was chosen as reference material.

Figure 1.3: Chamotte tubes produced applying cold isostatic pressing by the German company Schiedel with Loomis presses (Ceramitec trade fair in Munich 2006)

1.2 Outline of the thesis

The thesis is structured as follows. **Chapter 2** recalls basic continuum mechanics concepts, introduces the kinematic notations as well as balance principles and summarizes the material models needed in the simulation of metal powder compaction.

Chapter 3 introduces the important concept of linearization and the relationship of variation and directional derivative. This tool is subsequently applied to the nonlinear weak equation establishing the basis for numerical treatment.

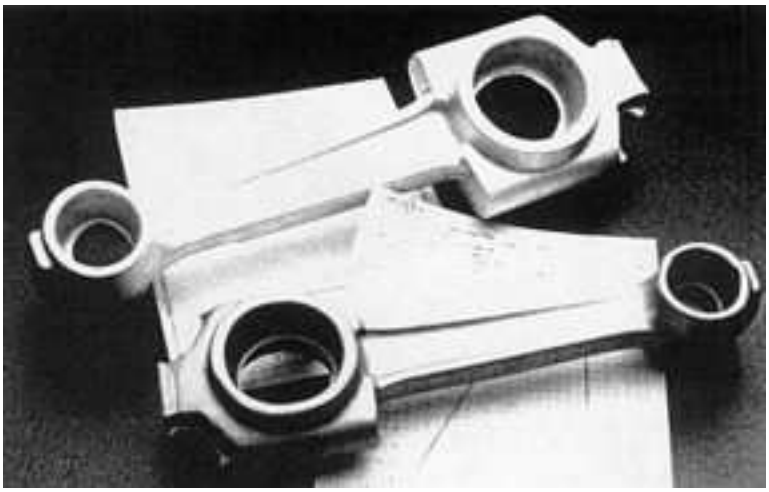


Figure 1.4: Forging preforms cold isostatically pressed with titanium alloy and aluminium powders. Dynament Technology, reproduced from (Price, 1998, p. 387)

The linearized variational equation is discretized in **Chapter 4** using high-order finite elements. Hierarchical shape functions are distinguished from standard high-order shape functions. Furthermore, different mapping concepts and their consequences are discussed. Various types of errors occurring when modeling a physical event by numerical methods are discerned and the inherent verification properties of the p-version are addressed. For the solution of the global nonlinear equation system two similar variants are kept apart: the Newton-Raphson method (NRM) and the multi-level Newton algorithm (MLNA) with a local iteration.

After the stage is set, **Chapter 5** confirms the implementation of the axisymmetric finite strain element and justifies the calculation of reaction forces when hierarchic high order shape functions are used. Next, the implementation of deformation-dependent loads (follower loading) encountered in cold isostatic pressing is verified against analytical solutions derived in this project. These analytical solutions allow to show that the locking-free property of the p-version, well known in the small strain setting, can also be found for finite strain hyperelasticity. Finally, powder metallurgical numerical examples are addressed drawing on the full arsenal of methods elaborated. We end with a 'complex' validation example where we can show good matching of the numerical simulation to the experimental result.

Chapter 6 concludes and gives an outlook for future work.

Chapter 2

Basic continuum mechanics

In the endeavor of man to understand the world the so called 'pre-socratic' philosophers mark an important stage, compare e.g. the collection (Mansfeld, 1983, 1986). They struggled to explain phenomena rejecting traditional mythologic explanations. Instead, they tried to introduce reason and logic arguments. One prominent question was what the nature of matter is. The school of the atomists, Leucippus (5th century BC) and Democritus (460–370 BC), argued that everything is composed of small, indestructible components, compare (Mansfeld, 1986, Sec. 9). These building blocks they called $\alpha\tau o\mu o\varsigma$ - indivisible. So they coined the notion 'atom'. However, in the mean time the meaning of 'atom' changed: what *we* call 'atoms' today is in turn made up from smaller components, protons, neutrons, and electrons. Modern physics shows that beyond these are even more smaller particles, physicists talk of a 'zoo' of particles and there is a world-wide hunt for the basic particles.

While we know of quantum mechanics and to some extend of the microstructure of matter, it is nevertheless sufficient for a broad range of engineering applications to apply classical mechanics and the concept of macrostructure entities. Many phenomena can be successfully described by approximating matter on a macroscopic level, thereby averaging properties over a given volume. Continuum mechanics is a phenomenological theory of fields. The term *continuum* is defined as a set of points continuously filling a given space. These points bear the material properties, like e.g. density, temperature and, velocity (Altenbach and Altenbach, 1994, p. 3).

The scope of continuum mechanics is the study of motion and deformation (kinematics), the study of stress in a continuum and the mathematical description of the models governing the motion of a continuum (balance principles) (Holzap fel, 2000, p. 55).

In this work we only give a brief introduction in the wide field of continuum mechanics, for in-depth knowledge the reader may consult monographs on this topic, for example (Altenbach and Altenbach, 1994; Haupt, 2000; Marsden and Hughes, 1993; Truesdell and Noll, 1965). Introductions to these concepts in the framework of finite element analysis can be found in (Belytschko et al., 2000; Bonet and Wood, 1997; Holzap fel, 2000; Wriggers, 2001).

2.1 Kinematics

The study of deformation and motion without reference to the cause is called *kinematics*. To this end every material point $\mathbf{P}$ is assigned to three coordinates ξ_1, ξ_2, ξ_3 (collected in the vector

$\vec{\xi}$) in the reference configuration $\mathcal{R}$ of a body $\mathcal{B}$. The point **P** moves through space in time and is labeled **X** in the reference configuration at time $t = 0$ and **x** in the current configuration. The series of configurations associated with the parameter t (time) is called motion of the point **P** and is described by the current position vector $\vec{x} = \kappa(t, \vec{\xi})$. The configuration at $t = 0$ is called reference configuration and the position vector is denoted by capital letter $\vec{X}$. Due to the construction we have an one-to-one map from the coordinates $\vec{\xi}$ of point **P** to a position vector $\vec{X}$ in space at time $t = 0$. Therefore, one can define a new function in $\vec{X}$

$$\vec{x} = \vec{\chi}_{\mathrm{R}}(\vec{X}, t) \tag{2.1}$$

to describe the motion with respect to the reference configuration. For a detailed discussion of configurations and motion in different coordinate systems it is refered to (Haupt, 2000, pp. 7–22). Compare Fig. 2.1 for a sketch of the notations. Already at this point we hint at the

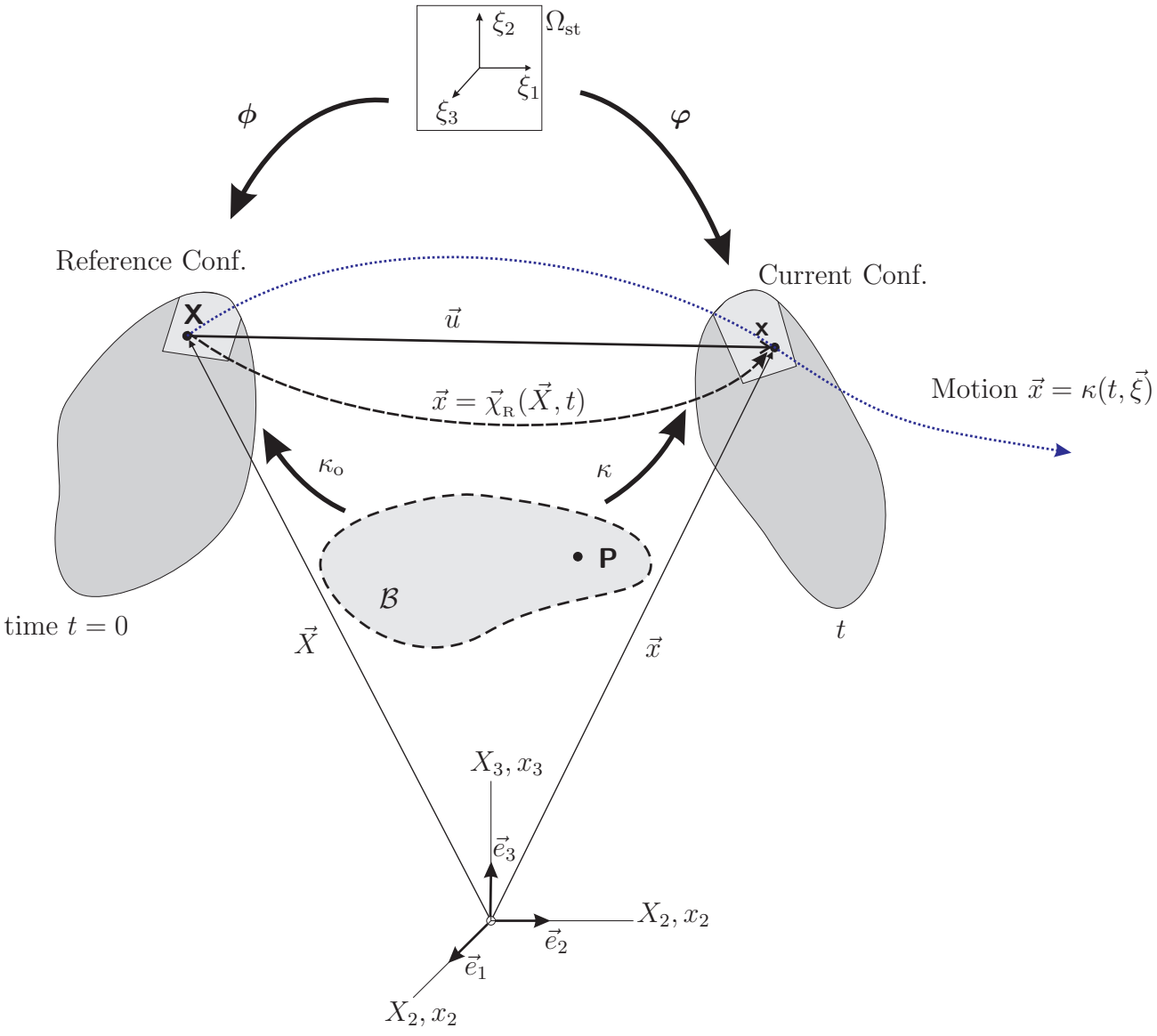


Figure 2.1: Configurations

partitioning of the domain in small regions in view of the finite element method and introduce the mapping from a standard element Ω_{st} to the reference configuration, $\vec{X} = \boldsymbol{\phi}(\vec{\xi})$ (4.30), and to the current configuration at a given time t, $\vec{x} = \boldsymbol{\varphi}(\vec{\xi})$, for later use.

We employ a total Lagrangian approach to describe the relevant tensor fields with relation to the reference configuration, i.e. in terms of the position vector $\vec{X}$. For example, the deformation at time t is defined as

$$\vec{u}(t, \vec{X}) = \vec{\chi}_{\text{R}}(\vec{X}, t) - \vec{X} \,. \tag{2.2}$$

A very fundamental quantity is the gradient of the deformation. The deformation gradient $\mathbf{F}$ is a local measure, how the deformation field in the vicinity of a point $\mathbf{P}$ is affected by the motion, or put differently, how neighboring points in the reference configuration move. It is defined as

$$\mathbf{F} = \text{Grad}\, \vec{\chi}_{\text{R}}(\vec{X}, t) = \frac{\partial x^k}{\partial X^L} \quad \vec{g}_k \otimes \vec{G}^L \tag{2.3}$$

where $x^k = \vec{\chi}_{\text{R}}(X^L, t)$ and X^L define the coordinates in the current configuration and reference configuration respectively. Furthermore, $\vec{g}_k$ denotes the tangent vectors to the isolines of coordinates in the current configuration and $\vec{G}^L$ the gradient vectors to the coordinate surfaces in the reference configuration.

The deformation gradient transforms material line elements $\text{d}\vec{X}$, i.e. tangent vectors of the coordinate lines in the reference configuration, to line elements $\text{d}\vec{x}$ of the spatial configuration. Accordingly, surface and volume elements are transformed as, cf. (Haupt, 2000, p. 28),

$$\text{d}\vec{x} = \mathbf{F}\text{d}\vec{X} \tag{2.4}$$

$$\text{d}\vec{a} = (\det \mathbf{F})\mathbf{F}^{-T}\text{d}\vec{A} \tag{2.5}$$

$$\text{d}v = (\det \mathbf{F})\text{d}V, \tag{2.6}$$

where we stipulate that the deformation is invertible and preserves the orientation, i.e.

$$\det \mathbf{F} \equiv J > 0 \,. \tag{2.7}$$

The deformation gradient can be decomposed in a number of ways.

- In the polar decomposition the motion is separated in a rigid body rotation, i.e an orthogonal rotation tensor $\mathbf{R}$ and a material stretch tensor $\mathbf{U}$ or a spatial stretch tensor $\mathbf{V}$, depending on the order of the operations

  $$\mathbf{F} = \mathbf{RU} = \mathbf{VR} \,. \tag{2.8}$$

  For a graphical interpretation of the split compare (Haupt, 2000, p. 29).

- In the volumetric decomposition we multiplicatively separate $\mathbf{F}$ into a volume-changing part $\hat{\mathbf{F}}$ and a volume preserving (or distortional or isochoric) component $\overline{\mathbf{F}}$,

  $$\mathbf{F} = \hat{\mathbf{F}}\overline{\mathbf{F}} \,. \tag{2.9}$$

  This decomposition is for example applied when developing strain energy density functions for nearly incompressible hyperelastic materials, cf. Eq. (2.60) in Sec. 2.3.1.

- A third decomposition is e.g. utilized for the constitutive model of metal powder compaction in Sec. 2.3.2. Here a multiplicative decomposition in an elastic and an inelastic part, $\hat{\mathbf{F}}_\mathrm{e}$ and $\mathbf{F}_\mathrm{v}$ respectively, is applied, see Eq. (2.69),

$$\mathbf{F} = \hat{\mathbf{F}}_\mathrm{e}\mathbf{F}_\mathrm{v}. \tag{2.10}$$

In the literature a number of further decompositions are proposed, compare the references in (Hartmann, 2003a, p. 50). All these multiplicative splits define intermediate configurations and if the splitting has physical meaning constitutive relations can be separately defined for the different physical phenomena.

The deformation gradient is used to define quantities related to the stretch of the material. The right and left Cauchy-Green tensors are introduced as

$$\mathbf{C} = \mathbf{F}^T\mathbf{F} = \mathbf{U}^2 \tag{2.11}$$
$$\mathbf{B} = \mathbf{F}\mathbf{F}^T = \mathbf{V}^2 \tag{2.12}$$

and building on them strain measures can be introduced. Strain measures are tensorial quantities that vanish under rigid body rotations and have monotonous properties, cf. (Hartmann, 2003a, p. 46). The Green-Lagrange strain tensor $\mathbf{E}$ is defined as

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T\mathbf{F} - \mathbf{I}) = \frac{1}{2}(\mathbf{C} - \mathbf{I}) = \frac{1}{2}(\mathbf{U}^2 - \mathbf{I}) \tag{2.13}$$

and a strain measure operating on the spatial configuration is the Euler-Almansi-strain tensor

$$\mathbf{A} = \frac{1}{2}(\mathbf{I} - \mathbf{F}^{-1}\mathbf{F}^{-T}) = \frac{1}{2}(\mathbf{I} - \mathbf{C}^{-1}) = \frac{1}{2}(\mathbf{I} - \mathbf{V}^{-2})\,, \tag{2.14}$$

both are connected by the relations

$$\mathbf{A} = \mathbf{F}^{-T}\mathbf{E}\mathbf{F}^{-1} \qquad \text{and} \qquad \mathbf{E} = \mathbf{F}^T\mathbf{A}\mathbf{F}\,. \tag{2.15}$$

Some materials show dependence on the deformation velocities. The relations describing the temporal rates corresponding to (2.4) – (2.6) are, see (Hartmann, 2003a, p. 47),

$$\mathrm{d}\dot{\vec{x}} = \dot{\mathbf{F}}\mathrm{d}\vec{X} = \mathbf{L}\mathrm{d}\vec{x} \tag{2.16}$$
$$\mathrm{d}\dot{\vec{a}} = J\Big((\mathbf{F}^{-1}\cdot\dot{\mathbf{F}}^T)\mathbf{I} - \mathbf{F}^{-T}\dot{\mathbf{F}}^T\Big)\mathbf{F}^{-T}\mathrm{d}\vec{A} = \Big((\mathrm{tr}\,\mathbf{L})\mathbf{I} - \mathbf{L}^T\Big)\mathrm{d}\vec{a} \tag{2.17}$$
$$\mathrm{d}\dot{v} = J(\mathbf{F}^{-1}\cdot\dot{\mathbf{F}}^T)\mathrm{d}V = (\mathrm{tr}\,\mathbf{L})\mathrm{d}v. \tag{2.18}$$

The material time derivative is denoted by the superimposed dot. The (spatial) velocity gradient is calculated from the velocity field in the spatial configuration $\vec{v}(\vec{x},t)$ as

$$\mathbf{L} = \dot{\mathbf{F}}\mathbf{F}^{-1} = \mathrm{grad}\,\vec{v}(\vec{x},t)\,. \tag{2.19}$$

We introduce the important split of $\mathbf{L}$ in the symmetric 'rate of deformation tensor' $\mathbf{D}$ and the antisymmetric 'spin tensor' $\mathbf{W}$ as

$$\mathbf{D} = \frac{1}{2}(\mathbf{L} + \mathbf{L}^T) \tag{2.20}$$
$$\mathbf{W} = \frac{1}{2}(\mathbf{L} - \mathbf{L}^T)\,. \tag{2.21}$$

The material time derivative of the Lagrangian strain tensor is obtained from (2.13) and (2.11) as

$$\dot{\mathbf{E}} = \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{E} = \frac{1}{2}\dot{\mathbf{C}} = \frac{1}{2}(\dot{\mathbf{F}}^T\mathbf{F} + \mathbf{F}^T\dot{\mathbf{F}}) \tag{2.22}$$

and by inserting Eq. (2.19) into (2.20) we arrive at

$$\mathbf{D} = \mathbf{F}^{-T}\dot{\mathbf{E}}\mathbf{F}^{-1}\,. \tag{2.23}$$

If we introduce Eq. (2.15) in (2.23) we have

$$\mathbf{D} = \mathbf{F}^{-T}\left(\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{E}\right)\mathbf{F}^{-1} = \mathbf{F}^{-T}\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{F}^T\mathbf{A}\mathbf{F})\mathbf{F}^{-1} \equiv \overset{\triangle}{\mathbf{A}}\,. \tag{2.24}$$

The quantity $\overset{\triangle}{\mathbf{A}}$ is known as the (covariant) Oldroyd rate of the Almansi strain tensor (Haupt, 2000, p. 48), (Holzap fel, 2000, p. 193). The three step operation of (i) pulling back a spatial tensor $(\bullet^*)$ to its material representation $(\bullet_*)$, (Holzap fel, 2000, p. 82),

$$(\bullet_*) = \mathbf{F}^T\,(\bullet^*)\,\mathbf{F} \tag{2.25}$$

(ii) material time derivative

$$\dot{(\bullet_*)} = \frac{\mathrm{d}}{\mathrm{d}t}(\bullet_*) \tag{2.26}$$

and (iii) push forward

$$(\bullet^*) = \mathbf{F}^{-T}\,(\bullet_*)\,\mathbf{F}^{-1} \tag{2.27}$$

is known as Lie time derivative (Bonet and Wood, 1997, p. 87), (Holzap fel, 2000, p. 106).

2.2 Balance principles and stress tensors

Balance principles

The kinematic relations introduced above give a description of motion and deformation of bodies. This section presents balance relations that relate kinematic quantities to influences 'from the outside world' (for a detailed discussion we refer to (Haupt, 2000, Ch. 2)). One can imagine dividing the world in two disjoint sets: a body (or parts of a body) and the rest of the material world (free-body principle). Under the assumption that key physical quantities, like mass, momentum and energy are continuously distributed and additive in the sense of the measure theory they can be represented by volume integrals of density functions and the exchange of those quantities over the surface of a control volume can be described by surface integrals. Balance relations establish a causal relationship of the temporal change of quantities inside the body with influences from the outside world.

Balance relations can be given in a global (integration over the whole reference volume V) or local form (for a particle), for example the global form of the balance of mass in reference configuration reads

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_V \rho_{\mathrm{R}}(\vec{X},t)\ \mathrm{d}V = 0 \tag{2.28}$$

with the density in the reference configuration ρ_{R}, while the local form is gained under the assumption that balance equations in general, and equation (2.28) in particular, are valid not only for the whole control volume but for every subvolume of V ((Haupt, 2000, Theorem 2.1, p. 77), (Belytschko et al., 2000, p. 108))

$$\frac{\mathrm{d}}{\mathrm{d}t}\rho_{\mathrm{R}}(\vec{X},t) = 0 \quad \Rightarrow \quad \rho_{\mathrm{R}}(\vec{X},t) = \rho_{\mathrm{R}}(\vec{X}) \quad . \tag{2.29}$$

And as the mass of the system will not change upon a pull back operation

$$\int_v \rho(\vec{x},t)\ \mathrm{d}v = \int_V \rho_{\mathrm{R}}(\vec{X})\ \mathrm{d}V \tag{2.30}$$

we can use the kinematic relation (2.6) with $J = \det \mathbf{F}$ to obtain

$$\rho_{\mathrm{R}} = J\rho \quad \rightarrow \quad \rho = \frac{1}{J}\rho_{\mathrm{R}}, \tag{2.31}$$

a relation that can be used to compute the current density from the determinant of the deformation gradient if the material (initial) density is known. In powder compaction processes an important quantity is the *relative density* $\rho_{\mathrm{rel}} = \rho/\rho_0$ defined as the quotient of the current density of the powder body over the density of the solid material ρ_0, e.g. tap density of copper powder over the density of solid copper (see (ISO/TC119, 1995)). As a consequence of Eq. (2.31) the relative density at time t can be calculated from the initial relative density at time $t = 0$ and the deformation gradient at time t as

$$\rho_{\mathrm{rel}}(\vec{x},t) = \frac{\rho(\vec{x},t)}{\rho_0} = \frac{1}{J}\frac{\rho_{\mathrm{R}}(\vec{X})}{\rho_0} = \frac{1}{J}\rho_{\mathrm{R}_{\mathrm{rel}}}(\vec{X}). \tag{2.32}$$

Summary of global balance equations. The thermodynamic balance relations can be written in unified form ((Haupt, 2000, p. 137), (Altenbach and Altenbach, 1994, p. 118), (Holzap fel, 2000, Sec. 4.7)) that was tabulated in (Hartmann, 2003a, p. 57) as the temporal change of a scalar (ψ) or vector valued ($\vec{\psi}$) variable that is related to a surface transport term ($\phi, \vec{\phi}$), a volume transport contribution ($\sigma, \vec{\sigma}$) and production term ($p, \vec{p}$),

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\int_v \psi(\vec{x},t)\rho(\vec{x},t)\ \mathrm{d}v &= \int_a \phi(\vec{x},t)\ \mathrm{d}a + \int_v (\sigma(\vec{x},t) + p(\vec{x},t))\ \mathrm{d}v \\
\frac{\mathrm{d}}{\mathrm{d}t}\int_v \vec{\psi}(\vec{x},t)\rho(\vec{x},t)\ \mathrm{d}v &= \int_a \vec{\phi}(\vec{x},t)\ \mathrm{d}a + \int_v (\vec{\sigma}(\vec{x},t) + \vec{p}(\vec{x},t))\ \mathrm{d}v,
\end{aligned} \tag{2.33}$$

here, $\rho(\vec{x},t)$ is the mass density of the body. Using Cauchy's lemma that stipulates a linear dependency of the surface-distributed interaction from the surface normal $\vec{n}$ and the flux,

	$\psi, \vec{\psi}$	$\vec{\Phi}, \mathbf{\Phi}$	$\sigma, \vec{\sigma}$	$p, \vec{p}$
mass	1	$\vec{0}$	0	0
linear momentum	$\vec{v}$	$\mathbf{T}$	$\rho\vec{k}$	$\vec{0}$
angular momentum	$(\vec{x}-\vec{x}_0)\times\vec{v}$	$(\vec{x}-\vec{x}_0)\times\mathbf{T}$	$(\vec{x}-\vec{x}_0)\times\rho\vec{k}$	$\vec{0}$
energy	$e+\frac{1}{2}\vec{v}\cdot\vec{v}$	$\mathbf{T}^T\vec{v}-\vec{q}$	$\rho(\vec{k}\cdot\vec{v}+r)$	0
entropy	s	$-\frac{1}{\theta}\vec{q}$	$\frac{r}{\theta}\rho$	$\gamma\geq 0$
mechanical energy	$\frac{1}{2}\vec{v}\cdot\vec{v}$	$\mathbf{T}^T\vec{v}$	$\rho\vec{k}\cdot\vec{v}-\mathbf{T}\cdot\mathbf{D}$	0

Table 2.1: Terms in the general balance relation

$\phi = \vec{\Phi}\cdot\vec{n}, \vec{\phi} = \mathbf{\Phi}\vec{n}$, the terms of the general balance equation (2.33) can be specified as collected in Tab. 2.1 with the fluxes $\vec{\Phi}, \mathbf{\Phi}$.

The mechanical energy is not an independent balance relation but the result of mass, linear momentum and angular momentum balance. However, it fits in the structure of the general balance equation and was thus added. The quantities in Tab. 2.1 are defined on the current configuration and have the following meaning: spatial velocity $\vec{v}(\vec{x},t)$, Cauchy stress tensor $\mathbf{T}(\vec{x},t)$, strain rate tensor $\mathbf{D}(\vec{x},t)$, density $\rho(\vec{x},t)$, volume or body force $\rho\vec{k}$, $\vec{x}_0$ as point of reference for the angular momentum, specific internal energy $e(\vec{x},t)$, heat flux $\vec{q}(\vec{x},t)$ and a spatially distributed heat supply $r(\vec{x},t)$, density of entropy $s(\vec{x},t)$ as well as a non-negative production of entropy $\gamma(\vec{x},t)\geq 0$. The absolute temperature is denoted by $\theta > 0$.

Local balance equations. Using the transport and divergence theorem the local forms of the balance equations can be derived (see, for example, (Haupt, 2000, Sec. 3.5.3), (Wriggers, 2006, Sec. 3.2), (Belytschko et al., 2000, Sec. 3.5 and Sec. 3.6)).

The *spatial representation* of the local balance equations consists of the conservation of mass

$$\dot{\rho}(\vec{x},t) + \rho \operatorname{div}\vec{v}(\vec{x},t) = 0, \tag{2.34}$$

the conservation of linear momentum

$$\operatorname{div}\mathbf{T}(\vec{x},t) + \rho\vec{k} = \rho\frac{\mathrm{d}\vec{v}(\vec{x},t)}{\mathrm{d}t}, \tag{2.35}$$

and the conservation of rotational momentum

$$\mathbf{T} = \mathbf{T}^T \tag{2.36}$$

manifesting the symmetry of the Cauchy stress tensor.

To express the balance of linear momentum in the material configuration Eq. (2.35) is pulled back to the reference configuration. We introduce the first Piola-Kirchhoff stress tensor

$$\mathbf{T}_\mathrm{R} = J\mathbf{T}\mathbf{F}^{-T} \tag{2.37}$$

to obtain

$$\operatorname{Div}\mathbf{T}_\mathrm{R}(\vec{X},t) + \rho_\mathrm{R}\vec{k} = \rho_\mathrm{R}\frac{\mathrm{d}\vec{v}(\vec{X},t)}{\mathrm{d}t}. \tag{2.38}$$

Remark 1 *A note concerning the notation: The differential operators starting with capitals, viz* Div *and* Grad *, denote differentiation with respect to coordinates in the reference configuration, e.g.* $\mathrm{Div}\,\vec{v} = \frac{\partial v_1}{\partial X_1} + \frac{\partial v_2}{\partial X_2} + \frac{\partial v_3}{\partial X_3}$ *while* div *,* grad *designate differentiation with respect to coordinates in the current configuration, e.g.* $\mathrm{div}\,\vec{v} = \frac{\partial v_1}{\partial x_1} + \frac{\partial v_2}{\partial x_2} + \frac{\partial v_3}{\partial x_3}$.

The material form of the balance equation of rotational momentum, cf. (2.35),

$$\mathbf{T}_{\mathrm{R}}\mathbf{F}^T = \mathbf{F}\mathbf{T}_{\mathrm{R}}^T \tag{2.39}$$

reveals that the first Piola Kirchhoff tensor is unsymmetrical and a two-field tensor, with one base vector belonging to the spatial configuration, the other to the reference configuration. To be able to work with a symmetric stress tensor the first Piola-Kirchhoff tensor is multiplied from the left with the inverse of the deformation gradient yielding the second Piola-Kirchhoff stress tensor $\tilde{\mathbf{T}}$, a symmetric stress tensor operating totally on the reference configuration

$$\tilde{\mathbf{T}} = \mathbf{F}^{-1}\mathbf{T}_{\mathrm{R}} = J\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}. \tag{2.40}$$

A number of commonly used stress tensors and their relation is compiled in Tab. 2.2 for later reference.

	$\mathbf{T}_{\mathrm{R}}$	$\tilde{\mathbf{T}}$	$\mathbf{S}$	$\mathbf{T}$
$\mathbf{T}_{\mathrm{R}}$		$\mathbf{F}\tilde{\mathbf{T}}$	$\mathbf{S}\mathbf{F}^{-T}$	$J\mathbf{T}\mathbf{F}^{-T}$
$\tilde{\mathbf{T}}$	$\mathbf{F}^{-1}\mathbf{T}_{\mathrm{R}}$		$\mathbf{F}^{-1}\mathbf{S}\mathbf{F}^{-T}$	$J\mathbf{F}^{-1}\mathbf{T}\mathbf{F}^{-T}$
$\mathbf{S}$	$\mathbf{T}_{\mathrm{R}}\mathbf{F}^T = \mathbf{F}\mathbf{T}_{\mathrm{R}}^T$	$\mathbf{F}\tilde{\mathbf{T}}\mathbf{F}^T$		$J\mathbf{T}$
$\mathbf{T}$	$\frac{1}{J}\mathbf{T}_{\mathrm{R}}\mathbf{F}^T = \frac{1}{J}\mathbf{F}\mathbf{T}_{\mathrm{R}}^T$	$\frac{1}{J}\mathbf{F}\tilde{\mathbf{T}}\mathbf{F}^T$	$\frac{1}{J}\mathbf{S}$	

1st Piola-Kirchhoff stress $\mathbf{T}_{\mathrm{R}}$, 2nd Piola-Kirchhoff stress $\tilde{\mathbf{T}}$, weighted Cauchy or Kirchhoff stress $\mathbf{S}$, Cauchy stress $\mathbf{T}$

Table 2.2: Conversion of commonly used stress tensors (Hartmann, 2001, p. 59)

Governing equations

For the numerical treatment of structural problems the local balance equations with appropriate boundary conditions have to be solved. The conservation of angular momentum is ensured by construction of the constitutive equations. For solid mechanical problems the mass balance is guaranteed by the fact that the control volume coincides with the body under consideration and there is no mass flux over the surface as well as no distributed production of mass. Restricting ourself to isothermal processes only the balance of linear momentum Eq. (2.35) or Eq. (2.38) has to be considered in particular.

Further assuming *quasi-static* processes the acceleration terms in the linear momentum equations are dropped, however the momentum equation still depends on time as the stresses (constitutive equations) and boundary conditions might be time-dependent (Hartmann, 2003a, p. 103). In the material reference frame the local momentum balance Eq. (2.38) reduces under the mentioned assumptions to

$$\mathrm{Div}\,\mathbf{T}_{\mathrm{R}}(\vec{X}, t) + \rho_{\mathrm{R}}(\vec{X})\vec{k} = \vec{0}. \tag{2.41}$$

According to the transformations collected in Tab. 2.2 this relation can be expressed using the symmetric second Piola-Kirchhoff-Tensor $\tilde{\mathbf{T}}$ as

$$\mathrm{Div}\left(\mathbf{F}(\vec{X},t)\,\tilde{\mathbf{T}}(\vec{X},t)\right)+\rho_{\mathrm{R}}(\vec{X})\vec{k}=\vec{0}. \tag{2.42}$$

To determine the unknown motion this equation has to be satisfied for every material point $\vec{X}$ and every time t for the time-dependent geometric and force boundary conditions.

2.3 Constitutive models

So far we have introduced the kinematics and the balance principles including stresses. The fields and influences can be arranged in a so-called Tonti diagram[1], Fig. 2.2, where the grey boxes represent the unknown fields and their relations. To link the left and right side of the

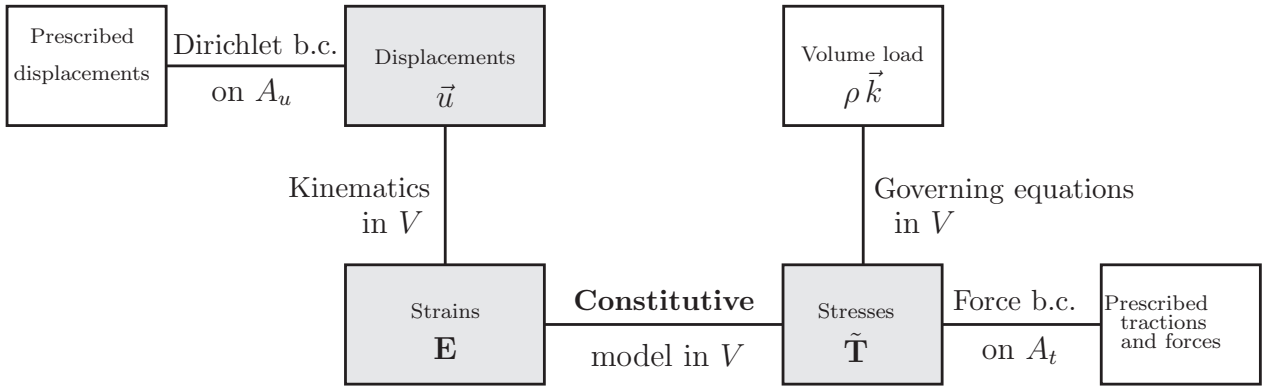


Figure 2.2: Tonti-Diagram: basic structure of the equations

diagram relations between stress and strain are needed. Those relations are *constitutive models* and they depend on the materials involved. At this point we complete the set of equations by assuming a fairly general constitutive model following (Hartmann, 2003a). Later on some specific constitutive relations for hyperelasticity and metal powder compaction are introduced. The constitutive relations for the stress state $\tilde{\mathbf{T}}$ depend on the deformation, represented by the symmetric right Cauchy-Green strain tensor $\mathbf{C}=\mathbf{F}^T\mathbf{F}$, and internal variables $\mathbf{q}$ evolving according to the process history:

$$\tilde{\mathbf{T}}(\vec{X},t) = \tilde{\mathbf{\Phi}}\left(\mathbf{C}(\vec{X},t),\mathbf{q}(\vec{X},t)\right) \tag{2.43}$$

$$\dot{\mathbf{q}}(\vec{X},t) = \mathbf{r}\left(\mathbf{C}(\vec{X},t),\mathbf{q}(\vec{X},t)\right) \tag{2.44}$$

obeying the initial conditions $\mathbf{q}(\vec{X},t_0)=\mathbf{q}_0(\vec{X})$. Furthermore, the initial conditions for the motion or equivalently for the displacement field $\vec{u}(\vec{X},t)=\vec{u}_0(\vec{X})$ have to be satisfied.

Many inelastic material models exhibit the structure of Eq. (2.43) – (2.44). In the case of inelastic models with yield function the internal variables $\mathbf{q}$ evolve only if a loading condition

[1]The Italian physicist Enzo Tonti ($*$ 1935) introduced classification diagrams for physical theories to study the analogies in physics. We refer to one of his early publications on this subject, (Tonti, 1972), and a current working paper that summarizes his research on the analogies of variational principles, (Tonti, 2003).

is fulfilled. Usually in the numerical treatment the plastic multiplier λ is not evaluated by the consistency condition. Instead λ is used as an additional variable and the yield condition $f(\mathbf{C}, \mathbf{q}) = 0$ is exploited to arrive locally at a differential-algebraic (DAE) system, cf. (Brenan et al., 1996),

$$\dot{\overline{\mathbf{q}}} = \lambda \overline{\mathbf{r}}(\mathbf{C}, \overline{\mathbf{q}}) \tag{2.45}$$

$$0 = f(\mathbf{C}, \overline{\mathbf{q}}) \tag{2.46}$$

abbreviated by

$$\mathbf{A}\dot{\mathbf{q}} - \mathbf{r}\left(\mathbf{C}(\vec{X}, t), \mathbf{q}(\vec{X}, t)\right) = \mathbf{0} \quad , \tag{2.47}$$

with $\mathbf{q}^T = \{\overline{\mathbf{q}}, \lambda\}$, $\mathbf{r}^T = \{\lambda\overline{\mathbf{r}}, f\}$, where the upper left part of the singular matrix $\mathbf{A}$

$$\mathbf{A} = \begin{bmatrix} \mathbf{I} & \\ & 0 \end{bmatrix} \tag{2.48}$$

recovers the evolution equations (2.44), while the lower right zero relates to the algebraic equation that governs the yield condition. In the viscoelastic case, $\mathbf{A} = \mathbf{I}$, and (2.47) is simplified to the form of Eq (2.44).

For investigations of the efficient solution of DAE systems with modern methods developed in numerical mathematics we refer to the following references and the literature cited therein (Ellsiepen and Hartmann, 2001; Hartmann, 2002, 2003a, 2004, 2005; Hartmann and Bier, 2008; Hartmann and Wensch, 2007). Hyperelastic constitutive relations show a simpler structure as no internal variables are involved.

2.3.1 Hyperelasticity

For a hyperelastic material the stresses are derived from a stored energy functional. More specifically, the work done by stresses during a deformation is path-independent, i.e. it depends only on the initial state at time t_0 and the final configuration at time t. Consequently a stored strain energy function or elastic potential Ψ depends only on the current deformation $\mathbf{F}$. A further restriction is given by the demand of objectivity – the stored strain energy must be invariant to rigid body rotations. In the view of the polar decomposition of the deformation gradient, Eq. (2.8), only the stretch must contribute. Therefore, the elastic potential is often defined as a function of the right Cauchy-Green tensor $\mathbf{C}$, Eq. (2.11), or Green strain $\mathbf{E}$, Eq. (2.13),

$$\hat{\Psi}(\mathbf{F}(\vec{X})) = \Psi(\mathbf{C}(\vec{X})) = \tilde{\Psi}(\mathbf{E}(\vec{X}))\,. \tag{2.49}$$

The stress is obtained as the derivative of the potential in direction of the deformation, i.e.

$$\tilde{\mathbf{T}} = \tilde{\mathbf{g}}(\mathbf{E}) = \rho_{\mathrm{R}} \frac{\mathrm{d}\tilde{\Psi}(\mathbf{E})}{\mathrm{d}\mathbf{E}} \tag{2.50}$$

$$= \mathbf{g}(\mathbf{C}) = 2\rho_{\mathrm{R}} \frac{\mathrm{d}\Psi(\mathbf{C})}{\mathrm{d}\mathbf{C}} \tag{2.51}$$

and the fourth-order elasticity tensor $\tilde{\mathcal{C}}$ valid at a state $\vec{u}$ connecting an increment in strain $\mathsf{D}_{\vec{u}}\mathbf{E}(\vec{u})[\Delta\vec{u}]$ due to a deformation $\Delta\vec{u}$ to an increment in stress $\mathsf{D}_{\vec{u}}\tilde{\mathbf{T}}(\vec{u})[\Delta\vec{u}]$,

$$\mathsf{D}_{\vec{u}}\tilde{\mathbf{T}}(\vec{u})[\Delta\vec{u}] = \tilde{\mathcal{C}}\mathsf{D}_{\vec{u}}\mathbf{E}(\vec{u})[\Delta\vec{u}] \tag{2.52}$$

is given by

$$\tilde{\mathcal{C}} = \frac{\mathrm{d}\tilde{\mathbf{g}}(\mathbf{E})}{\mathrm{d}\mathbf{E}} = 2\frac{\mathrm{d}\mathbf{g}(\mathbf{C})}{\mathrm{d}\mathbf{C}} = 4\rho_{\mathrm{R}}\frac{\mathrm{d}^2\Psi}{\mathrm{d}\mathbf{C}\mathrm{d}\mathbf{C}}\,. \tag{2.53}$$

For the definition of the directional derivatives, e.g. $\mathsf{D}_{\vec{u}}\tilde{\mathbf{T}}(\vec{u})[\Delta\vec{u}]$, compare Eq. (3.21).

Remark 2 *The factors* 2 *and* 4 *in the equations above originate from the application of the chain rule, where we used the representation*

$$\mathbf{C} = 2\mathbf{E} + \mathbf{I}\,, \tag{2.54}$$

compare Eq. (2.13)$_2$. These factors of course also show up in the discretized quantities.

Remark 3 *The simplest example of a hyperelastic constitutive model is the St.Venant-Kirchhoff model with the strain energy function*

$$\tilde{\Psi}(\mathbf{E}) = \frac{1}{2}\lambda(\operatorname{tr}\mathbf{E})^2 + \mu\mathbf{E}\cdot\mathbf{E} \tag{2.55}$$

where λ and μ are material coefficients and $\operatorname{tr}\mathbf{A} = a^k_k$ denotes the trace operator. The stress tensor is obtained according to Eq. (2.50) as

$$\tilde{\mathbf{T}} = \lambda(\operatorname{tr}\mathbf{E})\mathbf{I} + 2\mu\mathbf{E}\,. \tag{2.56}$$

See (Bonet and Wood, 1997, p. 120), (Wriggers, 2001, pp. 45),(Bathe, 2002, p. 589), (Holzap fel, 2000, pp. 250). It is a classical finite strain model for compressible hyperelastic materials that is often used for metals. Note that this model is suitable for large displacements but not for large compressive strains. The reason is a horizontal tangent in the stress-strain curve, i.e. zero stiffness in the compression domain as we discuss in Sec. 4.9.2.

If further knowledge about the material is given, this can be used to restrict the strain energy density functions even more. For the important class of isotropic materials like steel or rubber the energy potential can be expressed using the invariants of the right Cauchy-Green tensor $\mathbf{C}$.

In current finite element codes three types of polynomial assumptions of the strain energy functions are commonly used. The models of Rivlin and Saunders (Rivlin and Saunders, 1951), of Ogden (Ogden, 1972) and of Arruda and Boyce (Arruda and Boyce, 1993) were first formulated for the case of incompressibility, i.e. with the constraint $(\det\mathbf{F})^2 = J^2 = \mathrm{III}_{\mathsf{C}} = 1$. This means, the third invariant $\mathrm{III}_{\mathsf{C}} = \det\mathbf{C}$ of the right Cauchy-Green tensor $\mathbf{C}$ that describes volumetric change is not present in the equations. For example, the generalized polynomial Ansatz of Rivlin and Saunders states

$$w(\mathrm{I}_{\mathsf{C}}, \mathrm{II}_{\mathsf{C}}) = \sum_{i=0}^{m}\sum_{j=0}^{n} c_{ij}(\mathrm{I}_{\mathsf{C}} - 3)^i(\mathrm{II}_{\mathsf{C}} - 3)^j\,. \tag{2.57}$$

Depending on the coefficients c_{ij} and the parameters m,n a number of well known material models are recovered, for example the Neo-Hooke (Rivlin, 1948) model with a positive constant c_{10}, ($c_{00} = 0$, $m = 1$, $n = 0$),

$$w_{\mathrm{NH}}(\mathrm{I}_{\mathbf{C}}) = c_{10}(\mathrm{I}_{\mathbf{C}} - 3) \tag{2.58}$$

or the Mooney-Rivlin model (Mooney, 1940), (Rivlin and Saunders, 1951) with positive c_{10}, c_{01}, ($c_{00} = c_{11} = 0, m = 1, n = 1$),

$$w_{\mathrm{MR}}(\mathrm{I}_{\mathbf{C}}, \mathrm{II}_{\mathbf{C}}) = c_{10}(\mathrm{I}_{\mathbf{C}} - 3) + c_{01}(\mathrm{II}_{\mathbf{C}} - 3)\,. \tag{2.59}$$

For a survey compare (Hartmann, 2001).

To model *compressible* material behavior, i.e. $\det \mathbf{F} \neq 1$, the strain energy functions of incompressible hyperelastic materials can be augmented by a term depending on $J = \det \mathbf{F}$.

To this end the deformation gradient $\mathbf{F} = \mathrm{Grad}\,\vec{\chi}_{\mathrm{R}}(\vec{X}, t)$ is multiplicatively decomposed into a volume-changing part $\hat{\mathbf{F}}$ and a volume-preserving part $\overline{\mathbf{F}}$,

$$\mathbf{F} = \hat{\mathbf{F}}\overline{\mathbf{F}}, \tag{2.60}$$

where

$$\hat{\mathbf{F}} = J^{1/3}\mathbf{I}, \qquad \det \hat{\mathbf{F}} = \det \mathbf{F} \equiv J, \tag{2.61}$$

$$\overline{\mathbf{F}} = J^{-1/3}\mathbf{F}, \qquad \det \overline{\mathbf{F}} = 1. \tag{2.62}$$

This formulation goes back to Flory (Flory, 1961). Applying this decomposition one can define, for instance, the unimodular Right or Left Cauchy-Green tensors $\overline{\mathbf{C}}$ and $\overline{\mathbf{B}}$, respectively,

$$\overline{\mathbf{C}} = \overline{\mathbf{F}}^T\overline{\mathbf{F}} = J^{-2/3}\mathbf{C} \qquad \text{and} \qquad \overline{\mathbf{B}} = \overline{\mathbf{F}}\,\overline{\mathbf{F}}^T = J^{-2/3}\mathbf{B}. \tag{2.63}$$

This specific decomposition is used to formulate strain energy functions which have the decoupled form

$$\psi(J, \overline{\mathbf{C}}) = U(J) + \hat{w}(\overline{\mathbf{C}}), \tag{2.64}$$

i.e. the part $U(J)$ depends on the volume-changing deformation and the other part $\hat{w}(\overline{\mathbf{C}})$ is determined by the volume-preserving (dilatational) deformation.

Hartmann and Neff (Hartmann and Neff, 2003) proposed a class of generalized polynomial-type elasticity

$$\hat{w}(\mathrm{I}_{\overline{\mathbf{C}}}, \mathrm{II}_{\overline{\mathbf{C}}}) = \alpha(\mathrm{I}_{\overline{\mathbf{C}}}^3 - 3^3) + \sum_{i=1}^{m} c_{i0}(\mathrm{I}_{\overline{\mathbf{C}}} - 3)^i + \sum_{j=1}^{n} c_{0j}(\mathrm{II}_{\overline{\mathbf{C}}}^{3/2} - 3\sqrt{3})^j, \tag{2.65}$$

with $c_{i0} \geq 0$ and $c_{0j} \geq 0$ together with the strain energy function

$$U(J) = \frac{K}{50}(J^5 + J^{-5} - 2) \tag{2.66}$$

where K denotes the bulk modulus of the material. In a simple compressible Neo-Hookean model the compressible part is chosen as

$$U(J) = \frac{K}{2}(J - 1)^2 \tag{2.67}$$

and the resulting strain energy function evaluates to

$$\psi(J, \overline{\mathbf{C}}) = \frac{K}{2}(J-1)^2 + c_{10}(\mathrm{I}_{\overline{\mathbf{C}}} - 3)\,. \tag{2.68}$$

This material model is for example used in Sec. 5.2.2 and in Sec. 5.4.3 to model the mold in rubber isostatic pressing of metal powders.

2.3.2 Powder plasticity

As this thesis is concerned with the simulation of metal powder compaction we give a short introduction in the constitutive model. To recap the constitutive relations of the rate-dependent finite strain plasticity model, the kinematic relations are first summarized. The constitutive model proposed in (Bier and Hartmann, 2006) is based on the multiplicative decomposition of the deformation gradient (2.3),

$$\mathbf{F} = \hat{\mathbf{F}}_\mathrm{e}\mathbf{F}_\mathrm{v}, \qquad \det\mathbf{F} > 0,\ \det\hat{\mathbf{F}}_\mathrm{e} > 0,\ \det\mathbf{F}_\mathrm{v} > 0\,. \tag{2.69}$$

$\hat{\mathbf{F}}_\mathrm{e}$ defines the elastic and $\mathbf{F}_\mathrm{v}$ the inelastic part inducing the so-called inelastic intermediate configuration. A fictitious unloading leads to the inelastic Green strain tensor $\mathbf{E}_\mathrm{v} = 1/2(\mathbf{F}_\mathrm{v}^T\mathbf{F}_\mathrm{v} - \mathbf{I})$, where $\mathbf{E} = 1/2(\mathbf{F}^T\mathbf{F} - \mathbf{I})$ defines the Green strain tensor of the total deformation. The elastic strains are obtained by the difference $\mathbf{E}_\mathrm{e} = \mathbf{E} - \mathbf{E}_\mathrm{v}$, $\mathbf{E}_\mathrm{e} = 1/2(\mathbf{F}^T\mathbf{F} - \mathbf{F}_\mathrm{v}^T\mathbf{F}_\mathrm{v})$. The push-forward operation, cf. Eq. (2.27), of the strain-like tensorial variables relative to the reference configuration into quantities relative to the inelastic intermediate configuration reads

$$\hat{\mathbf{\Gamma}} = \mathbf{F}_\mathrm{v}^{-T}\mathbf{E}\mathbf{F}_\mathrm{v}^{-1}, \quad \hat{\mathbf{\Gamma}}_\mathrm{e} = \mathbf{F}_\mathrm{v}^{-T}\mathbf{E}_\mathrm{e}\mathbf{F}_\mathrm{v}^{-1}, \quad \hat{\mathbf{\Gamma}}_\mathrm{v} = \mathbf{F}_\mathrm{v}^{-T}\mathbf{E}_\mathrm{v}\mathbf{F}_\mathrm{v}^{-1} \tag{2.70}$$

leading to

$$\hat{\mathbf{\Gamma}} = \hat{\mathbf{\Gamma}}_\mathrm{e} + \hat{\mathbf{\Gamma}}_\mathrm{v}, \tag{2.71}$$

$$\hat{\mathbf{\Gamma}}_\mathrm{e} = \frac{1}{2}\left(\hat{\mathbf{F}}_\mathrm{e}^T\hat{\mathbf{F}}_\mathrm{e} - \mathbf{I}\right), \tag{2.72}$$

$$\hat{\mathbf{\Gamma}}_\mathrm{v} = \frac{1}{2}\left(\mathbf{I} - \mathbf{F}_\mathrm{v}^{-T}\mathbf{F}_\mathrm{v}^{-1}\right), \tag{2.73}$$

whereas the transformation of the material time derivatives of the Lagrangean quantities, $\dot{\mathbf{E}} = \dot{\mathbf{E}}_\mathrm{e} + \dot{\mathbf{E}}_\mathrm{v}$,

$$\overset{\triangle}{\hat{\mathbf{\Gamma}}} = \mathbf{F}_\mathrm{v}^{-T}\dot{\mathbf{E}}\mathbf{F}_\mathrm{v}^{-1}, \quad \overset{\triangle}{\hat{\mathbf{\Gamma}}}_\mathrm{e} = \mathbf{F}_\mathrm{v}^{-T}\dot{\mathbf{E}}_\mathrm{e}\mathbf{F}_\mathrm{v}^{-1}, \quad \overset{\triangle}{\hat{\mathbf{\Gamma}}}_\mathrm{v} = \mathbf{F}_\mathrm{v}^{-T}\dot{\mathbf{E}}_\mathrm{v}\mathbf{F}_\mathrm{v}^{-1} \tag{2.74}$$

yield an additive decomposition

$$\overset{\triangle}{\hat{\mathbf{\Gamma}}} = \overset{\triangle}{\hat{\mathbf{\Gamma}}}_\mathrm{e} + \overset{\triangle}{\hat{\mathbf{\Gamma}}}_\mathrm{v} = \dot{\hat{\mathbf{\Gamma}}} + \mathbf{L}_\mathrm{v}^T\hat{\mathbf{\Gamma}} + \hat{\mathbf{\Gamma}}\mathbf{L}_\mathrm{v}, \tag{2.75}$$

$$\overset{\triangle}{\hat{\mathbf{\Gamma}}}_\mathrm{e} = \dot{\hat{\mathbf{\Gamma}}}_\mathrm{e} + \mathbf{L}_\mathrm{v}^T\hat{\mathbf{\Gamma}}_\mathrm{e} + \hat{\mathbf{\Gamma}}_\mathrm{e}\mathbf{L}_\mathrm{v}, \tag{2.76}$$

$$\overset{\triangle}{\hat{\mathbf{\Gamma}}}_\mathrm{v} = \mathbf{D}_\mathrm{v} = \dot{\hat{\mathbf{\Gamma}}}_\mathrm{v} + \mathbf{L}_\mathrm{v}^T\hat{\mathbf{\Gamma}}_\mathrm{v} + \hat{\mathbf{\Gamma}}_\mathrm{v}\mathbf{L}_\mathrm{v} \tag{2.77}$$

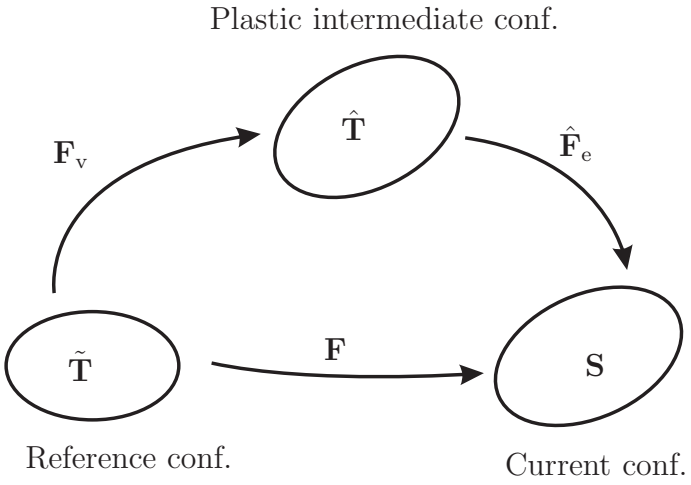


Figure 2.3: Plastic intermediate configuration

of Oldroyd-derivatives relative to the inelastic intermediate configuration, see, for example, (Haupt, 2000; Haupt and Tsakmakis, 1989, 1996). The configurations involved are sketched in Fig. 2.3.

In Eqs. (2.75) - (2.77), the inelastic velocity gradient tensor $\mathbf{L}_\mathrm{v}$ and its symmetric part $\mathbf{D}_\mathrm{v}$

$$\mathbf{L}_\mathrm{v} = \dot{\mathbf{F}}_\mathrm{v}\mathbf{F}_\mathrm{v}^{-1}, \tag{2.78}$$

$$\mathbf{D}_\mathrm{v} = \frac{1}{2}\left(\mathbf{L}_\mathrm{v} + \mathbf{L}_\mathrm{v}^T\right) \tag{2.79}$$

are applied. Furthermore, the elastic and inelastic right Cauchy-Green tensors are defined

$$\hat{\mathbf{C}}_\mathrm{e} = \hat{\mathbf{F}}_\mathrm{e}^T\hat{\mathbf{F}}_\mathrm{e}, \qquad \mathbf{C}_\mathrm{v} = \mathbf{F}_\mathrm{v}^T\mathbf{F}_\mathrm{v}, \tag{2.80}$$

for later use.

The strain measure (2.72) is purely elastic and the strain tensor (2.73) designates the strain rate (2.77), which is purely inelastic implying the motivation to make use of the quantities in the elasticity relation and the flow rule.

The elastic strain defines the stress state in the intermediate configuration

$$\hat{\mathbf{T}} = (\Lambda \ln J_e - \mu)\hat{\mathbf{C}}_\mathrm{e}^{-1} + \mu\mathbf{I}, \tag{2.81}$$

where a Simo&Pister type model is applied, see (Simo and Pister, 1984). $J_e = \det\hat{\mathbf{F}}_\mathrm{e} = (\det\hat{\mathbf{C}}_\mathrm{e})^{1/2}$ defines the elastic part of the volumetric deformation. The stress tensor relative to the inelastic intermediate configuration $\hat{\mathbf{T}}$ is related to the 2^{nd} Piola-Kirchhoff tensor $\tilde{\mathbf{T}}$ and the weighted Cauchy stress tensor $\mathbf{S}$ (Kirchhoff stress tensor) by the push-forward and pull-back operations

$$\hat{\mathbf{T}} = \mathbf{F}_\mathrm{v}\tilde{\mathbf{T}}\mathbf{F}_\mathrm{v}^T = \hat{\mathbf{F}}_\mathrm{e}^{-1}\mathbf{S}\hat{\mathbf{F}}_\mathrm{e}^{-T} \tag{2.82}$$

respectively, with $\mathbf{S} = (\det\mathbf{F})\mathbf{T}$, where $\mathbf{T}$ is the Cauchy stress tensor. In view of a thermo-mechanical consistent constitutive model, the Mandel stress tensor $\hat{\mathbf{P}} = \hat{\mathbf{C}}_\mathrm{e}\hat{\mathbf{T}}$ is introduced controlling the evolution of the inelastic flow

$$\overset{\triangle}{\hat{\mathbf{\Gamma}}}_\mathrm{v} = \lambda\frac{\partial F}{\partial\hat{\mathbf{P}}} = \lambda\left(\frac{\partial F}{\partial \mathrm{I}_1}\mathbf{I} + \frac{\partial F}{\partial \mathrm{J}_2}\hat{\mathbf{P}}^D\right), \tag{2.83}$$

for more details see (Bier and Hartmann, 2006). F is the yield surface defined in (2.87), $\mathrm{I}_1 = \operatorname{tr}\hat{\mathbf{P}}$ and $\mathrm{J}_2 = 1/2\,\hat{\mathbf{P}}^D \cdot \hat{\mathbf{P}}^D$ define the invariants of the Mandel stress tensor. $\mathbf{A}^D$ denotes the deviator operator $\mathbf{A} - 1/3(\operatorname{tr}\mathbf{A})\mathbf{I}$.

$$\lambda = \frac{1}{\eta}\left\langle \frac{F}{\sigma_0} \right\rangle^{r_v} \tag{2.84}$$

is a constitutive function expressing the case distinction into elastic and inelastic deformations, where $\langle x \rangle = (x + |x|)/2$ defines the Macauley-brackets, i.e. $\langle x \rangle = 0$ for $x \leq 0$ and $\langle x \rangle = x$ for $x > 0$.

The main ingredients of the model are the yield function and its evolution during the compaction process. There are a number of yield function based plasticity models in the literature (see (Bier and Hartmann, 2006) and the references therein). However, the application of Drucker-Prager type models with caps to reproduce the hardening behavior in the compressive range imply non-differentiable yield surfaces and, accordingly, require particular numerical techniques to solve the problem arising at the edges. The application of single surface models like those of (Aubertin and Li, 2004; Bigoni and Piccolroaz, 2004; Ehlers, 1995) pose the difficulty to guarantee the convexity during the evolution or to properly account for the non-uniqueness of additionally occurring regions in the stress-space. To this end the new convex single surface yield function concept discussed in (Bier and Hartmann, 2006) is applied. This yield surface is generated by interpolating an ellipsoidal function

$$g_1(\mathrm{I}_1, \sqrt{\mathrm{J}_2}) = \sqrt{\mathrm{J}_2 + \alpha\,(\mathrm{I}_1 - 3\xi)^2} - k \tag{2.85}$$

and an exponential function

$$g_2(\mathrm{I}_1, \sqrt{\mathrm{J}_2}) = \sqrt{\mathrm{J}_2 + \delta} - k + A_1 e^{A_2 I_1} \tag{2.86}$$

using an interpolation formula of (Kreisselmeier and Steinhauser, 1979) and (Arnold and Frischmuth, 1998),

$$F(\mathrm{I}_1, \sqrt{\mathrm{J}_2}) = ck \ln\left(\frac{e^{g_1(\mathrm{I}_1, \sqrt{\mathrm{J}_2})/(ck)} + e^{g_2(\mathrm{I}_1, \sqrt{\mathrm{J}_2})/(ck)}}{2} \right). \tag{2.87}$$

The abbreviations

$$k = \sqrt{\alpha(I_0 - 3\xi)^2}, \quad A_1 = \frac{k}{\left(1 - \sqrt{1 - r^2}\right)^{I_0/((3\xi - I_0)(1+r))}}, \quad A_2 = \ln(k/A_1)/I_0 \tag{2.88}$$

are introduced. For a schematic representation of F we refer to Fig. 2.4. The parameter c controls the closeness of the resulting function to the lower one of both functions so that the resulting function has the tear-drop like form.

The internal variable ξ controls the hardening in the direction of the hydrostatic stress state and α influences the form of the drop-like yield function containing the parameters I_0, r and c which are defined in advance. The small quantity δ has been introduced to avoid a vertex singularity at the intersection of the yield function with the hydrostatic axis close to I_0 following a proposal of (Abbo and Sloan, 1995). I_0 is a fixed right bound in our approach.

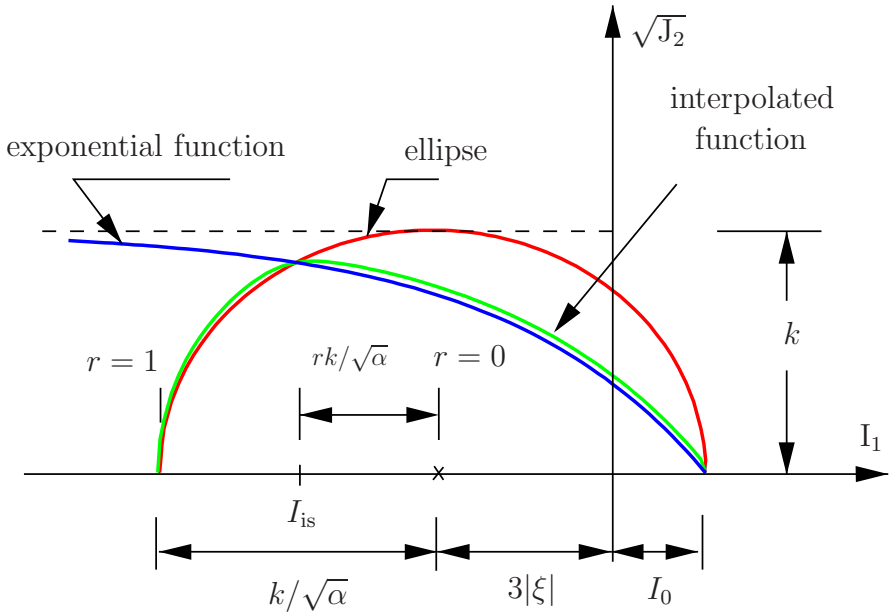


Figure 2.4: Yield function concept of interpolating simple convex models proposed by Bier and Hartmann (2006)

The volumetric inelastic deformation $\det \mathbf{F}_v$, or $\det \mathbf{C}_v$, is related to the center of the yield function defined by

$$\xi(r_K) = a_1 r_K + a_2 r_K^3 - \frac{a_3}{r_K - r_{K0}} - \frac{a_3}{r_{K0}}, \tag{2.89}$$

see proposal in Szanto et al. (2007), depending on the strain-like internal variable

$$r_K = \hat{r}_K(\mathbf{C}_v) = \ln(\det \mathbf{C}_v)/2 \quad \longrightarrow \quad \dot{r}_K = \operatorname{tr} \overset{\triangle}{\hat{\mathbf{\Gamma}}}_v = \operatorname{tr} \mathbf{L}_v. \tag{2.90}$$

a_1, a_2, a_3 and r_{K0} are material parameters. ξ defines the center of the ellipsoid and is obviously related to plastic volumetric deformation and, therefore, to the relative density of the material.

The internal variable α describes the "deviatoric behavior" and evolves according to

$$\dot{\alpha} = \lambda \left(\frac{c_D}{\alpha} (\hat{\mathbf{P}} - \xi \mathbf{I}) \cdot \frac{\partial F}{\partial \hat{\mathbf{P}}} - \alpha b_D \chi \right), \tag{2.91}$$

with the abbreviation

$$\chi = \hat{\chi}(\mathrm{I}_1, \mathrm{J}_2, \xi, \alpha) = \sqrt{3 \left(\frac{\partial F}{\partial \mathrm{I}_1} \right)^2 + 2\mathrm{J}_2 \left(\frac{\partial F}{\partial \mathrm{J}_2} \right)^2}. \tag{2.92}$$

b_D and c_D are material parameters. The process of identifying the material parameters is presented in (Bier et al., 2007) and the resulted values for a copper powder are recapped in Tab. 2.3.

Because the applied numerical scheme is based on the Total-Lagrangean formulation, the quantities relative to the reference configuration are required. In Tab. 2.4

Table 2.3: Material parameters (initial condition: $\alpha(0) = 0.558$)

identified material parameters								specified parameters					
Λ	μ	a_1	a_2	a_3	r_{K0}	b_D	c_D	r_v	r	c	I_0	δ	σ_0
GPa		MPa			MPa^{-1}		[/]			MPa^2		MPa	
5.2	8.3	24.8	23.6	15.9	-0.87	2.68	10^{-4}	1	0.3	0.01	1	0.1	1

Table 2.4: Constitutive model expressed with quantities relative to the reference configuration

	elasticity	*viscoplasticity*
loading condition	$F \leq 0$	$F > 0$
elasticity relation	$\tilde{\mathbf{T}} = (\Lambda \ln J_e - \mu)\mathbf{C}^{-1} + \mu \mathbf{C}_\mathrm{v}^{-1}$	
flow rule	$\dot{\mathbf{C}}_\mathrm{v} = \mathbf{0}$	$\dot{\mathbf{C}}_\mathrm{v} = 2\lambda \left(\frac{\partial F}{\partial \mathrm{I}_1}\mathbf{I} + \frac{\partial F}{\partial \mathrm{J}_2}(\mathbf{C}\tilde{\mathbf{T}} - (\mathrm{I}_1/3)\mathbf{I}) \right) \mathbf{C}_\mathrm{v}$
distortional hardening	$\dot{\alpha} = 0$	$\dot{\alpha} = \lambda \left(\frac{c_D}{\alpha} \left((\mathrm{I}_1 - 3\xi)\frac{\partial F}{\partial \mathrm{I}_1} + \sqrt{\mathrm{J}_2}\frac{\partial F}{\partial \sqrt{\mathrm{J}_2}} \right) - b_D \alpha \chi \right)$
abbrev.	$\mathrm{I}_1 = \mathrm{tr}\,(\tilde{\mathbf{T}}\mathbf{C})$, $\mathrm{J}_2 = (\mathbf{C}\tilde{\mathbf{T}} \cdot \tilde{\mathbf{T}}\mathbf{C} - \mathrm{I}_1^2/3)/2$, $J_e = ((\det \mathbf{C})/(\det \mathbf{C}_\mathrm{v}))^{1/2}$ $r_K = \ln(\det \mathbf{C}_\mathrm{v})/2$, $\xi = a_1 r_K + a_2 r_K^3 - \frac{a_3}{r_K - r_{K0}}$, $\lambda = \frac{1}{\eta}(F/\sigma_0)^{r_v}$	

the model is compiled indicating that in three dimensions 7 evolution equations have to be integrated, which are subject to a case distinction called loading condition (6 equations due to $\mathbf{C}_\mathrm{v}$ and one for α).

Chapter 3

Variational formulation

3.1 Variational formulation

An analytic solution of the system of equations characterized in Fig. 2.2 is possible only for simple problems. The range of application is dramatically enlarged if approximate solutions are employed. Among the numerical methods to solve partial differential equations are the finite difference method, the boundary element method and many variants of the finite element method. In Sec. 4.2 we introduce the p-version of the finite element method which possesses some very advantageous properties.

A straightforward way to derive a variational form yields the principle of virtual displacements, one-field variational principle as only the displacements are varied. An alternative approach results in multi-field variational principles (see, for an overview (Belytschko et al., 2000, Sec. 8.5), (Holzap fel, 2000, Sec. 8.5 – 8.6)) that are discretized as mixed finite elements to overcome shortcomings of traditional h-version elements like locking. The Hellinger-Reissner principle (see (Hellinger, 1914) and (Reissner, 1950)) is a two-field principle with unknown displacements and stresses. The most general form is the Hu-Washizu principle where also the strains are principal variables. Another way to overcome locking is the use of higher order elements. As the p-version was shown to perform well also for locking-dominated nonlinear problems, see Sec. 5.3, a purely displacement-based approach is not only simpler but also justified.

To arrive at the variational or weak form applying the principle of virtual displacements, we multiply the partial differential momentum equation (2.42) with a test function $\delta\vec{u}$ followed by partial integration. The first step yields

$$\int_V \mathrm{Div}\,\mathbf{T}_\mathrm{R}(\vec{X},t)\cdot\delta\vec{u}(\vec{X})\;\mathrm{d}V + \int_V \rho_\mathrm{R}(\vec{X})\vec{k}\cdot\delta\vec{u}(\vec{X})\;\mathrm{d}V = 0 \quad . \tag{3.1}$$

Without loss of generality we focus on three-dimensional continua. Then, in all three coordinate directions $d, d = 1,2,3$, two types of boundary conditions (b.c.) can be distinguished: Dirichlet (essential or geometric) b.c. and Neumann (force or natural) boundary conditions. The surface of the domain $A = \partial V$ is decomposed in non overlapping parts, A_{u_d} where the geometric boundary conditions $u_d(\vec{x},t) = \vec{u}(\vec{x},t)\cdot\vec{e}_d = \overline{u}_d(\vec{x},t)$, $\vec{x}\in A_{u_d}$ are given and a part A_{t_d} where the surface traction $t_d(\vec{x},t) = \vec{t}_\mathrm{R}(\vec{x},t)\cdot\vec{e}_d = \mathbf{T}_\mathrm{R}\vec{n}_\mathrm{R}\cdot\vec{e}_d = \overline{t}_d(\vec{x},t)$, $\vec{x}\in A_{t_d}$. The unit vectors in direction d are $\vec{e}_d$ and the normal vector on the surface in the undeformed

configuration is denoted by $\vec{n}_{\mathrm{R}}$. Formally this decomposition of the surface can be written as

$$A = A_{u_d} \cup A_{t_d}, \quad A_{u_d} \cap A_{t_d} = \emptyset, \quad d = 1, 2, 3\,. \tag{3.2}$$

The vector-valued test function that is identified as virtual displacement $\delta\vec{u}(\vec{X}, t)$ has to vanish on those parts of the surface of the body where displacement boundary conditions are prescribed. The virtual displacement is in principle independent of time t, however as the Dirichlet boundary conditions may change in time the conditions on the function $\delta\vec{u}(\vec{X}, t)$ can change: surface areas where no Dirichlet b.c. was set at time t_1 and consequently no restrictions for $\delta\vec{u}$ existed may be clamped at time t_2 with the consequence that in those regions $\delta\vec{u}(\vec{X}, t_2)$ must vanish. As we do not treat contact in this thesis, most processes can be split in sequences with constant Dirichlet b.c. regions for the partial process. Therefore, the dependency of $\delta\vec{u}$ on the time is neglected.

Integration by parts of the first term followed by the application of the divergence theorem and incorporating the condition on the surface traction $\vec{t}_{\mathrm{R}}$ we obtain the weak form in the reference configuration

$$\begin{aligned} \pi(t, \vec{u}, \delta\vec{u}, \mathbf{q}) \;&=\; \int_V \mathbf{T}_{\mathrm{R}}(\vec{X}, t) \cdot \mathrm{Grad}\, \delta\vec{u}(\vec{X})\; \mathrm{d}V \;- \\ &-\; \int_V \rho_{\mathrm{R}}(\vec{X})\vec{k} \cdot \delta\vec{u}(\vec{X})\; \mathrm{d}V - \int_A \vec{t}_{\mathrm{R}}(\vec{X}, t) \cdot \delta\vec{u}(\vec{X})\; \mathrm{d}A = 0 \end{aligned} \tag{3.3}$$

where $\vec{t}_{\mathrm{R}} = \mathbf{T}_{\mathrm{R}}\vec{n}_{\mathrm{R}}$ defines the traction vector operating on the undeformed surface A and $\vec{n}_{\mathrm{R}}$ designates the outward normal vector on the surface of the reference configuration. The internal variables $\mathbf{q}$ were introduced in (2.43).

The unsymmetrical first Piola-Kirchhoff stress tensor $\mathbf{T}_{\mathrm{R}}$ in Eq. (3.3) can be replaced by $\mathbf{T}_{\mathrm{R}} = \mathbf{F}\tilde{\mathbf{T}}$ and we obtain

$$\mathbf{T}_{\mathrm{R}}(\vec{X}, t) \cdot \mathrm{Grad}\, \delta\vec{u}(\vec{X}) = \tilde{\mathbf{T}} \cdot \mathbf{F}^T \mathrm{Grad}\, \delta\vec{u} = \tilde{\mathbf{T}} \cdot \frac{1}{2}(\mathbf{F}^T \mathrm{Grad}\, \delta\vec{u} + \mathrm{Grad}^T \delta\vec{u}\, \mathbf{F}) = \tilde{\mathbf{T}} \cdot \delta\mathbf{E}. \tag{3.4}$$

In Eq. $(3.4)_1$ we used the fact that for the double contraction of tensors the following relation is valid (Holzap fel, 2000, Eq. $(1.95)_1$)

$$(\mathbf{B}^T\mathbf{A}) \cdot \mathbf{C} = \mathbf{A} \cdot (\mathbf{B}\mathbf{C}) \quad \text{here:} \quad (\mathbf{F}\tilde{\mathbf{T}}) \cdot \mathrm{Grad}\, \delta\vec{u} = \tilde{\mathbf{T}} \cdot (\mathbf{F}^T \mathrm{Grad}\, \delta\vec{u}). \tag{3.5}$$

For Eq. $(3.4)_2$ the property is exploited that for the product of a symmetric tensor ($\tilde{\mathbf{T}}$ represented by $\mathbf{S}$) with an arbitrary tensor (here $\mathbf{B}$) only the symmetric part of the tensor $\mathbf{B}$ remains (Holzap fel, 2000, Eq. (1.115)),

$$\mathbf{S} \cdot \mathbf{B} = \mathbf{S} \cdot \frac{1}{2}(\mathbf{B} + \mathbf{B}^T). \tag{3.6}$$

Furthermore, for Eq. $(3.4)_3$ we used the relation that the virtual Green strain $\delta\mathbf{E}$, the directional derivative[1] (defined in Eq. (3.21)) of the Green strain $\mathbf{E}$ in direction of the virtual

[1] The chain rule for Gâteaux derivatives is for example given in (Gurtin, 2003, p. 26).

displacement $\delta\vec{u}$, is the symmetric part[2] of the tensor $\mathbf{F}^T\mathrm{Grad}\,\delta\vec{u}$ (Hartmann, 2003a, p. 119)

$$\begin{aligned}\delta\mathbf{E}(\vec{u},\delta\vec{u}) &= \mathsf{D}_{\vec{u}}\mathbf{E}(\vec{u})[\delta\vec{u}] = \mathsf{D}_{\mathbf{G}}\mathbf{E}(\mathbf{G})\big[\mathsf{D}_{\vec{u}}\mathbf{G}(\vec{u})[\delta\vec{u}]\big] = \frac{1}{2}(\mathbf{F}^T\delta\mathbf{G}+\delta\mathbf{G}^T\mathbf{F}) \\ &= \mathrm{sym}(\mathbf{F}^T\delta\mathbf{G}) = \frac{1}{2}\Big(\mathbf{F}^T\mathrm{Grad}\,\delta\vec{u}(\vec{X}) + \mathrm{Grad}^T\delta\vec{u}(\vec{X})\mathbf{F}\Big)\end{aligned} \tag{3.7}$$

with the displacement and virtual displacement gradients

$$\mathbf{G} = \mathrm{Grad}\,\vec{u}(\vec{X},t) \quad \text{and} \quad \delta\mathbf{G} = \mathrm{Grad}\,\delta\vec{u}(\vec{X}). \tag{3.8}$$

Finally, substituting Eq. (3.4) in Eq. (3.3) the principle of virtual work can be split in internal virtual work π_{int} and an external virtual work π_{ext} contribution as

$$\pi(t,\vec{u},\delta\vec{u},\mathbf{q}) = \pi_{\mathrm{int}}(t,\vec{u},\delta\vec{u},\mathbf{q}) - \pi_{\mathrm{ext}}(t,\vec{u},\delta\vec{u}) = 0. \tag{3.9}$$

The internal virtual work is identified as

$$\pi_{\mathrm{int}}(t,\vec{u},\delta\vec{u},\mathbf{q}) = \int_V \delta\mathbf{E}(\vec{X},t)\cdot\tilde{\mathbf{T}}(\mathbf{C},\mathbf{q})\;\mathrm{d}V \tag{3.10}$$

where the second Piola-Kirchhoff tensor $\tilde{\mathbf{T}}$ depends on the right Cauchy-Green tensor $\mathbf{C}(\vec{X},t)$ and the internal variables $\mathbf{q}(\vec{X},t)$ as stipulated in Eq. (2.43).

If we also take deformation dependent surface loads (follower loads) into account, the external work consists of the following parts

$$\begin{aligned}\pi_{\mathrm{ext}}(t,\vec{u},\delta\vec{u}) &= \int_V \delta\vec{u}(\vec{X})\cdot\rho_{\mathrm{R}}(\vec{X})\vec{k}\;\mathrm{d}V + \\ &+ \int_A \delta\vec{u}(\vec{X})\cdot\vec{t}_{\mathrm{R}}(\vec{X},t)\;\mathrm{d}A + \int_a \delta\vec{u}(\vec{x})\cdot\vec{t}(\vec{x},t)\;\mathrm{d}a,\end{aligned} \tag{3.11}$$

where the last term accounts for a traction contribution on the *deformed* surface a. Only this last term depends on the deformation $\vec{u}$. An example is hydrostatic pressure with an amplitude p that acts on the deformed surface described by the traction vector

$$\vec{t}(\vec{x},t) = \mathbf{T}\vec{n} = -p\mathbf{I}\vec{n} = -p\vec{n}(\vec{x},t)\,, \tag{3.12}$$

see Sec. 3.3.2.

The principle of virtual work (3.9) is both physically and geometrically nonlinear. For a numerical treatment it has to be linearized. This is the topic addressed in the following section.

Remark 4 *The principle of virtual work as derived above applies to both conservative and non-conservative systems where no energy functional exists. It is quite general as no particular material was taken into account, we just assume to have the stresses* $\tilde{\mathbf{T}}$ *'by some constitutive model'. If we restrict the range of application to conservative hyperelastic systems, we require the existence of an energy functional* Π *for both the stresses and the loads. Then, for a static*

[2] An alternative derivation is obtained from Eq. (3.33) if Δ is exchanged by δ.

problem the total potential energy of a system is composed of the internal Π_{int} *and external* Π_{ext} *potential energy, cf. (Holzap fel, 2000, Sec. 8.3), (Bonet and Wood, 1997, Sec. 6.6.1),*

$$\Pi(\vec{u}) = \Pi(\vec{u})_{int} - \Pi(\vec{u})_{ext} \tag{3.13}$$

with

$$\Pi(\vec{u})_{int} = \int_V \Psi(\vec{u}) \, \mathrm{d}V, \quad \Pi(\vec{u})_{ext} = \int_V \rho_R \vec{k} \cdot \vec{u} \, \mathrm{d}V + \int_A \vec{t}_R \cdot \vec{u} \, \mathrm{d}A \tag{3.14}$$

where $\Psi(\vec{u})$ *denotes the hyperelastic strain energy function and* $\Pi(\vec{u})_{ext}$ *the potential energy of dead load. The deformed shape, i.e. the equilibrium state for a given load, is that configuration where the potential energy reaches its minimum. This is achieved if the total potential energy is rendered stationary,*

$$\delta\Pi(\vec{u}, \delta\vec{u}) = \mathsf{D}_{\vec{u}}\Pi(\vec{u})[\delta\vec{u}] = 0, \tag{3.15}$$

i.e. the directional derivative has to vanish in all directions $\delta\vec{u}$*, compare Eq. (3.26) for the definition of variation as directional derivative. As before, the arbitrary vector field* $\delta\vec{u}$ *has to vanish where displacement boundary conditions are prescribed.*

3.2 Linearization

Nonlinearities in continuum mechanics can stem from different origins. There is geometrical nonlinearity as a consequence of a nonlinear strain measure like the Green-Lagrange strain. Physical nonlinearities result in nonlinear constitutive models and contact problems described as inequality relations also introduce nonlinearities.

Linearization enables to formulate approximate theories that make analytical solutions for more or less simple problems possible, for example, the classical linear beam theory. If the applications are more complex, approximate *numerical* solutions of the nonlinear problem are pursued. One popular example are Newton-type algorithms to solve nonlinear equations arising in the finite element method. In the finite element world the need for *consistent linearization* of all the quantities associated with a nonlinear problem was introduced by (Hughes and Pister, 1978) and (Simo and Taylor, 1985). Subsequently, it was shown that linearization is a systematic process based on the directional derivative (Gâteaux-derivative) and formally equivalent to the variation of fields.

This section summarizes the idea of linearization and performs the linearization of the weak form. It is based on (Bonet and Wood, 1997, Sec. 2.3), (Belytschko et al., 2000, Sec. 6.4), (Holzap fel, 2000, Sec. 8.4), (Marsden and Hughes, 1993, Ch. 4), and (Wriggers, 2001, Sec. 3.5).

The concept of linearization

We start from a general set of nonlinear equations

$$\mathcal{F}(\boldsymbol{x}) = \boldsymbol{0} \tag{3.16}$$

where $\mathcal{F}$ may be a tensor, vector or scalar valued function with a tensor, vector or scalar valued argument $\boldsymbol{x}$. Consider an initial guess $\boldsymbol{x}_0$ and an increment $\boldsymbol{u}$ that (we hope) will generate

$\boldsymbol{x} = \boldsymbol{x}_0 + \boldsymbol{u}$ closer to the solution of Eq. (3.16). As in general it is not immediately obvious, how to express the derivative of a complicated function $\mathcal{F}$ with respect to $\boldsymbol{x}$, an artificial scalar parameter ϵ is introduced allowing us to define a nonlinear function $\mathbf{F}$ in ϵ as

$$\mathbf{F}(\epsilon) = \mathcal{F}(\boldsymbol{x}_0 + \epsilon \boldsymbol{u}). \tag{3.17}$$

Sketched for a two-degree-of-freedom system $f(x_1, x_2)$ this re-parametrization defines a curve $\mathbf{F}(\epsilon)$ in the scalar parameter ϵ as shown in Fig. 3.1. A Taylor's series expansion of the nonlinear

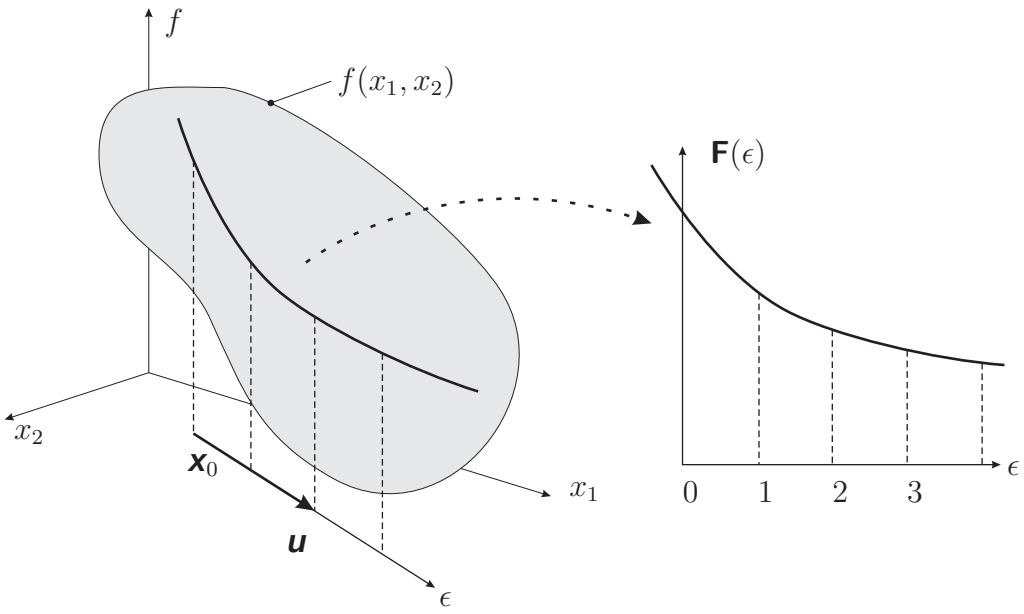


Figure 3.1: Two-degree-of-freedom case $f(x_1, x_2) = 0$ and re-parametrization $\mathbf{F}(\epsilon) = 0$

function $\mathbf{F}(\epsilon)$ at $\epsilon = 0$, corresponding to $\boldsymbol{x} = \boldsymbol{x}_0$, gives

$$\mathbf{F}(\epsilon) = \mathbf{F}(0) + \left.\frac{\mathrm{d}\mathbf{F}(\epsilon)}{\mathrm{d}\epsilon}\right|_{\epsilon=0} \epsilon + \frac{1}{2} \left.\frac{\mathrm{d}^2\mathbf{F}(\epsilon)}{\mathrm{d}\epsilon^2}\right|_{\epsilon=0} \epsilon^2 + \dots \quad . \tag{3.18}$$

Using the definition of $\mathbf{F}$ given in Eq. (3.17) we obtain

$$\mathcal{F}(\boldsymbol{x}_0 + \epsilon \boldsymbol{u}) = \mathcal{F}(\boldsymbol{x}_0) + \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\mathcal{F}(\boldsymbol{x}_0 + \epsilon \boldsymbol{u})\right|_{\epsilon=0} \epsilon + \frac{1}{2} \left.\frac{\mathrm{d}^2}{\mathrm{d}\epsilon^2}\mathcal{F}(\boldsymbol{x}_0 + \epsilon \boldsymbol{u})\right|_{\epsilon=0} \epsilon^2 + \dots \quad . \tag{3.19}$$

Truncating the series after the linear term and setting $\epsilon = 1$ gives a linear approximation to the change in the nonlinear function $\mathcal{F}(\boldsymbol{x})$ as

$$\mathcal{F}(\boldsymbol{x}_0 + \boldsymbol{u}) - \mathcal{F}(\boldsymbol{x}_0) \approx \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\mathcal{F}(\boldsymbol{x}_0 + \epsilon \boldsymbol{u})\right|_{\epsilon=0} \quad . \tag{3.20}$$

The right-hand side is identified as the directional derivative (Gâteaux derivative) of $\mathcal{F}(\boldsymbol{x})$ at $\boldsymbol{x}_0$ in the direction $\boldsymbol{u}$ and defined as

$$\mathsf{D}_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{x}_0)[\boldsymbol{u}] = \left.\frac{\mathrm{d}}{\mathrm{d}\epsilon}\mathcal{F}(\boldsymbol{x}_0 + \epsilon \boldsymbol{u})\right|_{\epsilon=0} \quad . \tag{3.21}$$

The Gâteaux derivative $\mathsf{D}_{\boldsymbol{u}} f(\boldsymbol{x}_0)[\Delta \boldsymbol{u}]$ is a linear operator and the usual properties of the derivative, like product rule and chain rule are satisfied, see e.g. (Bonet and Wood, 1997, Sec. 2.3.3). As this defining equation is generally valid for scalars, vectors, matrices and tensors the kind of multiplication (scalar product, matrix product, tensor contraction, etc.) is not explicitly written but indicated by putting the second term in square brackets, viz $[\boldsymbol{u}]$.

With the help of the directional derivative the linearized representation of the function $\mathcal{F}(\boldsymbol{x})$ at $\boldsymbol{x}_0$ in the direction $\boldsymbol{u}$ is defined as

$$\mathsf{L}[\mathcal{F}(\boldsymbol{x})]_{\boldsymbol{x}=\boldsymbol{x}_0} = \mathcal{F}(\boldsymbol{x}_0) + \mathsf{D}_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{x}_0)[\boldsymbol{u}] \quad . \tag{3.22}$$

Newton-Raphson procedure. For the solution of the nonlinear equation Eq. (3.16) the use of the linear approximation given in Eq. (3.22) allows to establish the well known Newton-Raphson algorithm, compare also (Parisch, 2003, Sec. 8). Evaluating the linearized approximation at an initial guess $\boldsymbol{x}_0$,

$$\mathcal{F}(\boldsymbol{x}_0) + \mathsf{D}_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{x}_0)[\boldsymbol{u}] = \boldsymbol{0}, \tag{3.23}$$

we solve for the increment $\boldsymbol{u}$. Introducing the iteration index $m = 0, 1, \ldots, n$, the procedure to obtain an updated solution is defined by

$$\mathsf{D}_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{x}_m)[\boldsymbol{u}] = -\mathcal{F}(\boldsymbol{x}_m) \qquad \rightsquigarrow \quad \boldsymbol{u} \tag{3.24}$$

$$\boldsymbol{x}_{m+1} = \boldsymbol{x}_m + \boldsymbol{u} \qquad \rightsquigarrow \quad \boldsymbol{x}_{m+1} \to \boldsymbol{x}_m. \tag{3.25}$$

A simple stopping criterion for the iterative process could be to check whether the norm of the residual $\mathcal{F}(\boldsymbol{x}_m)$ or of the increment $\boldsymbol{u}$ is below a prescribed threshold. The Newton procedure has the advantage of quadratic convergence in the vicinity of the solution. If the initial guess $\boldsymbol{x}_0$ however is too far away from the solution, the procedure may diverge. Further remarks on two variants of the Newton method are given in Sec. 4.7.

Variation. As already mentioned the directional derivative is the basis for linearization, as well as for the variation of a field quantity. The first variation, or Gâteaux variation, $\delta\mathbf{A}$ of a field $\mathbf{A}(\boldsymbol{x})$ is defined as the directional derivative (3.21) of $\mathbf{A}(\boldsymbol{x})$ in the direction of $\delta\boldsymbol{u}$:

$$\delta\mathbf{A}(\boldsymbol{x}, \delta\boldsymbol{u}) = \mathsf{D}_{\boldsymbol{x}}\mathbf{A}(\boldsymbol{x})[\delta\boldsymbol{u}] = \frac{\mathrm{d}}{\mathrm{d}\epsilon}\mathbf{A}(\boldsymbol{x} + \epsilon\delta\boldsymbol{u})\bigg|_{\epsilon=0} . \tag{3.26}$$

For details we refer to (Meyberg and Vachenauer, 2006, Sec. 13) and (Holzap fel, 2000, Sec. 8.1).

3.3 Linearization of the variational equation

If the nonlinear principle of virtual displacements (3.9) is treated by the Newton-Raphson algorithm, it has to be linearized. Here, we derive the linearization with respect to the material configuration starting from a deformation state $(\vec{u}, \mathbf{q})$.

The linearized representation of the weak form is obtained by applying the definition of Eq. (3.22) to the principle of virtual displacements Eq. (3.9),

$$\mathsf{L}[\pi]_{(\vec{u},\mathbf{q})} \quad = \pi(t, \vec{u}, \delta\vec{u}, \mathbf{q}) + \mathsf{D}_{\vec{u}}\,\pi(t, \vec{u}, \delta\vec{u}, \mathbf{q})[\Delta\vec{u}] = \pi + \mathsf{D}_{\vec{u}}\,\pi[\Delta\vec{u}] = 0\,. \tag{3.27}$$

The first term π is obtained by evaluating Eq. (3.9) at the known state $(\vec{u}, \mathbf{q})$.

In the following we consider the Gâteaux derivative of $\mathsf{D}_{\vec{u}}\,\pi$ in the direction $\Delta\vec{u}$.

3.3.1 Directional derivative of internal virtual work

The Gâteaux derivative of the internal virtual work π_{int}, Eq. (3.10), is evaluated using the product rule

$$\begin{aligned} \mathsf{D}_{\vec{u}}\, \pi_{\text{int}}(t, \vec{u}, \delta\vec{u}, \mathbf{q})[\Delta\vec{u}] &= \mathsf{D}_{\vec{u}} \left(\int_V \delta\mathbf{E} \cdot \tilde{\mathbf{T}}\ \mathrm{d}V \right)[\Delta\vec{u}] \\ &= \int_V \left(\mathsf{D}_{\vec{u}}\delta\mathbf{E}[\Delta\vec{u}] \cdot \tilde{\mathbf{T}} + \delta\mathbf{E} \cdot \mathsf{D}_{\vec{u}}\tilde{\mathbf{T}}[\Delta\vec{u}] \right)\ \mathrm{d}V. \end{aligned} \tag{3.28}$$

First we treat $\mathsf{D}_{\vec{u}}\delta\mathbf{E}[\Delta\vec{u}]$: starting from the representation of $\delta\mathbf{E}$ given in Eq. (3.7) we can evaluate

$$\mathsf{D}_{\vec{u}}\delta\mathbf{E}(\vec{u})[\Delta\vec{u}] = \operatorname{sym}(\mathsf{D}_{\vec{u}}\mathbf{F}^T[\Delta\vec{u}]\delta\mathbf{G}) = \operatorname{sym}(\operatorname{Grad}^T \Delta\vec{u}\ \operatorname{Grad}\delta\vec{u}), \tag{3.29}$$

where we used

$$\mathsf{D}_{\vec{u}}\mathbf{F}(\vec{u})[\Delta\vec{u}] = \frac{\mathrm{d}}{\mathrm{d}\epsilon}(\mathbf{1} + \operatorname{Grad}(\vec{u} + \epsilon\Delta\vec{u}))\Big|_{\epsilon=0} = \operatorname{Grad}\Delta\vec{u}. \tag{3.30}$$

For the directional derivative of the stresses $\mathsf{D}_{\vec{u}}\tilde{\mathbf{T}}[\Delta\vec{u}]$ as needed in (3.28) we assume that the internal variables $\mathbf{q}$ are implicit functions of the strains in the vicinity of an equilibrium state $(\vec{u}, \mathbf{q})$, i.e. we apply the implicit function theorem, cf. Sec. 4.7, to use the multi-level Newton algorithm. Then, the derivative of the stresses

$$\tilde{\mathbf{T}}(\vec{X}, t) = \tilde{\boldsymbol{\Phi}}\big(\mathbf{C}(\mathbf{E}), \mathbf{q}(\mathbf{C}(\mathbf{E}))\big) \tag{3.31}$$

can be evaluated applying the chain rule as

$$\mathsf{D}_{\mathbf{E}}\tilde{\boldsymbol{\Phi}}(\mathbf{E})[\Delta\mathbf{E}] = \left[\frac{\partial\tilde{\boldsymbol{\Phi}}}{\partial\mathbf{C}} + \frac{\partial\tilde{\boldsymbol{\Phi}}}{\partial\mathbf{q}} \cdot \frac{\mathrm{d}\mathbf{q}}{\mathrm{d}\mathbf{C}} \right] 2\,\Delta\mathbf{E} = \tilde{\mathcal{C}}\ \Delta\mathbf{E} \tag{3.32}$$

with the abbreviation

$$\begin{aligned} \Delta\mathbf{E} = \mathsf{D}_{\vec{u}}\mathbf{E}(\vec{u})[\Delta\vec{u}] &= \frac{1}{2}\left(\mathsf{D}_{\vec{u}}\mathbf{F}^T[\Delta\vec{u}]\,\mathbf{F} + \mathbf{F}^T\,\mathsf{D}_{\vec{u}}\mathbf{F}[\Delta\vec{u}] - \mathsf{D}_{\vec{u}}\mathbf{1}\,[\Delta\vec{u}] \right) \\ &= \frac{1}{2}\left(\operatorname{Grad}^T\Delta\vec{u}\,\mathbf{F} + \mathbf{F}^T\operatorname{Grad}\Delta\vec{u} \right) \\ &= \operatorname{sym}(\mathbf{F}^T\operatorname{Grad}\Delta\vec{u}) \end{aligned} \tag{3.33}$$

and the forth-order tensor $\tilde{\mathcal{C}}$ (tangent-operator) describing the linearized constitutive relationship at the state $(\vec{u}, \mathbf{q})$

$$\tilde{\mathcal{C}}(\vec{u}, \mathbf{q}) = 2\left[\frac{\partial\tilde{\boldsymbol{\Phi}}}{\partial\mathbf{C}} + \frac{\partial\tilde{\boldsymbol{\Phi}}}{\partial\mathbf{q}} \cdot \frac{\mathrm{d}\mathbf{q}}{\mathrm{d}\mathbf{C}} \right] \tag{3.34}$$

relating the increment in stress ('$\Delta\tilde{\mathbf{T}}$') to the increment in strain ('$\Delta\mathbf{E}$')

$$\mathsf{D}_{\vec{u}}\tilde{\mathbf{T}}[\Delta\vec{u}] = \tilde{\mathcal{C}}\ \mathsf{D}_{\vec{u}}\mathbf{E}[\Delta\vec{u}] = \tilde{\mathcal{C}}\ \operatorname{sym}(\mathbf{F}^T\operatorname{Grad}\Delta\vec{u}). \tag{3.35}$$

Remark 5 *The factor* 2 *in* $\tilde{\mathcal{C}}$ *stems from the fact that we want to express the right Cauchy-Green tensor depending on the Green strain tensor as* $\mathbf{C} = 2\mathbf{E} + \mathbf{1}$. *Hence the directional derivative* $\mathsf{D}_{\mathbf{E}}\mathbf{C}(\mathbf{E})[\Delta\mathbf{E}] = 2\Delta\mathbf{E}$ *bears the factor* 2.

The explicit calculation of the tangent-operator $\tilde{\mathcal{C}}$ depends on the assumed constitutive model and can be quite intricate as indicated in Sec. 2.3.

Substituting relations (3.29) – (3.35) in Eq. (3.28), we finally arrive at

$$\begin{aligned} \mathsf{D}_{\vec{u}}\,\pi_{\text{int}}(t,\vec{u},\delta\vec{u},\mathbf{q})[\Delta\vec{u}] &= \int_V \operatorname{sym}(\operatorname{Grad}^T\Delta\vec{u}\,\operatorname{Grad}\delta\vec{u})\cdot\tilde{\mathbf{T}}\;\mathrm{d}V + \int_V \delta\mathbf{E}\cdot\tilde{\mathcal{C}}\;\mathsf{D}_{\vec{u}}\mathbf{E}[\Delta\vec{u}]\;\mathrm{d}V \\ &= \int_V \operatorname{Grad}\Delta\vec{u}\,\tilde{\mathbf{T}}\cdot\operatorname{Grad}\delta\vec{u}\;\mathrm{d}V + \int_V \delta\mathbf{E}\cdot\tilde{\mathcal{C}}\;\mathsf{D}_{\vec{u}}\mathbf{E}[\Delta\vec{u}]\;\mathrm{d}V, \quad (3.36) \end{aligned}$$

compare for example Eq. (3.329) in (Wriggers, 2001) and (Holzap fel, 2000, p. 396). Where we used for the transformation of the first integral the symmetry of the stress tensor ($\tilde{\mathbf{T}}\cdot\operatorname{sym}(\mathbf{A}) = \tilde{\mathbf{T}}\cdot\mathbf{A}$, (Holzap fel, 2000, Eq. (1.115))) and the relations $\mathbf{A}\cdot(\mathbf{BC}) = (\mathbf{B}^T\mathbf{A})\cdot\mathbf{C} = \mathbf{C}\cdot(\mathbf{B}^T\mathbf{A})$. (Holzap fel, 2000, Eq. (1.95)), (Scholz, 2006, Eq. (2.35)). After discretizing the first integral gives rise to the 'initial stress tangent matrix' or 'geometric stiffness tangent matrix' as the stresses $\tilde{\mathbf{T}}$ are held constant and the influence of the deformation is taken into account. The second integral becomes the 'material tangent stiffness' as it involves the change of the stress and thus depends on the material response, see also (Belytschko et al., 2000, p. 338).

3.3.2 Follower load: directional derivative of external virtual work

To evaluate the Gâteaux derivative of the external virtual work π_{ext} (3.11), the contributions $(3.11)_1$, $(3.11)_2$ expressed in the reference configuration vanish as they do not depend on the displacement $\vec{u}$. The only remaining part concerns the follower load contribution. Different assumptions on the nature of the displacement dependent load $\vec{t}(\vec{x},t)$ can be made. Restricting ourself to the important case of pressure (traction always perpendicular to the current surface), we can follow (Schweizerhof and Ramm, 1984) and distinguish two contributions, resulting in different types of pressure load with consequences for the linearization

$$\vec{t} = -p(\vec{u})\,\vec{d}(\vec{u}) = \begin{cases} -p & \vec{n}(\vec{u}), \quad \text{body attached loading} \\ -p(\vec{u}) & \vec{n}(\vec{u}), \quad \text{space attached loading} \end{cases} . \quad (3.37)$$

- For *body attached loads* the magnitude is independent of the displacement, i.e. $p(\vec{u}) = p^0$, and only the load direction depends on the deformation dependent unit normal vector $\vec{d}(\vec{u}) = \vec{n}(\vec{u})$. An example would be isostatic gas or fluid pressure or the pressure for example in a cold isostatic pressing chamber.

- In the case of *space attached loads* the magnitude and distribution $p(\vec{u})$ also depends on the position (and thus on the displacement), and the direction is again given by the unit normal vector $\vec{d}(\vec{u}) = \vec{n}(\vec{u})$. Examples are hydrostatic pressure depending on the height of the fluid column above or a beam with a magnetic tip in a magnetic field.

There is a broad body of literature[3] addressing deformation dependent loading and the numerical treatment of it, e.g. (Sewell, 1967), (Hibbitt, 1979), (Argyris and Symeonidis, 1981), (Argyris et al., 1982), (Mang, 1980), (Schweizerhof, 1982; Schweizerhof and Ramm, 1984), (Mok et al., 1999), (Simo et al., 1991), (Rumpel and Schweizerhof, 2004) to name just a few. However, in the context of the p-version, follower loading was treated only recently by (Heisserer et al., 2005) and (Yosibash et al., 2007). The article (Noel and Szabó, 1997) discusses a p-version formulation in the spatial reference frame where all traction loads automatically act on the current configuration. Some textbooks that address follower loading are e.g. (Wriggers, 2001, Sec. 3.5.3, Sec. 4.2.5), (Bonet and Wood, 1997, Sec. 6.5), (Holzap fel, 2000, p. 383).

The literature reflects a controversial discussion of the question whether or not follower loading is conservative, leading to a symmetric stiffness after discretization. As every closed system is conservative, this discussion is more precisely addressing the part of the system that is modeled in the analysis. In a letter to the editor (Koiter, 1996) of the *Journal of Sound and Vibration* it was argued that in the absence of physical evidence, follower forces are purely a 'Gedankenexperiment' – the letter concluded "be aware of unrealistic follower forces". Less drastic the existence of a potential for *static fluid loads* was proved analytically by Bufler (Bufler, 1984, 1985) and it was shown that this conservativity can be destroyed by unrealistic simplifications of the load operator. Based on these investigations Schneider (Schneider, 1990) derived necessary conditions for the conservativity of deformation dependent pressure loading. One of his statements was that every static gas or fluid pressure load that can be imposed in a real experiment is conservative. In a similar manner Schweizerhof and Ramm (Schweizerhof and Ramm, 1984) used the symmetry of the discretized follower load stiffness matrix as indicator for conservative systems. Their point is that usually pressure loads are force fields in space that lead to symmetric load stiffness matrices indicating conservative loading. These *space* attached loads however are often simplified to the *body* attached case, and the dependence of load magnitude and distribution on the deformation $p(\vec{u}(\vec{X}))$ is lost, resulting in a lack of symmetry of the load stiffness matrices. There are, however, cases in reality that correspond to the *body* attached concept. What about those? Schweizerhof and Ramm (Schweizerhof and Ramm, 1984) performed an integration by part of the load stiffness matrix (for constant load magnitude p see also (Bonet and Wood, 1997, Eq. (6.21), Eq. (6.22))), yielding a domain integral '$\int$' and a boundary integral '$\oint$'. The matrix resulting from the domain integral after discretization is unsymmetric for *body attached* loads with the exception of the case when the load magnitude is constant (this special case is given in (Bonet and Wood, 1997, Eq. (6.22))). The boundary integral part always results in an unsymmetric stiffness matrix. While the possible unsymmetry in the domain integrals remains after assembly of the individual element matrices to the global stiffness matrix, the boundary terms of adjacent elements mutually cancel each other and only the outermost boundary terms remain. So if we model a static (for example linearly varying) fluid pressure as body attached load, the unsymmetric global stiffness matrix would imply nonconservative loading – a contradiction to the statement of Bufler that static fluid pressure loads are always conservative? The solution is to show that varying *body* attached pressure can not be realized physically. According to the investigations of Schweizerhof & Ramm (Schweizerhof and Ramm, 1984) and Schneider

[3]For example, the Journal of Sound and Vibration published over 100 papers since 1978 treating follower loading.

(Schneider, 1990) a varying deformation dependent *body attached* pressure can not be realized physically and can only be induced by an unsuitable simplification of the loading. The only case of body attached load that is physically realistic is constant pressure, and this leads to symmetric domain integrals as mentioned. If, furthermore, the deformation of the points along which the boundary integral is evaluated are fixed or prescribed (in two dimensions these are the endpoints of the loaded area), the boundary integral vanishes and the global stiffness matrix is symmetric. The boundary integral also vanishes if a symmetry plane bisects the pressure loaded region.

In the following we treat the case of *body attached* loads with *constant magnitude* p not depending on the deformation

$$\pi_{\text{ext}}(t, \vec{u}, \delta\vec{u}) \;=\; \int_a \delta\vec{u}(\vec{x}) \cdot \vec{t}(\vec{x}, t)\ \mathrm{d}a = -\int_a \delta\vec{u}(\vec{x}) \cdot p\vec{n}(\vec{x}, t)\ \mathrm{d}a \tag{3.38}$$

where the deformed surface element $\mathrm{d}a$ and the direction of the normal vector $\vec{n}$ depend on the deformed configuration and are subject to linearization.

To carry out the linearization those quantities have to be expressed depending on the displacement $\vec{u}$. One possibility would be to transform the spatial surface vector $\mathrm{d}\vec{a} = \vec{n}\mathrm{d}a$ with the help of Nanson's formula, (Holzap fel, 2000, Eq. (2.55)),

$$\mathrm{d}\vec{a} = \vec{n}\mathrm{d}a = J\mathbf{F}^{-T}\vec{n}_{\mathrm{R}}\mathrm{d}A = J\mathbf{F}^{-T}\mathrm{d}\vec{A}\,, \tag{3.39}$$

where $\vec{n}_{\mathrm{R}}$ denotes the normal in the reference configuration, and linearize the determinant J of the deformation gradient $\mathbf{F}$ (see for example (Wriggers, 2001, p. 91)), as well as the inverse of the deformation gradient (cf. (Wood et al., 1981, p. 50)).

A way which is more suitable for subsequent implementation in a finite element code is the parametrization of the current boundary surface as shown in Fig. 3.2. The region Ω_ξ in the parameter plane is mapped by $\vec{\varphi}$ to the deformed region in Cartesian space. The parametrization of the surface on which the pressure p acts is given as

$$\vec{x}(t) = \vec{\varphi}(t, \xi_1, \xi_2)\,. \tag{3.40}$$

The convective tangent vectors define the basis by

$$\vec{\varphi}_{,1} = \frac{\partial\vec{\varphi}}{\partial\xi_1} \quad \text{and} \quad \vec{\varphi}_{,2} = \frac{\partial\vec{\varphi}}{\partial\xi_2} \tag{3.41}$$

and the outward unit vector $\vec{n}$ can be expressed as the normalized cross product of the convective base vectors

$$\vec{n} = \frac{\vec{\varphi}_{,1} \times \vec{\varphi}_{,2}}{\|\vec{\varphi}_{,1} \times \vec{\varphi}_{,2}\|} \quad . \tag{3.42}$$

The infinitesimal surface element $\mathrm{d}a$ is evaluated after the coordinate transformation as

$$\mathrm{d}a = \det\mathbf{J}\, d\xi d\eta \tag{3.43}$$

where $\det\mathbf{J}$ is the determinant of the Jacobian matrix $\mathbf{J} = \dfrac{\partial\vec{\varphi}}{\partial\vec{\xi}}$ of the mapping $\vec{\varphi}$. As the determinant of the Jacobian is the norm of the cross product of the base vectors, (Westermann, 1997, Sec. 3.2.2),

$$\det\mathbf{J} = \|\vec{\varphi}_{,1} \times \vec{\varphi}_{,2}\| \tag{3.44}$$

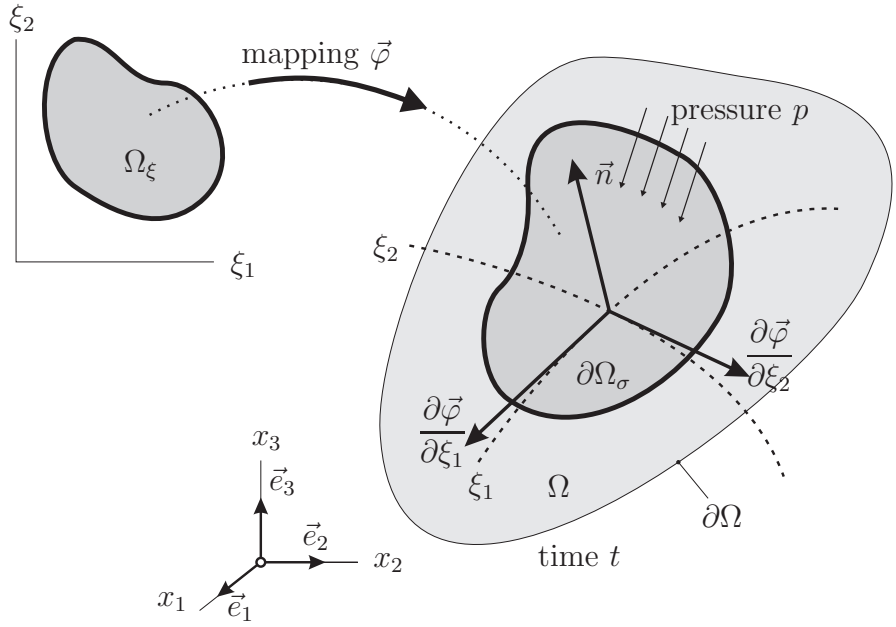


Figure 3.2: Parametrization of the pressure loaded surface $\partial\Omega_\sigma$

we obtain for the differential surface element

$$\mathrm{d}a = \|\vec{\varphi}_{,1} \times \vec{\varphi}_{,2}\| \, \mathrm{d}\xi_1 \mathrm{d}\xi_2. \tag{3.45}$$

Using relations (3.42) and (3.45) in (3.38), the norm $\|\vec{\varphi}_{,1} \times \vec{\varphi}_{,2}\|$ cancels and pulling out the constant pressure magnitude p the external deformation dependent virtual work may be expressed in convective coordinates as

$$\pi_{\mathrm{ext}}(t, \vec{u}, \delta\vec{u}) = -p \int_{\Omega_\xi} (\vec{\varphi}_{,1} \times \vec{\varphi}_{,2}) \cdot \delta\vec{u} \, \mathrm{d}\xi_1 \mathrm{d}\xi_2 = -p \int_{\Omega_\xi} \left(\frac{\partial\vec{\varphi}}{\partial\xi_1} \times \frac{\partial\vec{\varphi}}{\partial\xi_2} \right) \cdot \delta\vec{u} \, \mathrm{d}\xi_1 \mathrm{d}\xi_2 \quad . \tag{3.46}$$

To obtain the directional derivative we apply definition (3.21),

$$\begin{aligned} \mathsf{D}_{\vec{u}} \pi_{\mathrm{ext}}(t, \vec{u}, \delta\vec{u})[\Delta\vec{u}] &= -\frac{\mathrm{d}}{\mathrm{d}\epsilon} \left[p \int_{\Omega_\xi} \left(\frac{\partial(\vec{\varphi} + \epsilon\Delta\vec{u})}{\partial\xi_1} \right) \times \left(\frac{\partial(\vec{\varphi} + \epsilon\Delta\vec{u})}{\partial\xi_2} \right) \cdot \delta\vec{u} \, \mathrm{d}\xi_1 \mathrm{d}\xi_2 \right] \Bigg|_{\epsilon=0} \\ &= -p \int_{\Omega_\xi} (\Delta\vec{u}_{,1} \times \vec{\varphi}_{,2} + \vec{\varphi}_{,1} \times \Delta\vec{u}_{,2}) \cdot \delta\vec{u} \, \mathrm{d}\xi_1 \mathrm{d}\xi_2 \end{aligned} \tag{3.47}$$

where $\Delta\vec{u}_{,1}$ abbreviates $\frac{\partial\Delta\vec{u}}{\partial\xi_1}$.

Axisymmetric/2D case

In the special case of an axisymmetric or two-dimensional domain one of the base vectors coincides with the unit vector $\vec{e}_3$ perpendicular to the plane, see Fig. 3.3. Consequently, the

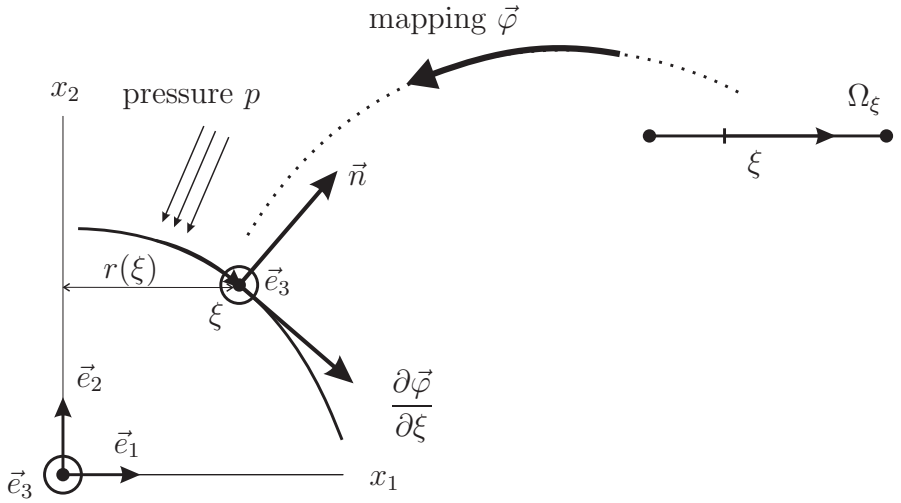


Figure 3.3: Parametrization for pressure loading in two dimensions

virtual work of follower loading, Eq. (3.46), takes the simpler form in the **plain stress/plain strain** case compared to the three dimensional case

$$\pi_{\text{ext}}(t,\vec{u},\delta\vec{u}) = -p\int\limits_{\Omega_\xi}\delta\vec{u}\cdot(\vec{e}_3\times\vec{\varphi}_{,\xi})\;\mathrm{d}\xi = -p\int\limits_{\Omega_\xi}\delta\vec{u}\cdot\left(\vec{e}_3\times\frac{\partial\vec{\varphi}}{\partial\xi}\right)\;\mathrm{d}\xi \tag{3.48}$$

and the directional derivative given in Eq. (3.47) reduces to

$$\mathsf{D}_{\vec{u}}\pi_{\text{ext}}(t,\vec{u},\delta\vec{u})[\Delta\vec{u}] = -p\int\limits_{\Omega_\xi}\delta\vec{u}\cdot\left(\vec{e}_3\times\frac{\partial\Delta\vec{u}}{\partial\xi}\right)\;\mathrm{d}\xi = -p\int\limits_{\Omega_\xi}\delta\vec{u}\cdot(\vec{e}_3\times\Delta\vec{u}_{,\xi})\;\mathrm{d}\xi \quad . \tag{3.49}$$

If **axisymmetric** problems are investigated, the differential $\mathrm{d}\xi$ in (3.48) has to be replaced by $r(\xi)\,\mathrm{d}\xi$, where $r(\xi)$ denotes the radius, i.e. the distance of the investigated point from the axis of rotation in radial direction and the integral must be multiplied by 2π as the integration is carried out over 2π radians to cover the whole volume. Consequently we obtain for the axisymmetric case

$$\pi_{\text{ext}}(t,\vec{u},\delta\vec{u}) = -2\pi p\int\limits_{\Omega_\xi}\delta\vec{u}\cdot(\vec{e}_3\times\vec{\varphi}_{,\xi})r(\xi)\;\mathrm{d}\xi = -2\pi p\int\limits_{\Omega_\xi}\delta\vec{u}\cdot\left(\vec{e}_3\times\frac{\partial\vec{\varphi}}{\partial\xi}\right)r(\xi)\;\mathrm{d}\xi\,. \tag{3.50}$$

To evaluate the directional derivative we note that the term $r(\xi) = X_1 + u_1 = \varphi_1$ depends on the (first, i.e. radial component) of the deformation $\vec{u}$ and has to be derived, too. Therefore, we carry out the directional derivative explicitly according to the definition (3.21) and apply the product rule to obtain the directional derivative in the axisymmetric case

$$\begin{aligned}\mathsf{D}_{\vec{u}}\pi_{\text{ext}}(t,\vec{u},\delta\vec{u})[\Delta\vec{u}] &= -2\pi p\,\frac{\partial}{\partial\epsilon}\left[\int\limits_{\Omega_\xi}\delta\vec{u}\cdot\left(\vec{e}_3\times\frac{\partial}{\partial\xi}\big(\vec{\varphi}+\epsilon\Delta\vec{u}\big)\right)\Big(\varphi_1+\epsilon\Delta u_1\Big)\;\mathrm{d}\xi\right]\Bigg|_{\epsilon=0}\\ &= -2\pi p\int\limits_{\Omega_\xi}\delta\vec{u}\cdot\Big((\vec{e}_3\times\Delta\vec{u}_{,\xi})\varphi_1+(\vec{e}_3\times\vec{\varphi}_{,\xi})\Delta u_1\Big)\;\mathrm{d}\xi\,.\end{aligned} \tag{3.51}$$

Summary

This paragraph collects the expressions of the linearized weak form for later reference. Starting from Eq. (3.27) the individual terms expand as

$$\mathsf{L}[\pi]_{(\vec{u},\mathbf{q})} \;=\; (\pi_{\text{int}} - \pi_{\text{ext}}) + (\mathsf{D}_{\vec{u}}\,\pi_{\text{int}}[\Delta\vec{u}] - \mathsf{D}_{\vec{u}}\,\pi_{\text{ext}}[\Delta\vec{u}]) \tag{3.52}$$

where we linearize around a 'known' state $(\vec{u}, \mathbf{q})$ and are interested in the increment $\Delta\vec{u}$ with

the internal virtual work (3.53)

$$\pi_{\text{int}}(t, \vec{u}, \delta\vec{u}, \mathbf{q}) \;=\; \int_V \delta\mathbf{E}(\vec{X}, t) \cdot \tilde{\mathbf{T}}(\mathbf{C}, \mathbf{q})\; \mathrm{d}V$$

the external virtual work (3.54)

$$\begin{aligned}\pi_{\text{ext}}(t, \vec{u}, \delta\vec{u}) \;=\; & \int_V \delta\vec{u}(\vec{X}) \cdot \rho_{\text{R}}(\vec{X})\vec{k}\; \mathrm{d}V \\ & + \int_A \delta\vec{u}(\vec{X}) \cdot \vec{t}_{\text{R}}(\vec{X}, t)\; \mathrm{d}A \\ & - p \int_{\Omega_\xi} \delta\vec{u} \cdot (\vec{\varphi}_{,1} \times \vec{\varphi}_{,2})\; \mathrm{d}\xi_1 \mathrm{d}\xi_2\end{aligned}$$

dir. derivative of the internal virtual work (3.55)

$$\begin{aligned}\mathsf{D}_{\vec{u}}\,\pi_{\text{int}}(t, \vec{u}, \delta\vec{u}, \mathbf{q})[\Delta\vec{u}] \;=\; & \int_V \operatorname{Grad}\Delta\vec{u}\,\tilde{\mathbf{T}} \cdot \operatorname{Grad}\delta\vec{u}\; \mathrm{d}V + \\ & \int_V \delta\mathbf{E} \cdot \tilde{\mathcal{C}}\; \mathsf{D}_{\vec{u}}\mathbf{E}[\Delta\vec{u}]\; \mathrm{d}V\end{aligned}$$

dir. derivative of the follower loading (3D) (3.56)

$$\mathsf{D}_{\vec{u}}\,\pi_{\text{ext}}(t, \vec{u}, \delta\vec{u})[\Delta\vec{u}] \;=\; -p \int_{\Omega_\xi} \delta\vec{u} \cdot (\Delta\vec{u}_{,1} \times \vec{\varphi}_{,2} + \vec{\varphi}_{,1} \times \Delta\vec{u}_{,2})\; \mathrm{d}\xi_1 \mathrm{d}\xi_2\,.$$

Chapter 4

Discretization

Although the analytical solution of the linearized continuous weak form (3.52) is intractable in most cases, approximate solutions can be obtained by numerical methods. The problem continuous in space and time is discretized leading to a problem with a finite number of unknowns. One could discretize time and space in one step, however in numerical differential calculus it proved useful to start with the discretization of either one of those 'dimensions' and analyze each step separately. For the application of the finite element method one first discretizes the spatial dimension. This is called the vertical method of lines (see Fig. 4.1)

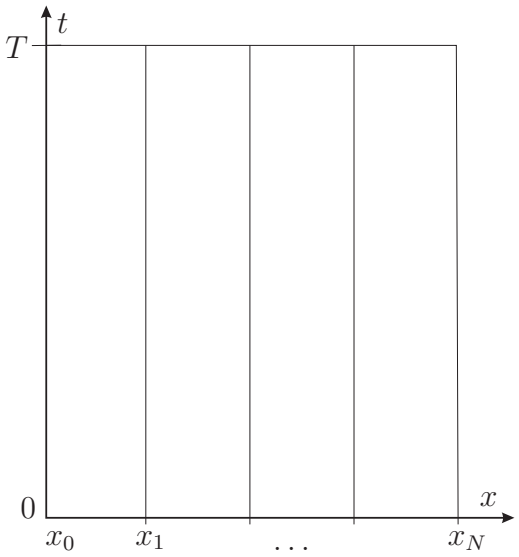


Figure 4.1: Vertical method of lines: spatial discretization

leading to semi-discrete equations still continuous in time. The alternative approach, first carrying out the temporal discretization is called the horizontal method of lines or Rothe-method, see (Grossmann and Roos, 2005, p. 317 vs. 337) and (Solín et al., 2003, Sec. 5.4.1). Now, different methods for temporal discretization can be applied. Historically first backward differentiation formulae (BDF) were used, recently also one-step methods like Runge-Kutta and extrapolation methods are employed too, compare (Solín et al., 2003, Sec. 5.4) for a brief introduction.

4.1 Spatial discretization by the finite element method

For the spatial discretization the continuous vector field $\vec{u}(\vec{X},t)$ is approximated by $\boldsymbol{u}^h(\vec{X},t)$ as a combination of linear independent functions ('modes') $N_i(\vec{X})$. These functions span a finite-dimensional vector space, and are collected in a matrix $\boldsymbol{N}(\vec{X}) \in \mathbb{R}^{n_{\mathrm{sd}} \times n_{\mathrm{dof}}}$ whereas the coefficients $u_{x\,a}(t), u_{y\,b}(t), u_{z\,c}(t)$ populate a vector $\boldsymbol{u}(t)$ that represents the remaining unknowns

$$\vec{u}(\vec{X},t) \approx \vec{u}^h = \boldsymbol{u}^h(\vec{X},t) = \left\{ \begin{array}{c} \sum_{a=1}^{n_{\mathrm{modes}}(x)} N_a(\vec{X}) u_{x\,a}(t) \\ \sum_{b=1}^{n_{\mathrm{modes}}(y)} N_b(\vec{X}) u_{y\,b}(t) \\ \sum_{c=1}^{n_{\mathrm{modes}}(z)} N_c(\vec{X}) u_{z\,c}(t) \end{array} \right\} = \tag{4.1}$$

$$= \left[\begin{array}{cccc} N_1(\vec{X}) & . & . & \cdots \\ . & N_1(\vec{X}) & . & \cdots \\ . & . & N_1(\vec{X}) & \cdots \end{array} \right] \left\{ \begin{array}{c} u_{x\,1}(t) \\ u_{y\,1}(t) \\ u_{z\,1}(t) \\ \vdots \\ u_{x\,n_{\mathrm{modes}}(x)}(t) \\ \vdots \\ u_{y\,n_{\mathrm{modes}}(y)}(t) \\ \vdots \\ u_{z\,n_{\mathrm{modes}}(z)}(t) \end{array} \right\} = \boldsymbol{N}(\boldsymbol{X})\boldsymbol{u}(t). \tag{4.2}$$

Note that the coefficients are not necessarily directly associated to displacements of 'nodes' in the case of p-FEM as will be discussed below.

The resulting total number of degrees of freedom n_{dof} is the sum of modes in each space dimension

$$n_{\mathrm{dof}} = n_{\mathrm{modes}}(x) + n_{\mathrm{modes}}(y) + n_{\mathrm{modes}}(z)\,. \tag{4.3}$$

The notation in Eq. (4.1) already hints at the possibility that the Ansatz space may have different number of basis functions for each spatial dimension x, y, z leading to anisotropic finite elements with advantages, e.g. for the computation of thin-walled structures (Düster, 2001; Scholz, 2006).

The idea of the finite element method that came to light in the 1940s (for the account of an early pioneer see (Clough, 2004)) is centered around the construction of the Ansatz space and the functions N_i. The original domain V with boundary ∂V is approximated by Ω and $\partial\Omega$. Instead of defining the Ansatz functions in a closed form over the whole domain, the domain is split into n_{e} non-overlapping subdomains, called elements

$$\bigcup_{e=1}^{n_{\mathrm{e}}} \Omega^{\mathrm{e}} = \Omega \quad \text{and} \quad \bigcap_{e=1}^{n_{\mathrm{e}}} \Omega^{\mathrm{e}} = \emptyset. \tag{4.4}$$

The union of the elements defines the mesh that either is an exact representation of the

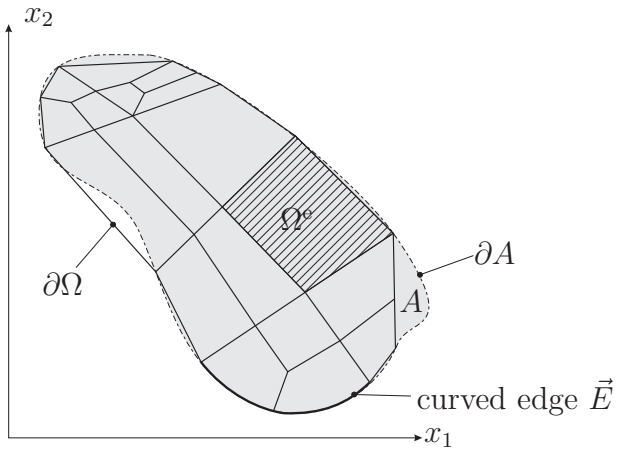


Figure 4.2: Two-dimensional domain A (grey region) and approximated domain Ω (solid lines) consisting of elements Ω^e. The boundary ∂A (dashed line) can be approximated by straight lines or more precisely described for example by the blending function method in its different flavors, viz Sec. 4.3.

original domain V or an approximation to it, compare Fig. 4.2. The global Ansatz functions N_i are piecewise polynomial functions constructed from the union (assembly) of local functions N_i^e defined on the elements Ω^e obeying certain continuity requirements. It is common and beneficial to construct the local functions N_i^e in a way that C^0 continuity of the global function N_i over element boundaries is ensured[1]. For numerical evaluation the functions N_i^e on the element are defined on a standard (reference, master) element $\Omega_{\text{st}} = \{|\xi_i| \leq 1, i = 1, \ldots, n_{\text{sd}}\}$ and mapped to the actual element in space as described in Sec. 4.3. Compare Fig. 4.3 for a sketch of the idea.

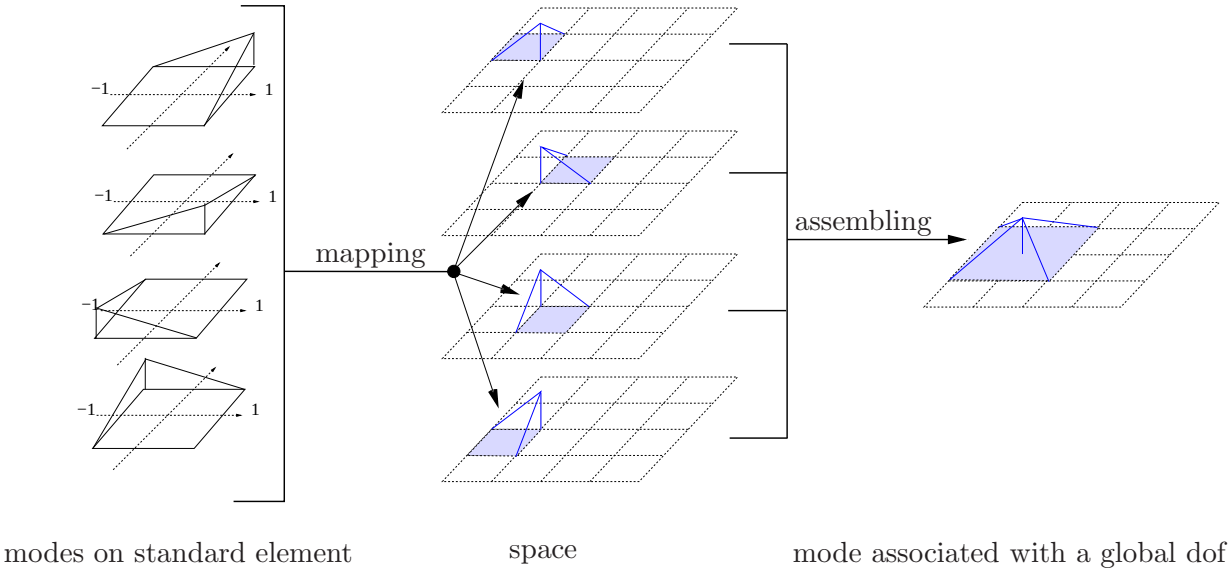


Figure 4.3: Construction of a global Ansatz function from the union of elemental local functions

The choice of the Ansatz functions on the master element and the mapping concept dis-

[1]If this requirement is abandoned, the discontinuous Galerkin method (Zienkiewicz et al., 2003) is recovered.

tinguishes the h- and the p-version and related variants of the finite element method.

- In the h-version, see e.g. (Bathe, 1996; Hughes, 2000; Zienkiewicz and Taylor, 2000b), the discretization error is controlled and convergence is achieved by reducing the element size (h being the size of the largest element in the mesh). Typically low-order Ansatz functions (linear or quadratic) are used keeping the polynomial degree p of the Ansatz functions fixed. The mapping is usually iso-parametric, i.e. the same (low-order) functions used to describe the deformation are applied to approximate the geometry. Consequently curved boundaries must be resolved by many small elements for a good representation. The displacement-based h-version is well established for linear and nonlinear applications but is prone to locking. Therefore, the concept of mixed finite elements was developed to circumvent this problem. Furthermore, the h-version has a lower convergence rate than the p-version for many kinds of problems (Szabó et al., 2004), cf. Sec. 4.4.

- The p-version on the contrary keeps the mesh fixed and achieves convergence by raising the polynomial degree p of the Ansatz functions. As the mesh is usually coarse it is crucial to guarantee a good representation of the boundary, for example by the blending function method described in Sec. 4.3. For smooth problems the p-version achieves exponential convergence and locking is overcome by raising the polynomial degree. This property, well established for small strain elastic problems (for a comprehensive discussion we refer to (Szabó and Babuška, 1991)), also carries over to finite strain hyperelastic applications (Düster et al., 2003; Heisserer et al., 2007; Yosibash et al., 2007) as investigated in Sec. 5.3.

- The hp-version (Demkowicz, 2006; Schwab, 1998) combines the advantages of both approaches. It uses large elements and a high polynomial degree where the solution is smooth and small elements with lower order in regions with strong gradients. This combination allows to achieve exponential convergence also for problems with singularities.

- For problems where different length-scales (multiscale problems) are involved the hp-version can be augmented by the solution on an overlay-mesh resolving regions of rapid variation in the solution or enriching e.g. a two dimensional calculation with three-dimensional analysis of a zoomed detail. The hp-d method was first introduced by Rank (Rank, 1992a,b) and further developed in (Düster, 2001; Düster et al., 2007; Krause, 1996; Krause and Rank, 2003; Niggl et al., 2003; Rank, 1993; Rank and Krause, 1997).

The p-version is accepted as an efficient approach for elliptic linear (Actis et al., 1999; Babuška and Strouboulis, 2001; Babuška and Szabó, 1982; Düster et al., 2001; Holzer et al., 1990; Rank et al., 2002, 1998a, 2001; Schwab, 1998; Szabó and Babuška, 1991) and geometrically nonlinear problems (Krause et al., 1995) but little work has been carried out for physically nonlinear applications. Small strain elastoplasticity with the p-version was investigated by Szabó et al. (Szabó et al., 1995), Holzer and Yosibash (Holzer and Yosibash, 1996), Jeremic and Xenophontos (Jeremic and Xenophontos, 1999). Düster and Rank compared the p-version to an adaptive h-version (Düster and Rank, 2001) for the deformation theory of plasticity. The high-order approach for the J_2 flow theory of plasticity was studied in (Düster et al., 2002; Düster and Rank, 2002). The elasto-plastic modeling of laminated anisotropic plates with

moderately large deflections and small rotations in the sense of von Kármán hypothesis by the p-version was investigated by (Woo et al., 2003).

Nübel (Nübel, 2005; Nübel et al., 2007) proposed an adaptive rp-method for small strain elasto-plastic problems where the mesh is adapted to capture the elasto-plastic interface. The benefit is that exponential convergence is recovered with this strategy.

There are even less investigations for the application of the p-version for *finite* strain *non*linear materials. For finite strain hyperelastic problems there are the papers published by (Düster et al., 2003), (Heisserer et al., 2007) and Yosibash (Yosibash et al., 2007). To our knowledge there exist no publications on the application of the p-version for *finite* strain elasto-plastic and visco-plastic problems that is reported in this work.

4.2 Hierarchical shape functions for high-order finite elements

After introducing different flavors of the finite element discretization we will focus on the p-version in the following. The presentation of the shape functions follows closely (Düster, 2001) and (Szabó and Babuška, 1991).

From a theoretical point of view the finite element space, described by the mesh, the polynomial degrees of the elements and the mapping functions, completely controls the quality of approximation. See (Szabó and Babuška, 1991, Sec. 3.1) for a mathematically rigorous treatment of this topic. Different sets of basis functions can span the same finite element space, however there are some important considerations (compare (Szabó and Babuška, 1991, p. 95)(Düster, 2005)) if finite element procedures are to be implemented on computer hardware with round-off errors and a limited number of digits

1. For a wide range of mapping parameters the round-off error accumulation with respect to increasing polynomial degree should be as small as possible.

2. The shape functions should permit computation of the stiffness matrices and load vectors as efficiently as possible.

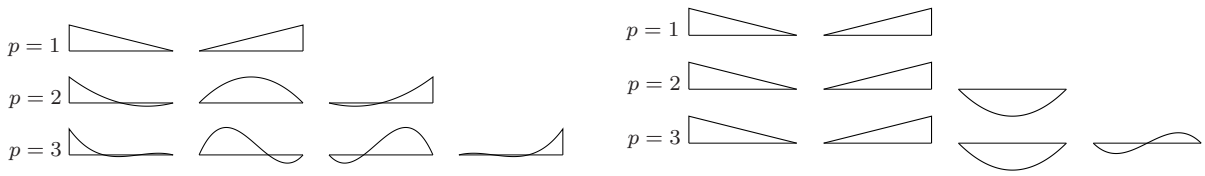
3. The shape functions should permit efficient enforcement of exact and minimal continuity.

4. The choice of the shape functions affects the performance of iterative solution procedures. For large problems this can be the dominant consideration.

For constructing high-order Ansatz spaces Szabó and Babuška (Szabó and Babuška, 1991) promote a hierarchical basis where shape functions of lower order are included in the set of higher-order shape functions. These functions are constructed from orthogonal polynomials with a minimum number of functions not vanishing on nodes, edges or faces. This construction is not obvious as the classical Lagrangian shape functions do not show this property. It is instructive to study the different concepts in one dimension first.

4.2.1 The one-dimensional hierarchic basis

To have a point of comparison we first introduce the classical or standard nodal basis on a standard element $\Omega_{\text{st}} = (-1, 1)$. The set of Lagrange polynomials defines the standard



(a) standard basis (b) hierarchical basis

Figure 4.4: Set of one-dimensional standard and hierarchical basis functions in 1D

high-order shape functions (Fig. 4.4(a)) in one dimension

$$N_i^p(\xi) \;=\; \prod_{j=1,\, j\neq i}^{p+1} \frac{\xi-\xi_j}{\xi_i-\xi_j}\,. \tag{4.5}$$

The points ξ_j where

$$N_i^p(\xi_j) \;=\; \delta_{ij} \tag{4.6}$$

are called nodes. Usually, the nodes are chosen to be equally distributed, i.e.

$$\xi_j \;=\; -1+2\,\frac{j-1}{p}, \quad j=1,...,p+1\,. \tag{4.7}$$

For each polynomial degree p a separate set of shape functions has to be defined, for example, for $p=1$

$$\begin{aligned} N_1^1(\xi) &= 1/2(1-\xi) \\ N_2^1(\xi) &= 1/2(1+\xi) \quad , \end{aligned} \tag{4.8}$$

for $p=2$

$$\begin{aligned} N_1^2(\xi) &= 1/2\,\xi\,(\xi-1) \\ N_2^2(\xi) &= (1+\xi)\,(1-\xi) \\ N_3^2(\xi) &= 1/2\,(\xi+1)\,\xi \quad , \end{aligned} \tag{4.9}$$

for $p=3$

$$\begin{aligned} N_1^3(\xi) &= -1/16\,(3\,\xi+1)\,(3\,\xi-1)\,(\xi-1) \\ N_2^3(\xi) &= 9/16\,(\xi+1)\,(3\,\xi-1)\,(\xi-1) \\ N_3^3(\xi) &= -9/16\,(\xi+1)\,(3\,\xi+1)\,(\xi-1) \\ N_4^3(\xi) &= 1/16\,(\xi+1)\,(3\,\xi+1)\,(3\,\xi-1) \end{aligned} \tag{4.10}$$

etc. Note that the sum of all Lagrange polynomials for a given polynomial degree p equals unity

$$\sum_{i=1}^{p+1} N_i^p(\xi) \;=\; 1\,, \tag{4.11}$$

in the case of the hierarchic basis introduced below this is true only for the nodal modes (4.24).

The space spanned by the standard basis can also be represented by a hierarchical basis, compare Fig. 4.4(b). Note that the set of higher-order basis functions contains all lower order shape functions. The set of one-dimensional hierarchic shape functions, introduced by (Szabó and Babuška, 1991) is shown in Fig. 4.4(b) and given by

$$N_1(\xi) = 1/2(1-\xi) \tag{4.12}$$
$$N_2(\xi) = 1/2(1+\xi) \tag{4.13}$$
$$N_i(\xi) = \phi_{i-1}(\xi), \quad i = 3,4,...,p+1 \tag{4.14}$$

with the integrated Legendre polynomials

$$\phi_j(\xi) = \sqrt{\frac{2j-1}{2}} \int_{-1}^{\xi} L_{j-1}(x)\,dx = \frac{1}{\sqrt{4j-2}}(L_j(\xi) - L_{j-2}(\xi))\,, \quad j = 2,3,... \tag{4.15}$$

where $L_j(\xi)$ are the well-known Legendre polynomials that can be computed either by applying the Rodriguez formula

$$L_n(x) = \frac{1}{2^n\,n!}\frac{d^n}{d\,x^n}\,(x^2-1)^n\,, \quad x \in (-1,1), \quad n = 0,1,2,... \tag{4.16}$$

or Bonnet's recursion formula

$$L_n(x) = \frac{1}{n}\left[(2n-1)xL_{n-1}(x) - (n-1)L_{n-2}(x)\right], \quad x \in (-1,1), \quad n = 2,3,4,... \tag{4.17}$$

Legendre polynomials are orthogonal on $I = (-1,1)$:

$$\int_{-1}^{1} L_n(x)L_m(x)\,dx = \begin{cases} \dfrac{2}{2n+1} & \text{if} \quad n = m \\ 0 & \text{else} \end{cases}\,. \tag{4.18}$$

The linear functions $N_1(\xi), N_2(\xi)$ are called *nodal shape functions* or *nodal modes* and constitute a partition of unity

$$N_1(\xi) + N_2(\xi) = \frac{1}{2}(1-\xi) + \frac{1}{2}(1+\xi) = 1\,. \tag{4.19}$$

Because the functions $N_i(\xi)$, $i = 3,4,...$ vanish at the boundary of the domain

$$N_i(-1) = N_i(1) = 0, \quad i = 3,4,... \tag{4.20}$$

they are called *internal shape functions*, *internal modes* or *bubble modes*. The orthogonality property of Legendre polynomials implies

$$\int_{-1}^{1} \frac{d\,N_i}{d\,\xi}\,\frac{d\,N_j}{d\,\xi}\,d\,\xi = \delta_{ij}, \quad i \geq 3 \text{ and } j \geq 1 \quad \text{or} \quad i \geq 1 \text{ and } j \geq 3\,. \tag{4.21}$$

The hierarchic shape functions for $p = 1, 2, 3, ..., 8$ are given by

$$\begin{aligned}
N_1(\xi) &= 1/2(1-\xi)\,, \\
N_2(\xi) &= 1/2(1+\xi)\,, \\
N_3(\xi) &= 1/4\,\sqrt{6}\left(\xi^2-1\right)\,, \\
N_4(\xi) &= 1/4\,\sqrt{10}\left(\xi^2-1\right)\xi\,, \\
N_5(\xi) &= 1/16\,\sqrt{14}\left(5\,\xi^4-6\,\xi^2+1\right)\,, \\
N_6(\xi) &= 3/16\,\sqrt{2}\xi\,\left(7\,\xi^4-10\,\xi^2+3\right)\,, \\
N_7(\xi) &= 1/32\,\sqrt{22}\left(21\,\xi^6-35\,\xi^4+15\,\xi^2-1\right)\,, \\
N_8(\xi) &= 1/32\,\sqrt{26}\xi\,\left(33\,\xi^6-63\,\xi^4+35\,\xi^2-5\right)\,, \\
N_9(\xi) &= 1/256\,\sqrt{30}\left(-140\,\xi^2-924\,\xi^6+630\,\xi^4+5+429\,\xi^8\right)\,.
\end{aligned} \tag{4.22}$$

Hierarchic shape functions result in a better conditioning of the one-dimensional element stiffness matrix, see (Düster, 2001, Sec. 2.2.4). That hierarchic shape functions also have benefits compared to Lagrangian shape functions in higher dimensions was shown in (Zienkiewicz and Craig, 1986).

Building on the one dimensional hierarchical basis functions introduced above hierarchical shape functions for two- and three-dimensional quadrilateral and hexahedral elements can be constructed as the tensor product of the one-dimensional hierarchic shape functions.

4.2.2 Hierarchic shape functions for quadrilaterals

As most of the numerical examples and applications given in this thesis are either two-dimensional or axisymmetric it is sufficient to introduce the hierarchic shape functions for two-dimensional elements.

The implementation of the p-version in two dimensions is based on a quadrilateral element formulation, using the Ansatz functions introduced by Szabó and Babuška (Szabó and Babuška, 1991). Fig. 4.5 depicts the standard quadrilateral finite element.

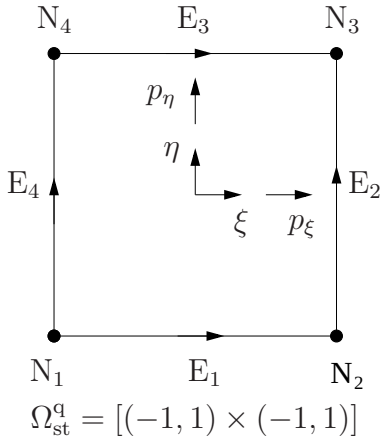


Figure 4.5: Standard quadrilateral element: definition of nodes, edges and polynomial degree

Two different types of Ansatz spaces are taken into consideration: the *trunk space* $\mathcal{S}_{\rm ts}^{p_\xi,p_\eta}(\Omega_{\rm st}^{\rm q})$ and the *tensor product space* $\mathcal{S}_{\rm ps}^{p_\xi,p_\eta}(\Omega_{\rm st}^{\rm q})$. The *tensor product space* $\mathcal{S}_{\rm ps}^{p_\xi,p_\eta}(\Omega_{\rm st}^{\rm q})$ consists of all polynomials on $\Omega_{\rm st}^{\rm q} = [(-1,1)\times(-1,1)]$ spanned by the set of monomials

- $\xi^i\,\eta^j \quad$ with $\quad i = 0,1,....,p_\xi, \quad j = 0,1,....,p_\eta,$

whereas the *trunk space* $\mathcal{S}_{\rm ts}^{p_\xi,p_\eta}(\Omega_{\rm st}^{\rm q})$ on $\Omega_{\rm st}^{\rm q} = [(-1,1)\times(-1,1)]$ is spanned by the subset containing the following monomials

- $\xi^i\,\eta^j \quad$ with $\quad i = 0,...,p_\xi, \quad j = 0,...,p_\eta, \quad i+j = 0,...,\max\{p_\xi,p_\eta\}$
- $\xi\,\eta \quad$ for $\quad p_\xi = p_\eta = 1$
- $\xi^{p_\xi}\,\eta \quad$ for $\quad p_\xi \geq 2$
- $\xi\,\eta^{p_\eta} \quad$ for $\quad p_\eta \geq 2 \quad .$

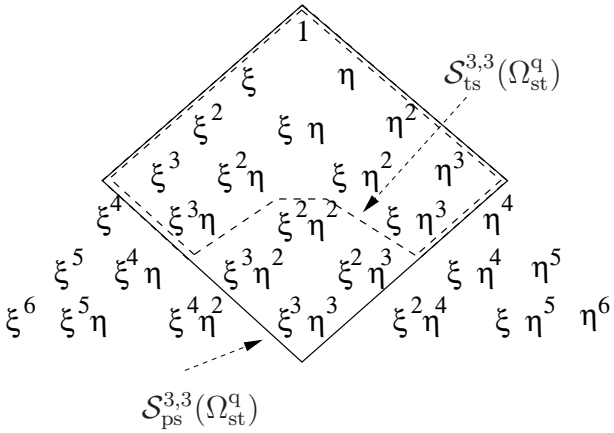


Figure 4.6: The trunk space $\mathcal{S}_{\rm ts}^{3,3}(\Omega_{\rm st}^{\rm q})$ and the tensor product space $\mathcal{S}_{\rm ps}^{3,3}(\Omega_{\rm st}^{\rm q})$

The difference between the two Ansatz spaces can be readily visualized when considering the spanning sets in Pascal's triangle. Fig. 4.6 depicts the set of monomials for $p_\xi = p_\eta = 3$ for both the trunk and the tensor product space. All monomials inside the dashed line span the trunk space $\mathcal{S}_{\rm ts}^{3,3}(\Omega_{\rm st}^{\rm q})$ whereas the monomials bordered by the solid line are contained in the tensor product space $\mathcal{S}_{\rm ps}^{3,3}(\Omega_{\rm st}^{\rm q})$.

By construction, the two-dimensional shape functions can be classified into three groups:

1. **Nodal modes:** The four nodal modes

$$N_{1,1}^{{\rm N}_i}(\xi,\eta) \;=\; \frac{1}{4}\,(1+\xi_i\,\xi)(1+\eta_i\,\eta)\,, \quad i = 1,...,4 \tag{4.23}$$

are the standard bilinear shape functions, well known from the isoparametric four-noded quadrilateral element. (ξ_i,η_i) denote the local coordinates (±1,±1) of the ith node. Fig. 4.7 depicts the mode for node 1. The sum of the four nodal modes equals unity

$$\sum_{i=1}^{4} N_{1,1}^{{\rm N}_i}(\xi,\eta) \;=\; 1\,, \tag{4.24}$$

a property important for the calculation of reaction forces in Sec. 4.9. Note that in the case of a Lagrangian nodal basis *all* shape functions for a given polynomial degree represent a partition of unity (Eq. (4.11)), whereas with the hierarchic basis introduced here this property is only valid for the nodal modes.

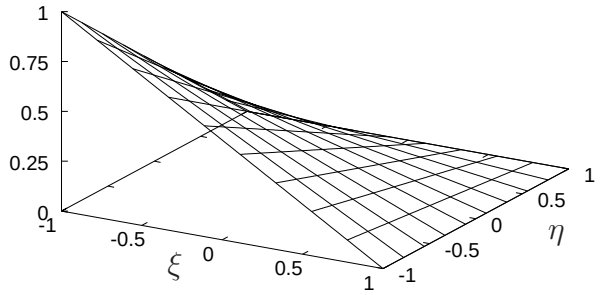


Figure 4.7: Node 1: $N_{1,1}^{\mathrm{N}_1}(\xi,\eta) = \frac{1}{4}(1-\xi)(1-\eta)$

2. **Edge modes:** There are $2(p_\xi - 1) + 2(p_\eta - 1)$ side modes for $p \geq 2$. These modes are defined separately for each individual edge, they vanish at all other edges. The corresponding $2(p_\xi - 1)$ modes for edge E_1 and E_3 in ξ-direction read:

$$N_{i,1}^{\mathrm{E}_1}(\xi,\eta) = \frac{1}{2}\phi_i(\xi)(1-\eta), \qquad i = 2,\ldots,p_\xi\,, \tag{4.25}$$

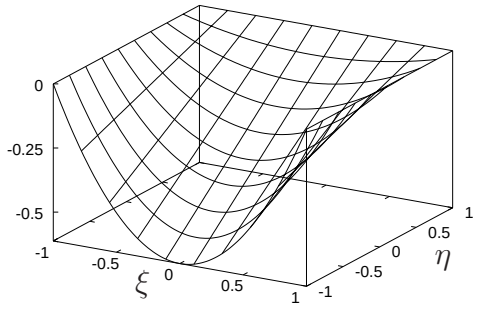
$$N_{i,1}^{\mathrm{E}_3}(\xi,\eta) = \frac{1}{2}\phi_i(\xi)(1+\eta), \qquad i = 2,\ldots,p_\xi\,, \tag{4.26}$$

while for edges E_2 and E_4 the $2(p_\eta - 1)$ modes in η-direction are

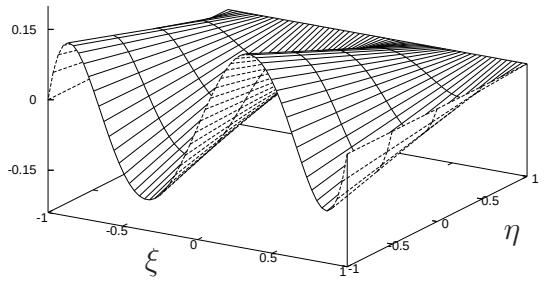
$$N_{1,i}^{\mathrm{E}_2}(\xi,\eta) = \frac{1}{2}(1+\xi)\phi_j(\eta), \qquad j = 2,\ldots,p_\eta\,, \tag{4.27}$$

$$N_{1,i}^{\mathrm{E}_4}(\xi,\eta) = \frac{1}{2}(1-\xi)\phi_j(\eta), \qquad j = 2,\ldots,p_\eta\,, \tag{4.28}$$

with ϕ_i defined in Eq. (4.15). In Fig. 4.8 the modes for edge 1 with $i = 2$ and $i = 5$ are plotted.



(a) $(i = 2)$: $N_{2,1}^{\mathrm{E}_1}(\xi,\eta)$



(b) $(i = 5)$: $N_{5,1}^{\mathrm{E}_1}(\xi,\eta)$

Figure 4.8: Two representative higher order edge modes

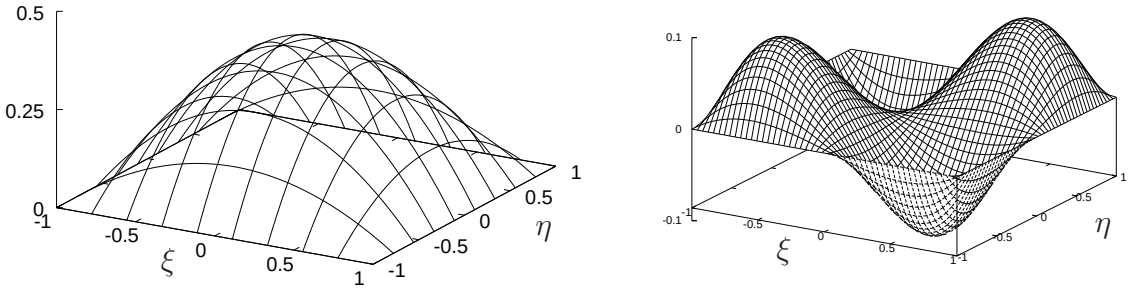
3. **Internal modes (bubble modes):** There are $(p-1)(q-1)$ internal modes $(p,q \geq 2)$ for the tensor product space $\mathcal{S}_{\mathrm{ps}}^{p_\xi,p_\eta}(\Omega_{\mathrm{st}}^{\mathrm{q}})$. For the trunk space $\mathcal{S}_{\mathrm{ts}}^{p_\xi,p_\eta}(\Omega_{\mathrm{st}}^{\mathrm{q}})$ the internal modes enter the scene later, for $p \geq 4$,

$$N_{i,j}^{\mathrm{int}}(\xi,\eta) = \phi_i(\xi)\phi_j(\eta) \tag{4.29}$$

$$\text{for} \quad \begin{cases} \text{product Sp.} & i = 2,...,p_\xi - 2, j = 2,...,p_\eta - 2 \\ \text{trunk Sp.} & i = 2,...,p_\xi - 4, j = 2,...,p_\eta - 4, \\ & i + j = 4,...,\max\{p_\xi, p_\eta\}. \end{cases}$$

The internal modes are purely local and vanish at the edges of the quadrilateral element. This provides the possibility to use static condensation on element level, a kind of Schur complement preconditioning, cf. Sec. 4.8. The computational effort on element level pays off for the solution of the global equation system by iterative solvers as the condition number of the global stiffness matrix decreases significantly, compare (Ainsworth, 1996; Mandel, 1990a,b). To give an impression of the rapid growth of the internal degrees of freedom when the tensor product space is used a comparison of the fraction of purely internal degrees of freedom to the total degrees of freedom is plotted in (Düster, 2001, p. 24). For a two-dimensional Lamé problem discretized with one quadrilateral element and uniform $p = 4$ the product space contains about 40% internal degrees of freedom while the trunk space exhibits only about 5%.

Fig. 4.9 depicts the internal modes for $i = j = 2$ and $i = j = 3$.



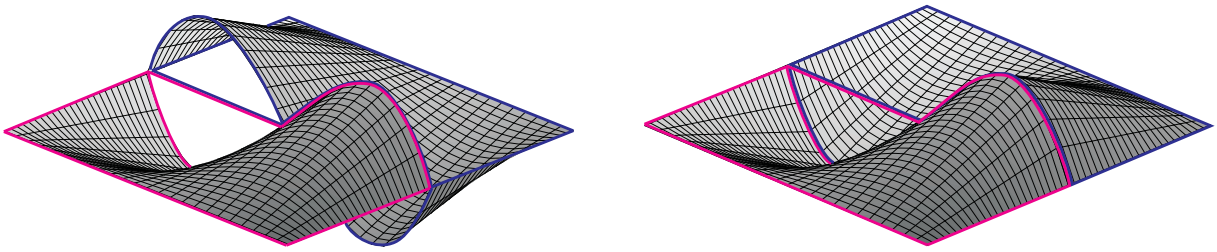
(a) $(i = j = 2)$: $N_{2,2}^{\mathrm{int}}(\xi,\eta)$ (b) $(i = j = 3)$: $N_{3,3}^{\mathrm{int}}(\xi,\eta)$

Figure 4.9: Two representative higher order internal modes

As already indicated, the indices i, j of the shape functions denote the polynomial degrees in the local directions ξ, η.

4.2.3 Inter-element continuity

Fig. 4.3 sketches how the union of shape functions defined on the individual elements form a global shape function pertaining to a certain global degree of freedom. If — as in our case — high-order shape functions are used special care has to be take to ensure C^0 continuity in the displacements over element edges and faces. To this end all edges bear an orientation and in setting up the elemental shape function this information is used for the construction of the Ansatz function. Fig. 4.10 shows the importance of this consideration for the union of two edge shape functions, $p = 3$. Further details can be found in (Heisserer, 2007, Ch. 2.8) and (Düster, 2005, Ch. 5.2).



(a) Discontinuous union of elemental functions (b) Continuous global function

Figure 4.10: Union of elemental shape functions to a global shape function

4.3 Mapping

In the p-version the element size is not reduced as the number of degrees of freedom is increased as in the h-version. Consequently, the description of the geometry must be independent of the number of elements. The need arises to construct elements with an accurate representation of the boundary. The blending function method described in the following provides this possibility.

In Sec. 4.2.2 the shape functions were defined on the standard element $\Omega^{\mathrm{q}}_{\mathrm{st}}$. To relate them to the element in the reference configuration a mapping $\boldsymbol{\phi}^e(\boldsymbol{\xi}) = \{\boldsymbol{\phi}^e_X(\boldsymbol{\xi}), \boldsymbol{\phi}^e_Y(\boldsymbol{\xi})\}^T$ of the four-noded standard element $\Omega^{\mathrm{q}}_{\mathrm{st}} = [(-1,1) \times (-1,1)]$ with local coordinates $\boldsymbol{\xi} = \{\xi, \eta\}^T$ to a general four-noded element in the global reference configuration with coordinates

$$\vec{X} = \{X, Y\}^T = \boldsymbol{\phi}^e(\boldsymbol{\xi}) \tag{4.30}$$

is defined as sketched in Fig. 4.11. The inverse mapping is introduced as

$$\boldsymbol{\xi} = \{\xi, \eta\}^T = (\boldsymbol{\phi}^e)^{-1}(\vec{X})\,. \tag{4.31}$$

The nodes $\vec{X}_i = \{X_i, Y_i\}^T$, $i = 1, 2, 3, 4$ and the four edges $\vec{E}_i = \{E_{iX}, E_{iY}\}^T$, $i = 1, 2, 3, 4$ given in parametric description define the geometry of the element in two-dimensional Euclidian space.

In the example depicted in Fig. 4.11 where only edge 2 is curved the transformation from

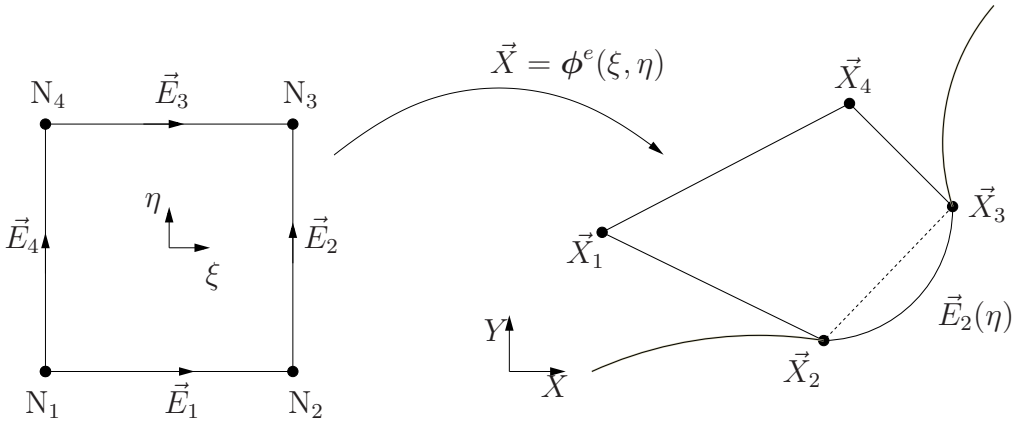


Figure 4.11: Mapping function of four-noded elements

local to global coordinates can be described by the two functions

$$\begin{aligned} X &= \phi_X^e(\xi,\eta) = \sum_{i=1}^{4} N_{1,1}^{\mathrm{N}_i}(\xi,\eta)\, X_i + \left(E_{2X}(\eta) - \left(\frac{1-\eta}{2} X_2 + \frac{1+\eta}{2} X_3 \right) \right) \frac{1+\xi}{2} \\ Y &= \phi_Y^e(\xi,\eta) = \sum_{i=1}^{4} N_{1,1}^{\mathrm{N}_i}(\xi,\eta)\, Y_i + \left(E_{2Y}(\eta) - \left(\frac{1-\eta}{2} Y_2 + \frac{1+\eta}{2} Y_3 \right) \right) \frac{1+\xi}{2} \end{aligned} \tag{4.32}$$

The idea of the blending function method as proposed by Gordon and Hall (Gordon and Hall, 1973a,b) is to augment the standard bilinear mapping given by the first term[2]

$$\vec{X} = \sum_{i=1}^{4} N_{1,1}^{\mathrm{N}_i}(\xi,\, \eta)\, \vec{X}_i \tag{4.33}$$

with the blended difference between the straight connection of the nodes 2 and 3 and the curved edge $\vec{E}_2$, where $N_{1,1}^{\mathrm{N}_i}$ are the nodal modes[3] defined in Eq. (4.23). This difference is linearly blended out towards the opposite edge by the term $\dfrac{1+\xi}{2}$ that is 1 along edge $\vec{E}_2$ and evaluates to 0 along the opposite edge bounded by the nodes 1 and 4.

The generalization to the case where all four edges are possibly curved is given by

$$\vec{X} = \boldsymbol{\phi}^e(\xi,\eta) = \sum_{i=1}^{4} N_{1,1}^{\mathrm{N}_i}(\xi,\, \eta)\, \vec{X}_i + \sum_{i=1}^{4} \mathbf{e}_i(\xi,\, \eta) \tag{4.34}$$

[2]The bilinear mapping contains affine maps as special cases as can be seen in Fig. 4.16 on p. 81.

[3]$(\bullet)_{1,1}$ denotes no derivative here but the polynomial orders in ξ and η direction

where the edge blending terms are defined as

$$\begin{aligned}
\mathbf{e}_1(\xi,\eta) &= \left[\vec{E}_1(\xi) - \frac{(1-\xi)\vec{X}_1 + (1+\xi)\vec{X}_2}{2}\right]\left(\frac{1-\eta}{2}\right)\\
\mathbf{e}_2(\xi,\eta) &= \left[\vec{E}_2(\eta) - \frac{(1-\eta)\vec{X}_2 + (1+\eta)\vec{X}_3}{2}\right]\left(\frac{1+\xi}{2}\right)\\
\mathbf{e}_3(\xi,\eta) &= \left[\vec{E}_3(\xi) - \frac{(1-\xi)\vec{X}_4 + (1+\xi)\vec{X}_3}{2}\right]\left(\frac{1+\eta}{2}\right)\\
\mathbf{e}_4(\xi,\eta) &= \left[\vec{E}_4(\eta) - \frac{(1-\eta)\vec{X}_1 + (1+\eta)\vec{X}_4}{2}\right]\left(\frac{1-\xi}{2}\right).
\end{aligned} \tag{4.35}$$

A mapping $\vec{X} = \boldsymbol{\phi}(\boldsymbol{\xi})$ is characterized by its Jacobian matrix[4]

$$\mathbf{J} = \frac{\partial \boldsymbol{\phi}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \frac{\partial \vec{X}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \begin{bmatrix} \dfrac{\partial X_1}{\partial \xi_1} & \dfrac{\partial X_2}{\partial \xi_1} \\ \dfrac{\partial X_1}{\partial \xi_2} & \dfrac{\partial X_2}{\partial \xi_2} \end{bmatrix}. \tag{4.36}$$

To ensure bijective mapping and the preservation of orientation the determinant of the Jacobian matrix of the mapping must be strictly positive

$$\det\mathbf{J} > 0\,. \tag{4.37}$$

4.3.1 Mapping concepts

As the mapping has a vital influence on the finite element space — defined by the mesh, the shape functions and the mapping — different mapping concepts are distinguished in the following (compare (Düster, 2001, Sec. 3.4), (Bröker, 2001, Sec. 4) and (Nübel, 2005, Sec. 4.4))

- *Iso-parametric mapping*: The same functions used to approximate the displacement are also used for the description of the geometry $\vec{X} = \boldsymbol{\phi}(\boldsymbol{\xi})$, e.g. for the evaluation of the Jacobian needed in the computations of integrals to calculate the discretized load vectors and matrices on the standard element (e.g. in Eq. 4.170). The function space for the mapping $\mathcal{S}_{\text{geo}}$ is equal to the function space $\mathcal{S}_{\text{disp}}$, in symbolic notation $\mathcal{S}_{\text{geo}} = \mathcal{S}_{\text{disp}}$.
- *Sub-parametric mapping*: The functions to describe the geometry are a subset of the function space $\mathcal{S}_{\text{disp}}$ used for the approximate solution $\vec{u}^h \approx \vec{u}_{\text{ex}}$, $\mathcal{S}_{\text{geo}} \subset \mathcal{S}_{\text{disp}}$.
- *Super-parametric mapping*: The functions used to describe the geometry are more complex than the functions $\vec{u}^h$ used to approximate the exact solution $\vec{u}_{\text{ex}}$. To avoid comparing apples and oranges it must be clarified what 'more complex' means. We do *not*

[4] In the literature often the Jacobian matrix is defined as the transpose of the form given here. Our definition is in accordance with (Szabó and Babuška, 1991, p. 113), Wriggers (Wriggers, 2001, p. 106) and Bathe (Bathe, 1996, p. 346), but transposed of the widely used convention in classical FE-texts as for example (Zienkiewicz and Taylor, 1989).

understand it in the strict hierarchic sense of set theory, where one would understand the super-set to include the sub-set like $\mathcal{S}_{\text{geo}} \supset \mathcal{S}_{\text{disp}}$. Instead we use super-parametric in a wider sense where the mapping functions can not be represented in the space of the displacement. For example, this is not only the case when the geometry is expressed by polynomials of higher order than those defining the FE space in which the approximation $\vec{u}^{\,h}$ lives, but also if the geometry is given in non-polynomial terms (like trigonometric functions). Expressed in symbolic notation we have $\mathcal{S}_{\text{geo}} \notin \mathcal{S}_{\text{disp}}$.

Exact, interpolated and approximated geometry

The blending function method as given in Eq. (4.32) relies on the parametric description of the geometry of the edges $\vec{E}_i(r)$. Different approaches are presented in (Bröker, 2001) and (Nübel, 2005, Sec. 4.4.3) for this task.

- Using the blending function method with the exact boundary representation $\vec{E}$ as introduced in Eq. (4.32) has the advantage of evaluating the Jacobian matrices based on the exact geometry. The prerequisite however is that the geometry is available analytically in parameterized form.

- One possibility besides the exact representation is the quasi-regional mapping described by (Királyfalvi and Szabó, 1997). It uses an interpolation

$$\vec{E}_i(r) \approx \boldsymbol{E}_i^{\,\text{interp}}(r) = \sum_{k=1}^{p_{\text{geo}}+1} N_k^{p_{\text{geo}}}(r) \boldsymbol{E}_i(r_k) \tag{4.38}$$

  according to Chen and Babuška (Chen and Babuška, 1995). Because for the interpolation Lagrange polynomials $N_k^{p_{\text{geo}}}$, see Eq. (4.5), are used that evaluate to 1 at a node and vanish at all other nodes the coefficients for the polynomial representation can be obtained without solving an equation system: the coefficients $\boldsymbol{E}_i(r_k)$ are the values of the exact geometry at the nodal or collocation points r_k.

  The locations r_k of the sampling points determine the quality of the approximation as they control the oscillations of the basis functions. A good set of basis functions has minimal oscillations of its sum

$$\lambda(r) = \sum_{k=1}^{p_{\text{geo}}+1} \left| N_k^{p_{\text{geo}}}(r) \right| . \tag{4.39}$$

  In (Chen and Babuška, 1995) an optimal set of sampling points for a given order of geometric approximation p_{geo} is derived by minimizing the Lebesgue constant of the set of sampling points. This results in non-equidistant nodes, the so called Chen-Babuška points.

- A third approach would be to approximate the exact geometry with the same hierarchic shape functions (viz Sec. 4.2.2) as the displacements

$$\vec{E}_i(r) \approx \boldsymbol{E}_i^{\,\text{approx}}(r) = \sum_{i=1}^{n_{\text{modes}}} N_i(r)\, \mathbf{a}_i. \tag{4.40}$$

Using this iso-parametric approach the rigid body modes can be exactly represented, however it requires the solution of a small equation system to obtain the coefficients $\mathbf{a}_i$, see (Demkowicz, 2004) and for a similar problem Remark 13 on p. 97. A related approach is recently introduced as 'iso-geometric analysis' by Hughes et al. (Bazilevs et al., 2006; Cottrell et al., 2006; Hughes et al., 2005) where, instead of the hierarchic shape functions introduced here, NURBS (Non-Uniform Rational B-Splines) or a super-set of them, T-splines (Sederberg et al., 2004, 2003), are used for both the Ansatz space, i.e. for $\vec{u}^h$ and for the exact representation of the geometry $\vec{X} = \boldsymbol{\phi}(\boldsymbol{\xi})$. The interesting consequence is that all rigid body motions and constant strain states can be represented exactly and the geometry can be used as given from a CAD system in its native representation.

4.3.2 Representation of rigid body modes

The different approaches have consequences for the modeling of rigid body modes $\vec{u}_{\text{rbm}}$. It is the defining property that rigid body motion does not cause any strains. In two dimensions all rigid body modes can be described by the linear combinations of two translations (governed by the scalars C_1, C_2) and one rotation

$$\vec{u}_{\text{rbm}} = C_1 \left\{ \begin{array}{c} 1 \\ 0 \end{array} \right\} + C_2 \left\{ \begin{array}{c} 0 \\ 1 \end{array} \right\} + \mathbf{C}_3(\alpha) \left\{ \begin{array}{c} x \\ y \end{array} \right\}, \tag{4.41}$$

where $\mathbf{C}_3$ is the rotation matrix for angle α. To describe the rotation we are interested in the displacement $\vec{u}$ corresponding to the rotation $\mathbf{R}(\alpha)$ of point $\vec{x}$ into $\vec{x}'$, $\vec{x}' = \mathbf{R}(\alpha)\vec{x}$. Hence we solve $\vec{x}' = \vec{x} + \vec{u}$ for the displacement and obtain $\vec{u} = \vec{x}' - \vec{x}$, i.e. $\vec{u} = \mathbf{R}(\alpha)\vec{x} - \vec{x} = \mathbf{C}_3\vec{x}$. Consequently, we obtain for the coefficient matrix $\mathbf{C}_3(\alpha) = \mathbf{R}(\alpha) - \mathbf{I}$ evaluating the *displacements* corresponding to a rotation of point $\vec{x}$ around the origin,

$$\mathbf{C}_3(\alpha) = \left[\begin{array}{cc} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{array} \right] - \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} \right] = \left[\begin{array}{cc} \cos\alpha - 1 & -\sin\alpha \\ \sin\alpha & \cos\alpha - 1 \end{array} \right]. \tag{4.42}$$

The representation of those rigid-body modes is only ensured if the function space for the description of the geometry is a subspace of the finite element approximation. Therefore, a general rigid body mode is represented exactly only if

$$\boldsymbol{u}^h = \vec{u}_{\text{rbm}} \tag{4.43}$$

$$\left\{ \begin{array}{c} \sum_{a=1}^{n_{\text{modes}}(x)} N_a(\xi,\eta) u_{x\,a} \\ \sum_{b=1}^{n_{\text{modes}}(y)} N_b(\xi,\eta) u_{y\,b} \end{array} \right\} = C_1 \left\{ \begin{array}{c} 1 \\ 0 \end{array} \right\} + C_2 \left\{ \begin{array}{c} 0 \\ 1 \end{array} \right\} + \mathbf{C}_3(\alpha) \left\{ \begin{array}{c} \boldsymbol{\phi}^e_X(\xi,\eta) \\ \boldsymbol{\phi}^e_Y(\xi,\eta) \end{array} \right\}$$

holds. This is always guaranteed for iso-parametric or sub-parametric mapping. Concerning the super-parametric concept the rigid body translations governed by C_1 and C_2 can always be represented by the nodal modes in the hierarchic Ansatz space given in Sec. 4.2.2. For the rigid body rotations specified by $\mathbf{C}_3(\alpha)\boldsymbol{\phi}^e$, however, there is no counterpart in the finite element space. This leads to artificial strains and stresses in the solution, compare (Bröker, 2001, Sec. 4.7). The error caused can be minimized in the p-version if the description of the

geometry is sufficiently smooth and the polynomial degree is raised, cf. (Szabó and Babuška, 1991), (Düster, 2001, p. 39).

The following example investigates *finite* deformation rigid body rotation and discusses the error introduced by super-parametric mapping.[5] A quarter of an annulus that is discretized by only one element with polynomial degree p is rotated $\alpha = 45$ and $\alpha = 90$ degrees by prescribing the displacement along the edges[6] $1-2$ and $3-4$, cf. Fig. 4.12. This corresponds to a rigid body mode $\vec{u}_{\text{rbm}}$ with the coefficients $C_1 = 0$, $C_2 = 0$ and $C_3(\alpha = \{\pi/4, \pi/2\})$ in Eq. (4.41). The arcs, $r_i = 10$ [m] and $r_a = 30$ [m], are exactly described using trigonometric functions applying the blending-function concept. For the plain stress investigation a linear

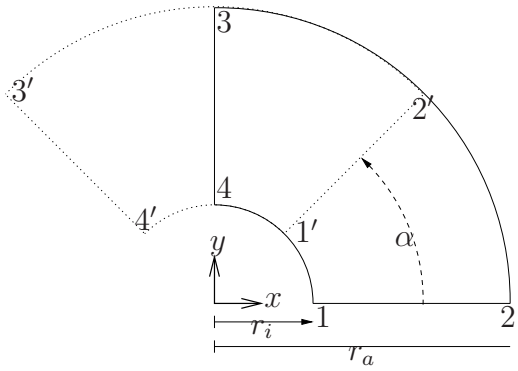


Figure 4.12: Rotated annulus, here $\alpha = 45$ degree

material model (St.Venant-Kirchhoff) with Young's modulus $E = 200.000$ [MPa] and Poisson ratio $\nu = 0.3$ is used. To rule out any influence of the quadrature scheme 30×30 integration points are used in the element for all runs. As rigid body rotation must not induce strains by definition, the numerically calculated strain energy corresponds to the error in energy norm caused by the super-parametric mapping. This error is plotted in a double logarithmic scale versus the degrees of freedom in Fig. 4.13.

The error decreases to machine precision as the polynomial order is raised. For example, with only one element and $p = 9$ the error is smaller than 0.1% for $\alpha = 90$ degrees and the tensor product space. The parallel curves in Fig. 4.13 show that there is only a small influence of the rotation angle α on the error, but a great difference in the number of degrees of freedom when the trunk and the tensor product space is compared, as can be explained by Pascal's triangle, Fig. 4.6. The polynomial degree in each direction is the same but the tensor product space has much more internal modes, i.e. more degrees of freedom that do not give an additional contribution for the approximation of the trigonometric functions. The fading out of the error in stresses with increasing polynomial degree along a cutline ($x = y$ in reference

[5] If the calculations are carried out using a linearized kinematic, i.e. engineering strains where the nonlinear terms in Eq. (4.48) are neglected, there are already on the analytical level spurious strains introduced by rotation as the engineering strain can not be used for large deformations. However, the Green-Lagrange strain is constructed in a way that finite rotations do not induce artificial strains, hence spurious strains occurring in the numerical treatment can be attributed to the numerical method.

[6] This is achieved by prescribing the displacement $\overline{u}_x$ and $\overline{u}_y$ at the points $1, 2, 3, 4$. The higher order inner modes along boundary $1-2$ and $3-4$ are set to vanish by the penalty method, cf. Sec. 4.9.1.3 and Remark 13 on p. 97. This results in a prescribed linear displacement in u_x and u_y along these edges. The inner modes along the curved boundaries are not prescribed. Nevertheless the circular arc is very well represented after the deformation.

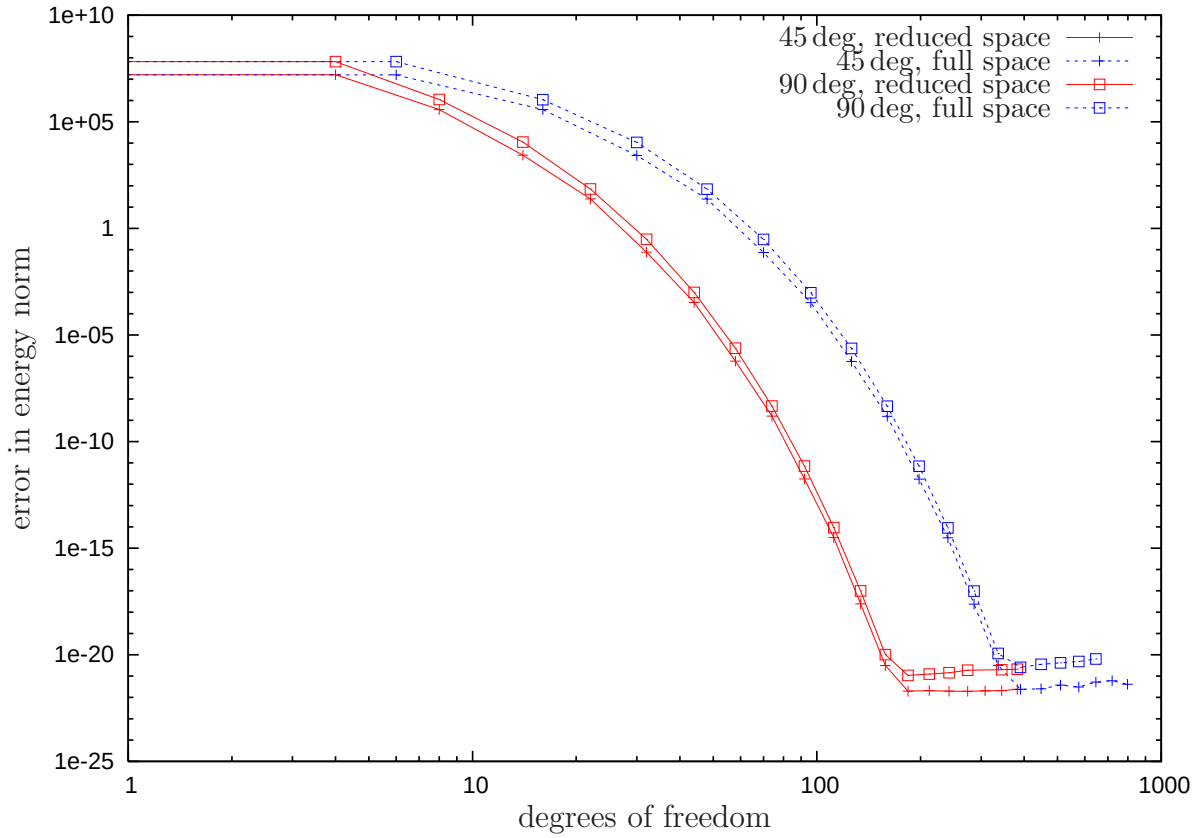


Figure 4.13: Error in energy norm for the rigid body rotation of 45 and 90 degree using the trunk (reduced) and the tensor product (full) space. Each point along the lines represents an increase in polynomial degree by one.

configuration) can be studied in Fig. 4.14. The plot is given for the trunk space however the results for the tensor product space are almost identical.

To show that the error discussed is only introduced by the curved boundary described with trigonometric functions we report the error in strain energy for exactly the same example with small modifications of the mesh in Tab. 4.1. The arcs are replaced by straight lines in one setting and by parabolas in an alternative design. For the straight lines the rigid body rotation induces no strain already for linear shape functions, $p = 1$. For the parabolic boundaries there is a significant error for $p = 1$ that vanishes if $p = 2$ is used as the Ansatz space now contains quadratic functions that exactly describe the rigid body rotation, cf. Eq. (4.44).

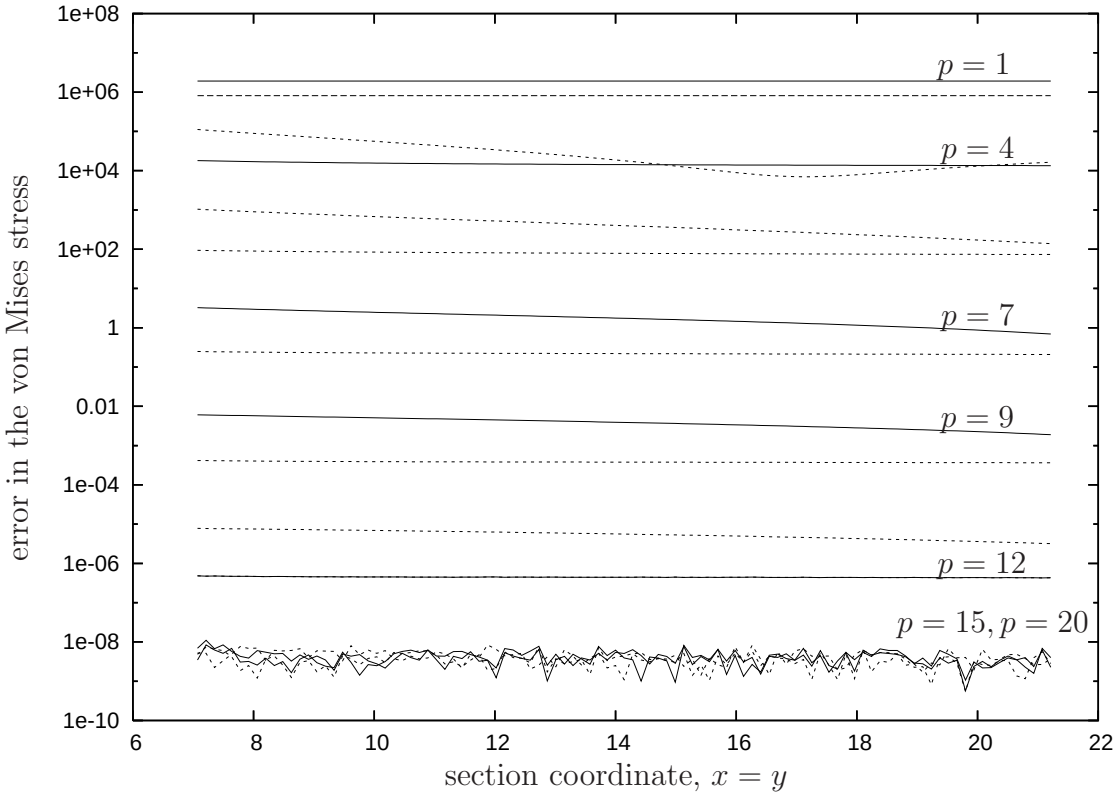


Figure 4.14: Rotation 45 degree: convergence of the absolute error in von Mises stress along a radial cutline ($x = y$) for increasing polynomial degree. Only the solid lines are labeled with their polynomial degree.

p	straight	parabola	arc	meshes
1	2.66E-22	1.28E+09	1.52E+09	
2	2.62E-22	6.60E-22	6.66E+07	
3	2.63E-22	6.60E-22	1.11E+06	
4	2.62E-22	6.67E-22	1.11E+04	
5	2.67E-22	6.65E-22	7.09E+01	
6	2.54E-22	6.60E-22	3.10E-01	
7	2.74E-22	6.99E-22	9.90E-04	
8	2.65E-22	6.74E-22	2.41E-06	
9	2.68E-22	6.71E-22	4.62E-09	
10	3.24E-22	6.75E-22	7.17E-12	

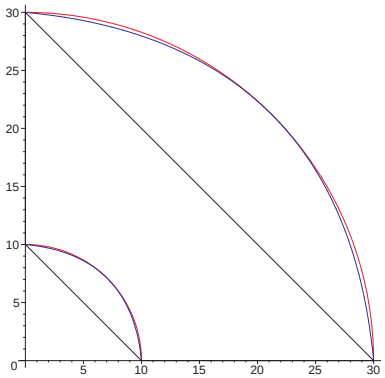
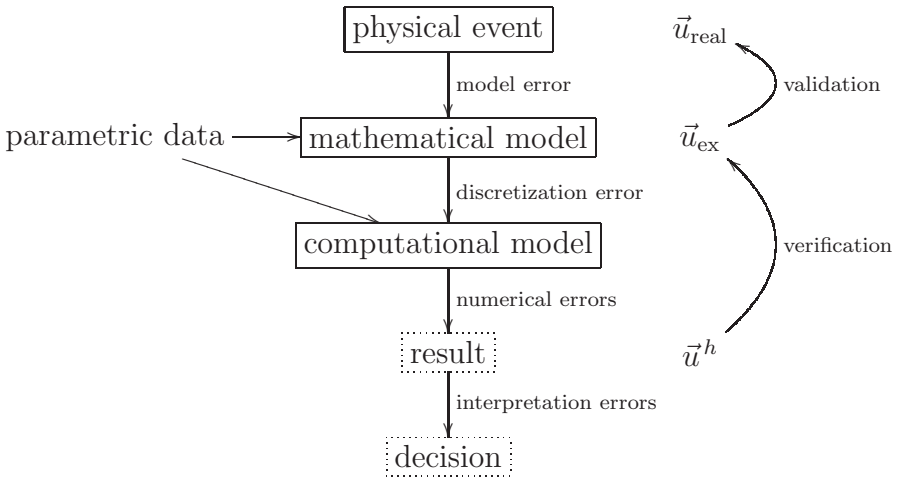


Table 4.1: Rigid-body rotation ($\alpha = 90$ deg.): strain energy of the p-extension (reduced space) for the domain bounded with straight lines, parabolic and circular arcs as shown.

4.4 Error control

The finite element discretization of a physical process approximates a continuous problem with a limited number of parameters. Therefore, the exact solution is only obtained in very special cases. In most cases the numerical analysis of a physical problem is subject to a number of errors and the concept of verification and validation addresses them. The sketch of the different levels involved



(4.44)

gives guidelines how to classify some of the errors.

- **Model errors** occur in the process of describing a physical phenomenon with a numerical model. Often assumptions are inevitable to derive a model that is to be solved by numerical methods. Common assumptions are linear kinematics, reduction of the dimension if e.g. a three-dimensional dam structure is modeled with plain strain assumptions. Frequently static or quasi-static analysis is chosen instead of a dynamic investigation. Also constitutive models and kinematic assumptions belong to the domain of modeling errors.

- **Discretization errors** result from the transition of the continuous mathematical model to a discrete system. Errors of this class can stem from the approximation of the geometry, the finite element mesh, the polynomial degree of the Ansatz functions, the resolution of the temporal discretization and from steep gradients in the solution.

- **Parameter errors** describe the uncertainty of material and geometric parameters that are needed to make a prediction for a specific real-world problem. A slightly different prominent case goes by the name of 'metric mishap' and resulted in a $125 million loss of the Mars Climate Orbiter (MCO) in September 1999 because a Lockhead Martin engineering team used English units while the NASA team used the metric system for spacecraft operation.[7]

[7] 'The MCO MIB has determined that the root cause for the loss of the MCO spacecraft was the failure to use metric units in the coding of a ground software file ... used in trajectory models. Specifically, thruster performance data in English units instead of metric units was used in the software application code ...', (NASA, 1999, p. 16).

- **Numerical errors** introduced by round-off can be controlled in most cases by using double precision arithmetics and applying stable algorithms.

- **Interpretation errors** describe a different type of error introducing humans who make decisions based on computed results. As the size of models grows the data available may become very complex to handle. Even if it is assumed that the model perfectly captured reality its predictions still must be accessed and judged. Visualization of computed results is often vital to understand and evaluate the results of the numerical model. For wrong interpretation based on inaccurate visualization, (Nelson and Kirby, 2006) introduced the notion 'visualization error'.

These sources contribute to the overall error. The error as the difference between real physical data, e.g. $\vec{u}_{\text{real}}$, and the numerical solution can only be assessed a posteriori if experiments are available. The difference $\vec{e}$ of the computational solution $\vec{u}^h$ from the exact solution $\vec{u}_{\text{ex}}$ of the mathematical model can also only be evaluated a posteriori, i.e. in this case after the computation, however it is independent from physical data,

$$\vec{e} = \vec{u}_{\text{ex}} - \vec{u}^h \,. \tag{4.45}$$

The p-version provides mechanisms to control most classes of the errors distinguished above. The hierarchic shape functions with orthogonality properties (Sec. 4.2.1) were chosen in a way to minimize round-off error. Anisotropic plate and shell analyses with a hierarchical Ansatz space allow to enrich the space step by step and thus construct a hierarchy of kinematic models, giving the possibility to assess the model error related to the kinematic. The different versions of the finite element method — $h, p, hp, \ldots$ — that are briefly sketched in Sec. 4.1 focus on the control of the discretization error.

As numerical methods are used to support significant engineering decisions, for example the design of an airplane, the awareness for the different types of errors and an approximation of their magnitude is of great importance. The American Society of Mechanical Engineers has only recently set up a working committee on Verification and Validation in Computational Solid Mechanics that proposed a draft in the year 2006, ASME V&V 10-2006. This is recognized as an important step towards implementing these topics in a national design code. This way of thinking will have major impact on the computational codes as the "currently used large finite element software tools were not designed to support V&V procedures. Major revisions will be necessary."[8]

The two keywords are *verification* and *validation* (V&V). A prominent mathematician working on finite elements, Ivo Babuška, and also one of the fathers of the p-version, considers V&V issues to be the important challenge for the next decade[9] On the other hand he is in line with the eminent philosopher of science, Karl Popper (1902–1994). The Popperian argument (Popper, 1959) is that, unlike mathematical propositions, claims in empirical science can not be proved, they can only be *tested* through observations. As one can never be sure to have

[8] Barna Szabó in a keynote lecture at the High-Order Finite Element Methods conference, May 18, 2007, Herrsching am Ammersee (near Munich), Germany

[9] See, for example his presentations at the MAFELAP 2006 'Will Computational Science Fulfill It's Promises? - Reliability of the Computational Engineering' and at the 303. WE-Heraeus seminar 2003 'V&V in computational engineering and science', as well as the technical reports (Babuška and Oden, 2003), (Babuška et al., 2004)

tested all the possible cases, a scientific theory can never be validated, but only *not falsified*. He proposes the notion of '*corroboration*' or 'proved its mettle' as long as a theory withstands severe tests and it not superseded by another theory. Babuška admits to be tempted to abandon the term 'validation' and use the more precise notion of 'corroboration', as theories in science are never validated. But as the term 'validation' has a long and widespread use in contemporary technical language he opts to continue to use 'validation', and to emphasize that it is subject of a burden: validation *with respect to specific tolerances for specific quantities of interest*. This said he proposes the following concise definitions (Babuška and Oden, 2003)

- *Verification:* The process of determining, whether a computational model obtained by discretizing a mathematical model of a physical event represents the mathematical model with sufficient accuracy.

- *Validation:* The process of determining, whether a mathematical model of a physical event represents the actual physical event with sufficient accuracy.

Validation can be paraphrased as 'are we solving the right model' while verification addresses the issue 'are we solving the mathematical model right'. Verification can be split in code verification and calculation verification. The calculation verification of numerics is equivalent to *a posteriori error estimation* of PDE's, a subject introduced by Babuška and Rheinboldt in 1979 (Babuška and Rheinboldt, 1979). Specific statements about errors can only be derived after the analysis (*a posteriori*). However, general characteristics of the error are also available *a priori* if there is existing knowledge of the structure of the problem (boundary conditions and geometry). For details we refer to (Babuška and Strouboulis, 2001; Schwab, 1998; Szabó and Babuška, 1991). The knowledge of the 'smoothness' of the solution gives guidelines for the selection of an appropriate discretization method and for mesh generation, see e.g. (Babuska and Rank, 1987). The convergence rate, i.e. the rate of reduction of the error, is known a priori for different classes of problems. If, for example, the error (e.g. in energy norm) is plotted versus the degrees of freedom, the p-version is proven to show exponential convergence for smooth problems while the h-version converges only algebraically. For problems with singular points the p-version with an a priori constructed geometrical mesh maintains exponential convergence. For more details see for example (Demkowicz, 2006; Düster, 2001; Szabó and Babuška, 1991)

4.5 A note on temporal discretization

This section briefly comments on some temporal discretization strategies for time-dependent phenomena, the second step in the vertical method of lines, cf. p. 39. In solid mechanics ordinary differential equations (ODEs) occur in the description of transient phenomena like heat conduction, structural dynamics and path-dependent materials described by the evolution of internal variables. One can observe that the numerical treatment of these problems in the finite element world is quite decoupled from recent developments in numerical mathematics. In numerical mathematics evolved methods for integration of ordinary differential equations were developed that slowly begin to spread into the finite element world. The time discretization methods commonly used in the finite element treatment of linear structural dynamics are summarized in Part Two of Hughes's book (Hughes, 2000). Most of them are variants of

Newmark's method, trying to balance stability, the order of the method and the possibility to damp out spurious modes by numerical dissipation. One of the later developments in this line is the generalized-α method (Chung and Hulbert, 1993) that was implemented in the p-version FE-code AdhoC, see (Heisserer, 2001) for large strain structural dynamics. In the context of material models with ordinary differential equations describing the evolution of internal variables, 'modern'[10] methods like the Runge-Kutta family (allowing adaptive step control) in combination with the multi-level Newton algorithm and Rosenbrock methods were investigated by Hartmann and co-workers in (Ellsiepen and Hartmann, 2001; Hartmann, 2002; Hartmann and Wensch, 2007). The backward Euler method used for the examples in this thesis is a special case, a diagonally implicit Runge-Kutta method of first order.

4.6 Discretized linearized variational form

For the spatial discretization of the linearized variational form, Eq. (3.52), we are now working in the FE-space defined by the mesh, the mapping, and the Ansatz functions. This approximation is indicated by the superscript h. To prepare a computer implementation, the column matrix form of the tensorial quantities is derived in the total Lagrangian framework in Sec. 4.6.1 and the subsequent spatial discretization is carried out in Sec. 4.6.2. For an overview of the approximation and discretization stages and corresponding notation compare Tab. 1 and Tab. 2.

4.6.1 Column matrix representation of the tensorial quantities

4.6.1.1 Strain tensor and related quantities

To obtain a column matrix of the principle of virtual work using Voigt notation (cf. footnote 11 on p. 61) we start with the Green-Lagrange strain tensor $\mathbf{E}$ (2.13) and its variation $\delta\mathbf{E}$ (3.7). Exploiting the symmetry $\mathbf{E} = \mathbf{E}^T$ we have n_{st} individual components that can be stored in a column matrix $\boldsymbol{E}^h(\boldsymbol{X}, t) \in \mathbb{R}^{n_{\mathrm{st}}}$. Here n_{st}, the number of strains/stresses is six in the three-dimensional (3D) case, four for axisymmetric (AXS) problems and three for plain strain computations[11],

$$\text{3D: } \boldsymbol{E}^h = \begin{Bmatrix} E^h_{11} \\ E^h_{22} \\ E^h_{33} \\ 2E^h_{12} \\ 2E^h_{23} \\ 2E^h_{31} \end{Bmatrix} = \begin{Bmatrix} u_{1,1} \\ u_{2,2} \\ u_{3,3} \\ u_{1,2} + u_{2,1} \\ u_{2,3} + u_{3,2} \\ u_{1,3} + u_{3,1} \end{Bmatrix} + \begin{Bmatrix} \frac{1}{2}\left(u^2_{1,1} + u^2_{2,1} + u^2_{3,1}\right) \\ \frac{1}{2}\left(u^2_{1,2} + u^2_{2,2} + u^2_{3,2}\right) \\ \frac{1}{2}\left(u^2_{1,3} + u^2_{2,3} + u^2_{3,3}\right) \\ u_{1,1}u_{1,2} + u_{2,1}u_{2,2} + u_{3,1}u_{3,2} \\ u_{1,2}u_{1,3} + u_{2,2}u_{2,3} + u_{3,2}u_{3,3} \\ u_{1,1}u_{1,3} + u_{2,1}u_{2,3} + u_{3,1}u_{3,3} \end{Bmatrix}, \tag{4.46}$$

[10] Well, 'modern' is in the context of the finite element method indeed very old. Runge and Kutta developed their method around 1900, see (Runge, 1895) and (Kutta, 1901).

[11] The factor two in the column matrix representation accompanying the off-diagonal terms (e.g. $2E^h_{12}$) stems from the fact that the contraction of the stress and strain tensor $\delta\mathbf{E} \cdot \tilde{\mathbf{T}}$ in the principle of virtual work is to be described by the scalar product of the vectors $\delta\boldsymbol{E}^h$ and $\tilde{\boldsymbol{T}}^h$, so one of the column vectors must have the factor two for the off-diagonal entries. For the details of the so-called Voigt notation, compare (Hartmann, 2003a, Appendix 2) or (Belytschko et al., 2000, Appendix 1) and (Wriggers, 2001, p. 121). Note, however that Belytschko uses a different arrangement of the off-diagonal entries.

$$\text{AXS: } \boldsymbol{E}^h = \begin{Bmatrix} E^h_{11} \\ E^h_{22} \\ E^h_{33} \\ 2E^h_{12} \end{Bmatrix} = \begin{Bmatrix} u_{1,1} \\ u_{2,2} \\ \dfrac{u_1}{X_1} \\ u_{1,2}+u_{2,1} \end{Bmatrix} + \begin{Bmatrix} \frac{1}{2}(u^2_{1,1}+u^2_{2,1}) \\ \frac{1}{2}(u^2_{1,2}+u^2_{2,2}) \\ \dfrac{1}{2}\left(\dfrac{u_1}{X_1}\right)^2 \\ u_{1,1}u_{1,2}+u_{2,1}u_{2,2} \end{Bmatrix}, \tag{4.47}$$

$$\text{2D: } \boldsymbol{E}^h = \begin{Bmatrix} E^h_{11} \\ E^h_{22} \\ 2E^h_{12} \end{Bmatrix} = \begin{Bmatrix} u_{1,1} \\ u_{2,2} \\ u_{1,2}+u_{2,1} \end{Bmatrix} + \begin{Bmatrix} \frac{1}{2}(u^2_{1,1}+u^2_{2,1}) \\ \frac{1}{2}(u^2_{1,2}+u^2_{2,2}) \\ u_{1,1}u_{1,2}+u_{2,1}u_{2,2} \end{Bmatrix}, \tag{4.48}$$

where u_i are the components of $\boldsymbol{u}^h$ and the comma notation denotes the derivative with respect to material coordinates, e.g. $u_{1,2} = \frac{\partial u_1}{\partial X_2}$. In the axisymmetric case the denominator X_1 in the fraction designates the radial distance from the axis of rotation.

Following Crisfield (Crisfield, 1991, Ch. 5) we denote by $\boldsymbol{\Theta}^h$ the column vector 'picking' the $n_{\rm gr}$ entries of the material displacement gradient $\mathbf{G} = \operatorname{Grad} \vec{u}^h$, Eq. (3.8), in a specific order. $\operatorname{Grad} \vec{u}^h$ has nine entries in the three-dimensional case, five for axisymmetry and four in the 2D case. In three dimensions we have $\boldsymbol{\Theta}^h(\boldsymbol{u}^h) = \{u_{1,1}, u_{1,2}, u_{1,3}, u_{2,1}, u_{2,2}, u_{2,3}, u_{3,1}, u_{3,2}, u_{3,3}\}^T$, while in the other cases the representations are

$$\text{axisymmetric, } \boldsymbol{\Theta}^h(\boldsymbol{u}^h) = \begin{Bmatrix} u_{1,1} \\ u_{1,2} \\ u_{2,1} \\ u_{2,2} \\ \dfrac{u_1}{X_1} \end{Bmatrix}, \quad \text{and 2D, } \boldsymbol{\Theta}^h(\boldsymbol{u}^h) = \begin{Bmatrix} u_{1,1} \\ u_{1,2} \\ u_{2,1} \\ u_{2,2} \end{Bmatrix}. \tag{4.49}$$

Due to the linear property of 'picking', the corresponding virtual displacement gradient vectors $\delta\boldsymbol{\Theta}^h$ are defined accordingly, e.g. in the 2D case

$$\delta\boldsymbol{\Theta}^h(\boldsymbol{u}^h) = \boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h) = \{\delta u_{1,1}, \delta u_{1,2}, \delta u_{2,1}, \delta u_{2,2}\}^T. \tag{4.50}$$

Then, the strain column matrix can be expressed as

$$\begin{aligned} \boldsymbol{E}^h(\boldsymbol{X},t) = \boldsymbol{E}^h_\ell + \boldsymbol{E}^h_{n\ell} &= \boldsymbol{H}\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t)) + \frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t))\right)\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t)) \\ &= \left[\boldsymbol{H} + \frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t))\right)\right]\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t)) \end{aligned} \tag{4.51}$$

using for the three cases the incidence matrix $\boldsymbol{H} \in \mathbb{R}^{n_{\rm st}\times n_{\rm gr}}$ and a matrix $\boldsymbol{M}_\Theta \in \mathbb{R}^{n_{\rm st}\times n_{\rm gr}}$ con-

taining the elements of the gradient $\boldsymbol{\Theta}^h$ in a special arrangement

3D (4.52)

$$\boldsymbol{H} = \begin{bmatrix} 1 & . & . & . & . & . & . & . & . \\ . & . & . & . & 1 & . & . & . & . \\ . & . & . & . & . & . & . & . & 1 \\ . & 1 & . & 1 & . & . & . & . & . \\ . & . & . & . & . & 1 & . & 1 & . \\ . & . & 1 & . & . & . & 1 & . & . \end{bmatrix}, \quad \boldsymbol{M}_\Theta = \begin{bmatrix} u_{1,1} & . & . & u_{2,1} & . & . & u_{3,1} & . & . \\ . & u_{1,2} & . & . & u_{2,2} & . & . & u_{3,2} & . \\ . & . & u_{1,3} & . & . & u_{2,3} & . & . & u_{3,3} \\ u_{1,2} & u_{1,1} & . & u_{2,2} & u_{2,1} & . & u_{3,2} & u_{3,1} & . \\ . & u_{1,3} & u_{1,2} & . & u_{2,3} & u_{2,2} & . & u_{3,3} & u_{3,2} \\ u_{1,3} & . & u_{1,1} & u_{2,3} & . & u_{2,1} & u_{3,3} & . & u_{3,1} \end{bmatrix}$$

AXS (4.53)

$$\boldsymbol{H} = \begin{bmatrix} 1 & . & . & . & . \\ . & . & . & 1 & . \\ . & . & . & . & 1 \\ . & 1 & 1 & . & . \end{bmatrix}, \quad \boldsymbol{M}_\Theta = \begin{bmatrix} u_{1,1} & . & u_{2,1} & . & . \\ . & u_{1,2} & . & u_{2,2} & . \\ . & . & . & . & \frac{u_1}{X_1} \\ u_{1,2} & u_{1,1} & u_{2,2} & u_{2,1} & . \end{bmatrix}$$

2D (4.54)

$$\boldsymbol{H} = \begin{bmatrix} 1 & . & . & . \\ . & . & . & 1 \\ . & 1 & 1 & . \end{bmatrix}, \quad \boldsymbol{M}_\Theta = \begin{bmatrix} u_{1,1} & . & u_{2,1} & . \\ . & u_{1,2} & . & u_{2,2} \\ u_{1,2} & u_{1,1} & u_{2,2} & u_{2,1} \end{bmatrix},$$

where dots '.' denote zero entries. The correctness can be proved by element-wise computation.

The vector form of the virtual strain tensor $\delta\mathbf{E}$ (3.7) is obtained by applying the definition of the variation (3.26) as directional derivative, see (3.21), in the direction of $\delta\boldsymbol{u}^h$ to the vector form of the strains $\boldsymbol{E}^h$ (4.51),

$$\begin{aligned} \delta\boldsymbol{E}^h(\boldsymbol{u}^h, \delta\boldsymbol{u}^h) &= \mathsf{D}_{\boldsymbol{u}^h}\boldsymbol{E}^h(\boldsymbol{u}^h)[\delta\boldsymbol{u}^h] \qquad (4.55)\\ &= \frac{\mathrm{d}}{\mathrm{d}\epsilon}\left[\boldsymbol{H}\boldsymbol{\Theta}^h(\boldsymbol{u}^h + \epsilon\delta\boldsymbol{u}^h) + \frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h + \epsilon\delta\boldsymbol{u}^h)\right)\boldsymbol{\Theta}^h(\boldsymbol{u}^h + \epsilon\delta\boldsymbol{u}^h)\right]_{\epsilon=0} \\ &= \boldsymbol{H}\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h) \\ &+ \left[\frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h)\right)\boldsymbol{\Theta}^h(\boldsymbol{u}^h + \epsilon\delta\boldsymbol{u}^h) + \frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h + \epsilon\delta\boldsymbol{u}^h)\right)\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h)\right]_{\epsilon=0} \\ &= \boldsymbol{H}\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h) + \frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h)\right)\boldsymbol{\Theta}^h(\boldsymbol{u}^h) + \frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h)\right)\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h). \end{aligned}$$

As the matrix $\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h)\right)$, e.g. Eq. (4.52), is just a rule how to pick the entries of the displacement gradient we see the equivalence of

$$\frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h)\right)\boldsymbol{\Theta}^h(\boldsymbol{u}^h) = \frac{1}{2}\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h)\right)\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h)\,. \qquad (4.56)$$

Consequently we can simplify Eq. (4.55) to obtain the column matrix representation of the virtual strain tensor

$$\delta\boldsymbol{E}^h(\boldsymbol{u}^h(t), \delta\boldsymbol{u}^h) = \left[\boldsymbol{H} + \boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t))\right)\right]\boldsymbol{\Theta}^h(\delta\boldsymbol{u}^h(t)). \qquad (4.57)$$

As the definition of the variation, Eq. (3.26), is based on the directional derivative we can immediately evaluate the directional derivative of the strain column matrix

$$\Delta\boldsymbol{E}^h = \mathsf{D}_{\boldsymbol{u}^h}\boldsymbol{E}^h(\boldsymbol{u}^h(t))[\Delta\boldsymbol{u}^h] = \left[\boldsymbol{H} + \boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t))\right)\right]\boldsymbol{\Theta}^h(\Delta\boldsymbol{u}^h(t)) \qquad (4.58)$$

if we replace the 'direction' $\delta\boldsymbol{u}^h$ by $\Delta\boldsymbol{u}^h$.

Remark 6 *The factor $\frac{1}{2}$ that is present in the vector form of the Green-Lagrange strain tensor $\boldsymbol{E}^h$ (4.51) is missing in its variation $\delta\boldsymbol{E}^h$ (4.57) and directional derivative $\Delta\boldsymbol{E}^h$ (4.58) as the product rule together with the equivalence relation (4.56) is applied. The strain tensor $\mathbf{E}$ (corresponding to the factor $\frac{1}{2}$) does not occur in the terms of the (linearized) virtual work (3.52), only its variation or the directional derivative. However, it might be for postprocessing or other reasons desirable to compute the spatially discrete Green-Lagrange strain tensor $\boldsymbol{E}$. Hence, one can implement a generic $\boldsymbol{B}(\beta)$ matrix, cf. Eq. (4.81), governed by the factor β that evaluates the operator $[\boldsymbol{H} + \beta\boldsymbol{M}_\Theta]\,\boldsymbol{G}$. If the variation or the directional derivative is needed, set $\beta = 1$, for the strains we choose $\beta = \frac{1}{2}$. Another application of the $\beta = \frac{1}{2}$ case is the case of a linear material model where the stress tensor $\tilde{\mathbf{T}}$ in the virtual internal work (3.53) corresponding to the known state $\vec{u}$ can be computed by multiplication of a material matrix $\tilde{\boldsymbol{C}}_L$ and the discretized strains $\boldsymbol{E}$. An example is an extension of Hooke's law to large strains, the so called St.Venant-Kirchhoff model where we would substitute $\tilde{\boldsymbol{T}}^h = \tilde{\boldsymbol{C}}_L\boldsymbol{E}^h$ in the discretized form of Eq. (3.53).*

4.6.1.2 Stress tensor and related quantities

The symmetric second Piola-Kirchhoff stress tensor $\tilde{\mathbf{T}} = \tilde{\mathbf{T}}^T$ is transformed to a column matrix $\tilde{\boldsymbol{T}}^h(\boldsymbol{X}, t) \in \mathbb{R}^{n_{\mathrm{st}}}$

$$\text{3D:} \quad \tilde{\boldsymbol{T}}^h = \{\tilde{T}^h_{11}, \tilde{T}^h_{22}, \tilde{T}^h_{33}, \tilde{T}^h_{12}, \tilde{T}^h_{23}, \tilde{T}^h_{13}\}^T \tag{4.59}$$

$$\text{AXS:} \quad \tilde{\boldsymbol{T}}^h = \{\tilde{T}^h_{11}, \tilde{T}^h_{22}, \tilde{T}^h_{33}, \tilde{T}^h_{12}\}^T \tag{4.60}$$

$$\text{2D:} \quad \tilde{\boldsymbol{T}}^h = \{\tilde{T}^h_{11}, \tilde{T}^h_{22}, \tilde{T}^h_{12}\}^T \tag{4.61}$$

and is governed by a constitutive relation of the type

$$\tilde{\boldsymbol{T}}^h(\boldsymbol{X}, t) = \tilde{\boldsymbol{\Phi}}^h\Big(\boldsymbol{C}^h\big(\boldsymbol{E}^h(\boldsymbol{X}, t)\big), \boldsymbol{q}^h\big(\boldsymbol{C}^h(\boldsymbol{E}^h(\boldsymbol{X}, t))\big)\Big), \tag{4.62}$$

the vector form of Eq. (2.43), where $\boldsymbol{C}^h$ is the spatially discretized right Cauchy-Green tensor

$$\boldsymbol{C}^h = 2\boldsymbol{E}^h + \boldsymbol{I} \tag{4.63}$$

and the internal variables $\boldsymbol{q}^h$ are given by the initial value problem,

$$\boldsymbol{A}\dot{\boldsymbol{q}}^h(\boldsymbol{X}, t) - \boldsymbol{r}(\boldsymbol{E}^h(\boldsymbol{X}, t), \boldsymbol{q}^h(\boldsymbol{X}, t)) = \boldsymbol{0}, \quad \boldsymbol{q}^h(\boldsymbol{X}, t_0) = \boldsymbol{q}^h_o(\boldsymbol{X}), \tag{4.64}$$

the matrix form of Eq. (2.47).

Analogously to the tensorial consistent tangent operator $\tilde{\mathcal{C}}$ (3.34), the matrix form $\tilde{\boldsymbol{C}}^h_L$ is derived evoking the Gâteaux derivative of the constitutive equation Eq. (4.62) in the direction of the strain increment $\Delta\boldsymbol{E}^h$ (4.58),

$$\mathsf{D}_{\boldsymbol{E}^h}\,\tilde{\boldsymbol{\Phi}}^h(\boldsymbol{E}^h)[\Delta\boldsymbol{E}^h] = \left[\frac{\partial\tilde{\boldsymbol{\Phi}}^h}{\partial\boldsymbol{C}^h} + \frac{\partial\tilde{\boldsymbol{\Phi}}^h}{\partial\boldsymbol{q}^h}\cdot\frac{\mathrm{d}\boldsymbol{q}^h}{\mathrm{d}\boldsymbol{C}^h}\right] 2\,\Delta\boldsymbol{E}^h = \tilde{\boldsymbol{C}}^h_L(\boldsymbol{X}, t)\;\Delta\boldsymbol{E}^h\,. \tag{4.65}$$

The factor 2 in $\tilde{\boldsymbol{C}}_L^h$ stems from the fact that we want to express the right Cauchy-Green tensor depending on the Green strain tensor as $\boldsymbol{C}^h = 2\boldsymbol{E}^h + \boldsymbol{I}$. Hence the directional derivative $\mathsf{D}_{\boldsymbol{E}^h}\boldsymbol{C}^h(\boldsymbol{E}^h)[\Delta\boldsymbol{E}^h] = 2\Delta\boldsymbol{E}^h$ has the factor 2. Now we can express the increment in stress needed in Eq. (3.28) in column matrix representation as

$$\mathsf{D}_{\boldsymbol{u}^h}\tilde{\boldsymbol{T}}^h(\boldsymbol{X},t)[\Delta\boldsymbol{u}^h] = \tilde{\boldsymbol{C}}_L^h\Delta\boldsymbol{E}^h\,. \tag{4.66}$$

4.6.2 Spatial discretization of the linearized weak form

We use the standard Bubnov-Galerkin[12] approach to approximate the continuous vector fields of displacement $\vec{u}(\vec{X},t)$ and virtual displacement $\delta\vec{u}(\vec{X},t)$ by the same Ansatz functions N_j, compare Eq. (4.2),

$$\begin{aligned}
\boldsymbol{u}^h(\boldsymbol{X},t) &= \boldsymbol{N}_a(\boldsymbol{X})\,\boldsymbol{u}_a(t), \quad \boldsymbol{u}^h \in \mathbb{R}^{n_{\mathrm{sd}}} && (4.67)\\
\delta\boldsymbol{u}^h(\boldsymbol{X}) &= \boldsymbol{N}_a(\boldsymbol{X})\,\delta\boldsymbol{u}_a, \quad \delta\boldsymbol{u}^h \in \mathbb{R}^{n_{\mathrm{sd}}} && (4.68)\\
\Delta\boldsymbol{u}^h(\boldsymbol{X}) &= \boldsymbol{N}_a(\boldsymbol{X})\,\Delta\boldsymbol{u}_a, \quad \Delta\boldsymbol{u}^h \in \mathbb{R}^{n_{\mathrm{sd}}} && (4.69)
\end{aligned}$$

with the global vectors collecting the displacement coefficients $\boldsymbol{u}_a \in \mathbb{R}^{n_{\mathrm{dof}}}$, virtual displacement coefficients[13] $\delta\boldsymbol{u}_a \in \mathbb{R}^{n_{\mathrm{dof}}}$ and the increment vector $\Delta\boldsymbol{u}_a \in \mathbb{R}^{n_{\mathrm{dof}}}$. The corresponding shape functions $N_j(\boldsymbol{X}) \in \mathbb{R}$ introduced in Sec. 4.2.2 are collected in the matrix $\boldsymbol{N}_a(\boldsymbol{X}) \in \mathbb{R}^{n_{\mathrm{sd}}\times n_{\mathrm{dof}}}$, $\boldsymbol{X} \in \mathbb{R}^{n_{\mathrm{sd}}}$ represents the column vector of Cartesian coordinates in reference configuration of point $\vec{X}$.

Remark 7 *Acknowledging the history of the finite element method where the Lagrangian shape functions were dominating in the beginning when the notations were coined, we speak of the* 'displacement' *vector* $\boldsymbol{u}_a$*. However, for a general basis the unknowns* $\boldsymbol{u}_a$ *are just coefficients of the basis functions. Two different systems of high order bases are compared in Sec. 4.2. Using a Lagrangian (or standard) basis the solution coefficients of the shape functions are indeed the displacements at these 'nodes' due to property Eq. (4.6) that states that all shape functions but one vanish at a node and the non-vanishing function has the value* 1 *there. But using a hierarchical basis of integrated Legendre polynomials as we do, only the coefficients belonging to the nodal modes, Eq. (4.23), have the direct meaning of displacements and all the other values are just scaling factors of the higher order modes. Therefore, we speak of '***m***odal shape functions' instead of '***n***odal shape functions' and all summations are over the number of modes.*

Following (Hartmann, 2003a) we distinguish known (prescribed) and unknown displacement degrees of freedom. This prepares for the computation of reaction forces as derived in Sec. 4.9. All (mnemonic subscript **a**) displacement coefficients ('degrees of freedom') are collected in column matrices, the displacement coefficients in $\boldsymbol{u}_a(t) \in \mathbb{R}^{n_{\mathrm{dof}}}$, their virtual counterparts in $\delta\boldsymbol{u}_a \in \mathbb{R}^{n_{\mathrm{dof}}}$. The shape functions are collected in the matrix $\boldsymbol{N}_a(\boldsymbol{X}) \in \mathbb{R}^{n_{\mathrm{sd}}\times n_{\mathrm{dof}}}$. For partitioning we denote by $\boldsymbol{u}(t) \in \mathbb{R}^{n_{\mathrm{u}}}$ the n_{u} **u**nknown degrees of freedom, while $\overline{\boldsymbol{u}}(t) \in \mathbb{R}^{n_{\mathrm{p}}}$

[12] Contrary to the Petrov-Galerkin approach where test ($\delta\vec{u}$) and trial ($\vec{u}$) functions differ, cf. (Belytschko et al., 2000, p. 406)

[13] The time-dependence of the virtual displacement is neglected. See the discussion on page 26.

represents the vector of the n_p **p**rescribed displacements, $n_\mathrm{dof} = n_\mathrm{u} + n_\mathrm{p}$. As the virtual displacements have to vanish by construction where Dirichlet boundary conditions are prescribed, we have $\delta\overline{\boldsymbol{u}} = \boldsymbol{0} \in \mathbb{R}^{n_\mathrm{p}}$, while the remaining virtual displacements $\delta\boldsymbol{u} \in \mathbb{R}^{n_\mathrm{u}}$ are arbitrary.

Consequently we can write

$$\boldsymbol{u}_a(t) = \left\{ \begin{array}{c} \boldsymbol{u}(t) \\ \overline{\boldsymbol{u}}(t) \end{array} \right\}, \quad \Delta\boldsymbol{u}_a = \left\{ \begin{array}{c} \Delta\boldsymbol{u} \\ \Delta\overline{\boldsymbol{u}} \end{array} \right\} \quad \text{and} \quad \delta\boldsymbol{u}_a = \left\{ \begin{array}{c} \delta\boldsymbol{u} \\ \delta\overline{\boldsymbol{u}} \end{array} \right\} = \left\{ \begin{array}{c} \delta\boldsymbol{u} \\ \boldsymbol{0} \end{array} \right\} \tag{4.70}$$

and introducing this partitioning in Eqs. (4.67) – (4.68) we divide the matrix $\boldsymbol{N}_a$ accordingly to obtain

$$\boldsymbol{u}^h(\boldsymbol{X},t) = \big[\boldsymbol{N}(\boldsymbol{X})\ \overline{\boldsymbol{N}}(\boldsymbol{X})\big] \left\{ \begin{array}{c} \boldsymbol{u}(t) \\ \overline{\boldsymbol{u}}(t) \end{array} \right\} = \boldsymbol{N}(\boldsymbol{X})\boldsymbol{u}(t) + \overline{\boldsymbol{N}}(\boldsymbol{X})\overline{\boldsymbol{u}}(t) \tag{4.71}$$

$$\Delta\boldsymbol{u}^h(\boldsymbol{X},t) = \big[\boldsymbol{N}(\boldsymbol{X})\ \overline{\boldsymbol{N}}(\boldsymbol{X})\big] \left\{ \begin{array}{c} \Delta\boldsymbol{u} \\ \Delta\overline{\boldsymbol{u}} \end{array} \right\} = \boldsymbol{N}(\boldsymbol{X})\Delta\boldsymbol{u} + \overline{\boldsymbol{N}}(\boldsymbol{X})\Delta\overline{\boldsymbol{u}}\,. \tag{4.72}$$

$$\delta\boldsymbol{u}^h(\boldsymbol{X},t) = \big[\boldsymbol{N}(\boldsymbol{X})\ \overline{\boldsymbol{N}}(\boldsymbol{X})\big] \left\{ \begin{array}{c} \delta\boldsymbol{u} \\ \boldsymbol{0} \end{array} \right\} = \boldsymbol{N}(\boldsymbol{X})\delta\boldsymbol{u}\,. \tag{4.73}$$

Spatial discretization of strain-like quantities

Substituting the Ansatz (4.67) and (4.68) in the definition of the gradient vectors $\boldsymbol{\Theta}^h$ (4.49) and $\delta\boldsymbol{\Theta}^h$ (4.50) we obtain the discretized column vector of the displacement gradient $\boldsymbol{\Theta}(\boldsymbol{u}_a)$ (note that the superscript $(\bullet)^h$ is omitted for discretized quantities)

$$\boldsymbol{\Theta}(\boldsymbol{u}_a(t)) = \boldsymbol{G}\,\boldsymbol{u}_a(t), \quad \delta\boldsymbol{\Theta}(\boldsymbol{u}_a(t)) = \boldsymbol{G}\,\delta\boldsymbol{u}_a, \quad \text{and} \quad \Delta\boldsymbol{\Theta}(\boldsymbol{u}_a(t)) = \boldsymbol{G}\,\Delta\boldsymbol{u}_a \tag{4.74}$$

with

$$\boldsymbol{G} = [\boldsymbol{G}_1, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_{n_\mathrm{modes}}] \tag{4.75}$$

where one of the block matrices $\boldsymbol{G}_i \in \mathbb{R}^{n_\mathrm{gr} \times n_\mathrm{sd}}, i = 1, \ldots, n_\mathrm{modes}$ in $\boldsymbol{G} \in \mathbb{R}^{n_\mathrm{gr} \times n_\mathrm{dof}}$ has the following structure

$$\text{3D:}\boldsymbol{G}_i = \begin{bmatrix} N_{i,1} & . & . \\ N_{i,2} & . & . \\ N_{i,3} & . & . \\ . & N_{i,1} & . \\ . & N_{i,2} & . \\ . & N_{i,3} & . \\ . & . & N_{i,1} \\ . & . & N_{i,2} \\ . & . & N_{i,3} \end{bmatrix} \quad \text{AXS:}\boldsymbol{G}_i = \begin{bmatrix} N_{i,1} & . \\ N_{i,2} & . \\ . & N_{i,1} \\ . & N_{i,2} \\ \frac{N_i}{X_1} & . \end{bmatrix} \quad \text{2D:}\boldsymbol{G}_i = \begin{bmatrix} N_{i,1} & . \\ N_{i,2} & . \\ . & N_{i,1} \\ . & N_{i,2} \end{bmatrix}. \tag{4.76}$$

The derivatives, $N_{i,k} = \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial X_k}$, are with respect to global material coordinates, however the shape functions are given on the standard element $\Omega^\mathrm{q}_\mathrm{st}$ in local coordinates $\boldsymbol{\xi} = \{\xi_1, \xi_2, \xi_3\}^T$. Therefore, we apply the chain rule, viz

$$\frac{\partial N_i\big(\boldsymbol{\xi}(\boldsymbol{X})\big)}{\partial X_j} = \frac{\partial N_i}{\partial \xi_1}\frac{\partial \xi_1}{\partial X_j} + \frac{\partial N_i}{\partial \xi_2}\frac{\partial \xi_2}{\partial X_j} + \frac{\partial N_i}{\partial \xi_3}\frac{\partial \xi_3}{\partial X_j} \qquad \text{for } j = 1, 2, 3\,. \tag{4.77}$$

Collected in matrix/vector form we have in three dimensions

$$\left\{\begin{array}{c} \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial X_1} \\ \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial X_2} \\ \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial X_3} \end{array}\right\} = \left[\begin{array}{ccc} \dfrac{\partial \xi_1}{\partial X_1} & \dfrac{\partial \xi_2}{\partial X_1} & \dfrac{\partial \xi_3}{\partial X_1} \\ \dfrac{\partial \xi_1}{\partial X_2} & \dfrac{\partial \xi_2}{\partial X_2} & \dfrac{\partial \xi_3}{\partial X_2} \\ \dfrac{\partial \xi_1}{\partial X_3} & \dfrac{\partial \xi_2}{\partial X_3} & \dfrac{\partial \xi_3}{\partial X_3} \end{array}\right] \left\{\begin{array}{c} \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial \xi_1} \\ \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial \xi_2} \\ \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial \xi_3} \end{array}\right\} = \boldsymbol{J}^{-T} \left\{\begin{array}{c} \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial \xi_1} \\ \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial \xi_2} \\ \dfrac{\partial N_i(\boldsymbol{\xi})}{\partial \xi_3} \end{array}\right\} \tag{4.78}$$

with the Jacobian matrix $\mathbf{J}(\boldsymbol{\xi})$ (4.36) of the mapping $\boldsymbol{\phi}$, (Eq. (4.30)),

$$\boldsymbol{J} = \frac{\partial \boldsymbol{\phi}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \frac{\partial \boldsymbol{X}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \left[\begin{array}{ccc} \dfrac{\partial X_1}{\partial \xi_1} & \dfrac{\partial X_2}{\partial \xi_1} & \dfrac{\partial X_3}{\partial \xi_1} \\ \dfrac{\partial X_1}{\partial \xi_2} & \dfrac{\partial X_2}{\partial \xi_2} & \dfrac{\partial X_3}{\partial \xi_2} \\ \dfrac{\partial X_1}{\partial \xi_3} & \dfrac{\partial X_2}{\partial \xi_3} & \dfrac{\partial X_3}{\partial \xi_3} \end{array}\right]. \tag{4.79}$$

To spatially discretize the (continuous) column strain matrices $\boldsymbol{E}^h$ (4.51), $\delta\boldsymbol{E}^h$ (4.57) and $\Delta\boldsymbol{E}^h$ (4.58) we start from a generic form $\square\boldsymbol{E}^h$ with placeholders $\square$ and β

$$\square\boldsymbol{E}^h = \left[\boldsymbol{H} + \beta\boldsymbol{M}_\Theta\left(\boldsymbol{\Theta}^h(\boldsymbol{u}^h(t))\right)\right]\boldsymbol{\Theta}^h(\square\boldsymbol{u}^h(t)) \quad \text{for} \left\{\begin{array}{rlll} \boldsymbol{E}^h & \text{set} & \square = & 1, \quad \beta = \frac{1}{2} \\ \delta\boldsymbol{E}^h & \text{set} & \square = & \delta, \quad \beta = 1 \\ \Delta\boldsymbol{E}^h & \text{set} & \square = & \Delta, \quad \beta = 1 \end{array}\right. \tag{4.80}$$

and discretize $\square\boldsymbol{\Theta}^h$ according to (4.74) with the help of the gradient-displacement matrix $\boldsymbol{G}$ (4.75) to arrive at

$$\square\boldsymbol{E} = \underbrace{\left[\boldsymbol{H} + \beta\boldsymbol{M}_\Theta\left(\boldsymbol{G}\boldsymbol{u}_a(t)\right)\right]\boldsymbol{G}}_{\widehat{\boldsymbol{B}}(\beta)}\square\boldsymbol{u}_a(t) \quad \text{for} \left\{\begin{array}{rlll} \boldsymbol{E} & \text{set} & \square = & 1, \quad \beta = \frac{1}{2} \\ \delta\boldsymbol{E} & \text{set} & \square = & \delta, \quad \beta = 1 \\ \Delta\boldsymbol{E} & \text{set} & \square = & \Delta, \quad \beta = 1 \end{array}\right., \tag{4.81}$$

the small strain case is recovered if $\beta = 0$. Splitting linear and nonlinear parts as in (4.51) we can introduce strain-displacement matrices $\boldsymbol{B} \in \mathbb{R}^{n_{\text{st}} \times n_{\text{dof}}}$ as

$$\begin{array}{rcll} \square\boldsymbol{E} & = & \boldsymbol{E}_\ell + \square\boldsymbol{E}_{n\ell} & \\ \boldsymbol{E}_\ell & = & \boldsymbol{B}_\ell\,\boldsymbol{u}_a(t) & = \boldsymbol{H}\boldsymbol{G}\,\boldsymbol{u}_a(t) \\ \square\boldsymbol{E}_{n\ell} & = & \beta\boldsymbol{B}_{n\ell}\,\square\boldsymbol{u}_a(t) & = \left(\beta\boldsymbol{M}_\Theta\left(\boldsymbol{G}(\boldsymbol{u}_a(t))\right)\boldsymbol{G}\right)\square\boldsymbol{u}_a(t)\,. \end{array} \tag{4.82}$$

We define the abbreviations for the individual matrices

$$\boldsymbol{B}_\ell(\boldsymbol{X}) = \quad \boldsymbol{H}\boldsymbol{G} \quad = [\boldsymbol{B}_{\ell\,1}, \boldsymbol{B}_{\ell\,2}, \ldots, \boldsymbol{B}_{\ell\,n_{\text{modes}}}] \tag{4.83}$$

$$\boldsymbol{B}_{n\ell}(\boldsymbol{u}_a(t), \boldsymbol{X}) = \boldsymbol{M}_\Theta\left(\boldsymbol{G}\boldsymbol{u}_a(t)\right)\boldsymbol{G} = [\boldsymbol{B}_{n\ell\,1}, \boldsymbol{B}_{n\ell\,2}, \ldots, \boldsymbol{B}_{n\ell\,n_{\text{modes}}}] \tag{4.84}$$

and the combinations differing by the factor β

$$\boldsymbol{B}^*_a(\boldsymbol{u}_a(t), \boldsymbol{X}) = [\boldsymbol{B}^*_1, \ldots, \boldsymbol{B}^*_{n_{\text{modes}}}] = \boldsymbol{B}_\ell + \frac{1}{2}\boldsymbol{B}_{n\ell} \tag{4.85}$$

$$\boldsymbol{B}_a(\boldsymbol{u}_a(t), \boldsymbol{X}) = [\boldsymbol{B}_1, \ldots, \boldsymbol{B}_{n_{\text{modes}}}] = \boldsymbol{B}_\ell + \boldsymbol{B}_{n\ell}\,. \tag{4.86}$$

The component-wise evaluation of (4.83) and (4.84) yields for the block matrices $\boldsymbol{B}_i \in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{sd}}}$ in the three-dimensional case (3D)

$$\boldsymbol{B}_{\ell\, i} = \begin{bmatrix} N_{i,1} & . & . \\ . & N_{i,2} & . \\ . & . & N_{i,3} \\ N_{i,2} & N_{i,1} & . \\ . & N_{i,3} & N_{i,2} \\ N_{i,3} & . & N_{i,1} \end{bmatrix}, \boldsymbol{B}_{n\ell\, i} = \begin{bmatrix} u_{1,1}N_{i,1} & u_{2,1}N_{i,1} & u_{3,1}N_{i,1} \\ u_{1,2}N_{i,2} & u_{2,2}N_{i,2} & u_{3,2}N_{i,2} \\ u_{1,3}N_{i,3} & u_{2,3}N_{i,3} & u_{3,3}N_{i,3} \\ u_{1,2}N_{i,1} + u_{1,1}N_{i,2} & u_{2,2}N_{i,1} + u_{2,1}N_{i,2} & u_{3,2}N_{i,1} + u_{3,1}N_{i,2} \\ u_{1,3}N_{i,2} + u_{1,2}N_{i,3} & u_{2,3}N_{i,2} + u_{2,2}N_{i,3} & u_{3,3}N_{i,2} + u_{3,2}N_{i,3} \\ u_{1,3}N_{i,1} + u_{1,1}N_{i,3} & u_{2,3}N_{i,1} + u_{2,1}N_{i,3} & u_{3,3}N_{i,1} + u_{3,1}N_{i,3} \end{bmatrix}, \tag{4.87}$$

for axisymmetric problems (AXS)

$$\boldsymbol{B}_{\ell\, i} = \begin{bmatrix} N_{i,1} & . \\ . & N_{i,2} \\ \frac{N_i}{X_1} & . \\ N_{i,2} & N_{i,1} \end{bmatrix}, \quad \boldsymbol{B}_{n\ell\, i} = \begin{bmatrix} u_{1,1}N_{i,1} & u_{2,1}N_{i,1} \\ u_{1,2}N_{i,2} & u_{2,2}N_{i,2} \\ \frac{u_1}{X_1}\frac{N_i}{X_1} & . \\ u_{1,2}N_{i,1} + u_{1,1}N_{i,2} & u_{2,2}N_{i,1} + u_{2,1}N_{i,2} \end{bmatrix}, \tag{4.88}$$

and in two dimensions (2D)

$$\boldsymbol{B}_{\ell\, i} = \begin{bmatrix} N_{i,1} & . \\ . & N_{i,2} \\ N_{i,2} & N_{i,1} \end{bmatrix}, \quad \boldsymbol{B}_{n\ell\, i} = \begin{bmatrix} u_{1,1}N_{i,1} & u_{2,1}N_{i,1} \\ u_{1,2}N_{i,2} & u_{2,2}N_{i,2} \\ u_{1,2}N_{i,1} + u_{1,1}N_{i,2} & u_{2,2}N_{i,1} + u_{2,1}N_{i,2} \end{bmatrix}. \tag{4.89}$$

Remark 8 *Only for the case $\beta = 1$, i.e. for the discretized column matrices $\delta\boldsymbol{E}$ (4.92) and $\Delta\boldsymbol{E}$ (4.93), but not for $\boldsymbol{E}$ (4.91), an alternative representation can be obtained. Therefore, the sum $\boldsymbol{B}_\ell + 1\boldsymbol{B}_{n\ell}$ is evaluated and we take advantage of the component representation of the deformation gradient $\mathbf{F}$ as $F_{ij} = u_{i,j} + \delta_{ij}$ (with the Kronnecker[14] delta δ_{ij}). Applying this e.g. the first entry can be rewritten as $\boldsymbol{B}_{\ell\, 11} + \boldsymbol{B}_{nl\, 11} = N_{i,1} + u_{1,1}N_{i,1} = (1 + u_{1,1})N_{i,1} = F_{11}N_{i,1}$. The alternative representation is e.g. in the 2D case*[15]

$$\boldsymbol{B}_i = \begin{bmatrix} F_{11}N_{i,1} & F_{21}N_{i,1} \\ F_{12}N_{i,2} & F_{22}N_{i,2} \\ F_{12}N_{i,1} + F_{11}N_{i,2} & F_{22}N_{i,1} + F_{21}N_{i,2} \end{bmatrix}. \tag{4.90}$$

Introducing the discretization (4.83)–(4.84) into the representations of the strain vector (4.81) we finally obtain the spatially discrete strain column matrices

$$\begin{aligned} \boldsymbol{E}(\boldsymbol{X}, t) &= \boldsymbol{B}_a^*(\boldsymbol{u}_a(t), \boldsymbol{X})\, \boldsymbol{u}_a(t) && (4.91) \\ \delta\boldsymbol{E}(\boldsymbol{X}, t) &= \mathsf{D}_{\boldsymbol{u}}\boldsymbol{E}(\boldsymbol{u}_a)[\delta\boldsymbol{u}_a] = \boldsymbol{B}_a(\boldsymbol{u}_a(t), \boldsymbol{X})\, \delta\boldsymbol{u}_a && (4.92) \\ \Delta\boldsymbol{E}(\boldsymbol{X}, t) &= \mathsf{D}_{\boldsymbol{u}}\boldsymbol{E}(\boldsymbol{u}_a)[\Delta\boldsymbol{u}_a] = \boldsymbol{B}_a(\boldsymbol{u}_a(t), \boldsymbol{X})\, \Delta\boldsymbol{u}_a, && (4.93) \end{aligned}$$

notice the difference between $\boldsymbol{B}_a^*$ (4.85) and $\boldsymbol{B}_a$ (4.86), see Remark 6 on p. 64.

[14] $\delta_{ij} = 1$ if $i = j$, otherwise 0

[15] See, for example (Wriggers, 2001, Eq. (4.78)) and (Hartmann, 2003a, Eq. (5.81))

The split of the modal displacement vector $\boldsymbol{u}_a$ into unknown $\boldsymbol{u}$ and prescribed $\overline{\boldsymbol{u}}$ modal displacements as set forth in Eq. (4.70) can be reflected in partitioning the strain-displacement matrices accordingly as $\boldsymbol{B}_a^* = [\boldsymbol{B}^* \; \overline{\boldsymbol{B}}^*]$ and $\boldsymbol{B}_a = [\boldsymbol{B} \, \overline{\boldsymbol{B}}]$. We obtain for (4.91)

$$\boldsymbol{E}(\boldsymbol{X},t) = \boldsymbol{B}^*(\boldsymbol{u}_a(t),\boldsymbol{X})\,\boldsymbol{u}(t) + \overline{\boldsymbol{B}}^*(\boldsymbol{u}_a(t),\boldsymbol{X})\,\overline{\boldsymbol{u}}(t) \quad \text{with } \boldsymbol{B}^* \in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{u}}} \text{ and } \overline{\boldsymbol{B}}^* \in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{p}}}, \tag{4.94}$$

and the strain increment (4.93)

$$\Delta\boldsymbol{E}(\boldsymbol{X},t) = \boldsymbol{B}_a \Delta\boldsymbol{U}_a = \boldsymbol{B}(\boldsymbol{u}_a(t),\boldsymbol{X})\,\Delta\boldsymbol{u} + \overline{\boldsymbol{B}}(\boldsymbol{u}_a(t),\boldsymbol{X})\,\Delta\overline{\boldsymbol{u}} \tag{4.95}$$

with $\boldsymbol{B} \in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{u}}}$ and $\overline{\boldsymbol{B}} \in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{p}}}$.

For the variation (4.92) the prescribed displacement modes vanish by definition ($\delta\overline{\boldsymbol{u}} = \boldsymbol{0}$) and we have

$$\delta\boldsymbol{E}(\boldsymbol{X},t) = \boldsymbol{B}(\boldsymbol{u}_a(t),\boldsymbol{X})\,\delta\boldsymbol{u} \quad \text{with } \boldsymbol{B} \in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{u}}}. \tag{4.96}$$

Spatial discretization of the variational equation

Remark 9 *Unless otherwise stated we omit the subscript a of $\boldsymbol{u}_a$ for a leaner notation in the following. If the prescribed displacements $\overline{\boldsymbol{u}}$ are treated as unknowns with conditions to be satisfied by the solution like in the Lagrange multiplier or penalty method, see Sec. 4.9.1.2, one would replace $\boldsymbol{u}$ by $\boldsymbol{u}_a$ and accordingly $\boldsymbol{N}$ by $\boldsymbol{N}_a$.*

After this legwork we are in the position to derive the matrix/vector form of the nonlinear variational equation (3.9),

$$\begin{aligned} \pi^h(t,\boldsymbol{u}^h,\delta\boldsymbol{u}^h,\boldsymbol{q}^h) &= \pi^h_{\mathrm{int}}(t,\boldsymbol{u}^h,\delta\boldsymbol{u}^h,\boldsymbol{q}^h) - \pi^h_{\mathrm{ext}}(t,\boldsymbol{u}^h,\delta\boldsymbol{u}^h) = 0 && (4.97)\\ &= \delta\boldsymbol{u}(\tilde{\boldsymbol{g}}_{\mathrm{int}} - \tilde{\boldsymbol{g}}_{\mathrm{ext}}) = \delta\boldsymbol{u}(\boldsymbol{f}_{\mathrm{int}} - \boldsymbol{f}_{\mathrm{ext}}) = 0\,, && (4.98) \end{aligned}$$

and its linearization $\mathsf{L}[\pi^h]$ (3.52),

$$\begin{aligned} \mathsf{L}[\pi^h]_{(\boldsymbol{u}^h,\boldsymbol{q}^h)} &= (\pi^h_{\mathrm{int}} - \pi^h_{\mathrm{ext}}) + (\mathsf{D}_{\boldsymbol{u}^h}\,\pi^h_{\mathrm{int}}[\Delta\boldsymbol{u}^h] - \mathsf{D}_{\boldsymbol{u}^h}\,\pi^h_{\mathrm{ext}}[\Delta\boldsymbol{u}^h]) && (4.99)\\ &= \delta\boldsymbol{u}\Big((\tilde{\boldsymbol{g}}_{\mathrm{int}} - \tilde{\boldsymbol{g}}_{\mathrm{ext}}) + (\mathsf{D}_{\boldsymbol{u}}\,\tilde{\boldsymbol{g}}_{\mathrm{int}}[\Delta\boldsymbol{u}] - \mathsf{D}_{\boldsymbol{u}}\,\tilde{\boldsymbol{g}}_{\mathrm{ext}}[\Delta\boldsymbol{u}])\Big) = 0\,. \end{aligned}$$

As the test functions $\delta\boldsymbol{u}^h = \boldsymbol{N}\delta\boldsymbol{u}$ and consequently also their coefficients $\delta\boldsymbol{u}$ are arbitrary, the term in brackets must vanish and we can focus in the following on the expressions[16]

$$\begin{aligned} \tilde{\boldsymbol{g}}(t,\boldsymbol{u},\boldsymbol{q}) &= \tilde{\boldsymbol{g}}_{\mathrm{int}} - \tilde{\boldsymbol{g}}_{\mathrm{ext}} = \boldsymbol{f}_{\mathrm{int}} - \boldsymbol{f}_{\mathrm{ext}} = \boldsymbol{0} && (4.100)\\ \mathsf{L}[\tilde{\boldsymbol{g}}]_{(\boldsymbol{u},\boldsymbol{q})} &= (\tilde{\boldsymbol{g}}_{\mathrm{int}} - \tilde{\boldsymbol{g}}_{\mathrm{ext}}) + (\mathsf{D}_{\boldsymbol{u}}\,\tilde{\boldsymbol{g}}_{\mathrm{int}}[\Delta\boldsymbol{u}] - \mathsf{D}_{\boldsymbol{u}}\,\tilde{\boldsymbol{g}}_{\mathrm{ext}}[\Delta\boldsymbol{u}]) = \boldsymbol{0}\,. && (4.101) \end{aligned}$$

The integrals over the volume V and boundary A are approximated by the integrals over the discretized domain Ω and its surface $\Gamma = \partial\Omega$ and the individual terms for the four contributions are given below.

[16] A note on the notation: the tilde in $\tilde{\boldsymbol{g}}$ denotes that the integrals are evaluated over the whole domain Ω and no numerical integration is used. Consequently for a consistent notation the internal and external modal force vectors $\boldsymbol{f}_{\mathrm{int}}, \boldsymbol{f}_{\mathrm{ext}}$ in Eq. (4.100) should also bear a tilde. After the split in elements is carried out one could use $\hat{\boldsymbol{g}}$ and finally $\boldsymbol{g}$, Eq. (4.170), if numerical integration is applied. However, not to overload the notation the tilde is omitted for the force vectors.

1. The internal virtual work, (3.53), is written in column matrix form as

$$\pi^h_{\text{int}}(t, \boldsymbol{u}^h, \delta\boldsymbol{u}^h, \boldsymbol{q}^h) = \int_\Omega \delta\boldsymbol{E}^{hT}\big(\boldsymbol{X}, t\big)\tilde{\boldsymbol{T}}^h\big(\boldsymbol{C}^h, \boldsymbol{q}^h(\boldsymbol{C}^h)\big)\ \mathrm{d}\Omega \tag{4.102}$$

and its discretization exploiting the arbitrariness of $\delta\boldsymbol{u}$ using (4.73), (4.96), (4.62) allows to identify the internal force vector

$$\tilde{\boldsymbol{g}}_{\text{int}}(t, \boldsymbol{u}, \boldsymbol{q}) = \boldsymbol{f}_{\text{int}} = \int_\Omega \boldsymbol{B}^T\big(\boldsymbol{u}(t), \boldsymbol{X}\big)\tilde{\Phi}\Big(\boldsymbol{C}\big(\boldsymbol{E}(\boldsymbol{X}, t)\big), \boldsymbol{q}\big(\boldsymbol{C}(\boldsymbol{E}(\boldsymbol{X}, t))\big)\Big)\ \mathrm{d}\Omega. \tag{4.103}$$

2. For the external virtual work, (3.54), we obtain the vector representation

$$\begin{aligned}
\pi^h_{\text{ext}}(t, \boldsymbol{u}^h, \delta\boldsymbol{u}^h) \;&=\; \int_\Omega \delta\boldsymbol{u}^{hT}\rho_{\text{R}}(\boldsymbol{X})\boldsymbol{k}\ \mathrm{d}\Omega && \text{(work of body forces)} \\
&+\; \int_{\Gamma_{\text{tR}}} \delta\boldsymbol{u}^{hT}\boldsymbol{t}_{\text{R}}(\boldsymbol{X}, t)\ \mathrm{d}\Gamma_{\text{tR}} && \text{(work of surface traction)} \\
&-\; p\int_{\Omega_\xi} \delta\boldsymbol{u}^{hT}(\boldsymbol{x}_{,1} \times \boldsymbol{x}_{,2})\ \mathrm{d}\xi_1\mathrm{d}\xi_2\,. && \text{(work of follower load)}
\end{aligned} \tag{4.104}$$

The discretization is obtained by substituting the Ansatz (4.73) and taking the arbitrariness of $\delta\boldsymbol{u}$ into account we arrive at

$$\tilde{\boldsymbol{g}}_{\text{ext}}(t, \boldsymbol{u}) = \boldsymbol{f}_{\text{ext}} = \big(\boldsymbol{f}_{\text{ext k}} + \boldsymbol{f}_{\text{ext}\,t_{\text{R}}} + \boldsymbol{f}_{\text{ext}\,t_{\text{f}}}\big). \tag{4.105}$$

The load vector for conservative volume loading $\boldsymbol{k}$, Eq. $(4.104)_1$, is

$$\boldsymbol{f}_{\text{ext k}} = \int_\Omega \boldsymbol{N}^T\rho_{\text{R}}(\boldsymbol{X})\boldsymbol{k}\ \mathrm{d}\Omega, \tag{4.106}$$

while for conservative traction $\boldsymbol{t}_{\text{R}}$ loading, Eq. $(4.104)_2$, defined on the reference configuration we get

$$\boldsymbol{f}_{\text{ext}\,t_{\text{R}}} = \int_{\Gamma_{\text{tR}}} \boldsymbol{N}^T\boldsymbol{t}_{\text{R}}(\boldsymbol{X}, t)\ \mathrm{d}\Gamma_{\text{tR}}. \tag{4.107}$$

For the discretization of the follower load term $(4.104)_3$ where the integration is carried out over the deformed configuration (cf. Sec. 3.3.2) we note that the placement $\boldsymbol{x}$, the discretized form of $\boldsymbol{\phi}$ (3.40), is additively composed of the placement in reference configuration, the coordinate vector $\boldsymbol{X}$ (4.30), and the displacement $\boldsymbol{u}^h(t)$ (4.67), both can be expressed in coordinates $\boldsymbol{\xi}$ of the standard element $\Omega^{\text{q}}_{\text{st}}$, hence

$$\boldsymbol{x}(t) = \boldsymbol{X} + \boldsymbol{u}^h(t) = \boldsymbol{\phi}(\boldsymbol{\xi}) + \boldsymbol{N}(\boldsymbol{\xi})\boldsymbol{u}(t). \tag{4.108}$$

Consequently the convective base vectors $\boldsymbol{x}_{,i}$ are

$$\boldsymbol{x}_{,i} = \frac{\partial}{\partial\xi_i}\big(\boldsymbol{\phi}(\boldsymbol{\xi}) + \boldsymbol{N}(\boldsymbol{\xi})\boldsymbol{u}(t)\big) = \boldsymbol{\phi}_{,\xi_i}(\boldsymbol{\xi}) + \boldsymbol{N}_{,\xi_i}\boldsymbol{u}(t) \quad i = 1, \ldots, (n_{\text{sd}} - 1). \tag{4.109}$$

The derivatives of the mapping $\boldsymbol{\phi}_{,\xi_i}(\boldsymbol{\xi})$ can be easily obtained once a specific mapping is chosen.

Remark 10 *This is the general case where all the different mapping concepts discussed in Sec. 4.3 are included and typical for the separation of geometry and deformation in the p-version. The basis of the function space for* $\boldsymbol{X}$ *given by the mapping* $\boldsymbol{\phi}$ *is in general different from the basis used to describe the deformation* $\boldsymbol{u}^h$*. In the case of iso-parametric mapping more compact formulations can be obtained, see (Wriggers, 2001, Sec. 4.2.5) and (Simo et al., 1991) loosing the insight what contributions stem from the mapping and what is influenced by the Ansatz space.*

The follower load vector stemming from $(4.104)_3$ is in three dimensions

$$\boldsymbol{f}_{\text{ext}\,t_\text{f}} = -p \int_{\Omega_\xi} (\boldsymbol{x}_{,1} \times \boldsymbol{x}_{,2})\, \mathrm{d}\xi_1 \mathrm{d}\xi_2 = -p \int_{\Omega_\xi} \boldsymbol{n}\, \mathrm{d}\xi_1 \mathrm{d}\xi_2 \,. \tag{4.110}$$

For the implementation of the cross product it is best to evaluate its entries according to (4.109) and use the non-normalized outward normal vector $\boldsymbol{n}$

$$\boldsymbol{n} = \boldsymbol{x}_{,1} \times \boldsymbol{x}_{,2} = \begin{vmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \boldsymbol{e}_3 \\ \boldsymbol{x}_{1,1} & \boldsymbol{x}_{2,1} & \boldsymbol{x}_{3,1} \\ \boldsymbol{x}_{1,2} & \boldsymbol{x}_{2,2} & \boldsymbol{x}_{3,2} \end{vmatrix} = \begin{Bmatrix} \boldsymbol{x}_{2,1}\boldsymbol{x}_{3,2} - \boldsymbol{x}_{3,1}\boldsymbol{x}_{2,2} \\ \boldsymbol{x}_{3,1}\boldsymbol{x}_{1,2} - \boldsymbol{x}_{1,1}\boldsymbol{x}_{3,2} \\ \boldsymbol{x}_{1,1}\boldsymbol{x}_{2,2} - \boldsymbol{x}_{2,1}\boldsymbol{x}_{1,2} \end{Bmatrix} . \tag{4.111}$$

For the two-dimensional (3.48) and axisymmetric case (3.50) the cross product is simpler, as one of the base vectors is the unit vector $\boldsymbol{e}_3$. However, now we have to take care on which edge of the quadrilateral we are to calculate the outward and not the inward pointing normal vector (check the 'right hand rule'[17] in Fig. 4.15). As the mapping was stipulated to preserve the orientation of the coordinate system, $\det\mathbf{J} > 0$ (4.37), these considerations can be derived on the standard element and we evaluate

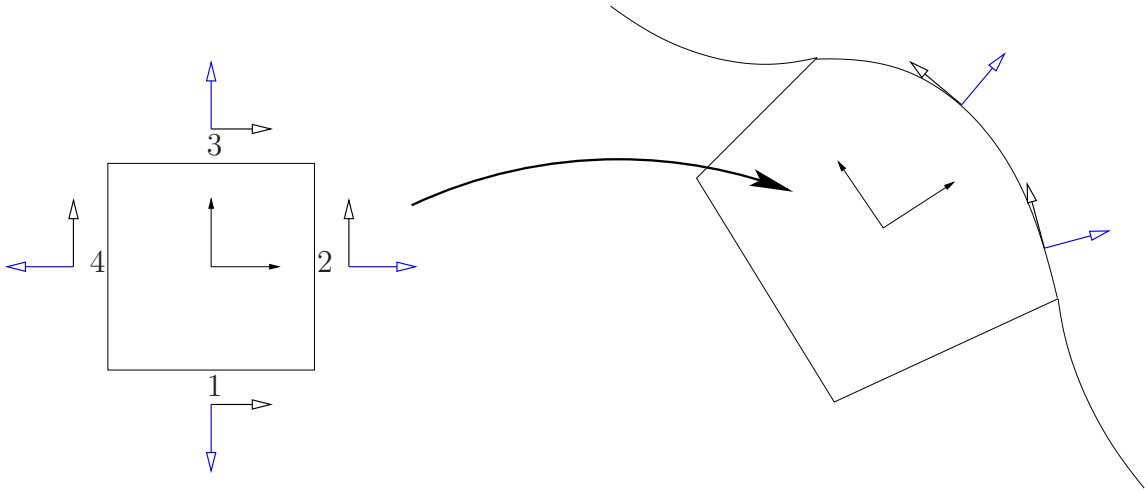


Figure 4.15: Standard and mapped element with tangential and normal vectors. Positive tangent vectors always point in the positive direction of the local coordinates.

$$\boldsymbol{n} = \begin{cases} \boldsymbol{x}_{,r} \times \boldsymbol{e}_3, & \text{for edge 1: } r = \xi, s = -1, \quad \text{for edge 2: } r = \eta, s = +1 \\ \boldsymbol{e}_3 \times \boldsymbol{x}_{,r}, & \text{for edge 3: } r = \xi, s = +1, \quad \text{for edge 4: } r = \eta, s = -1 \end{cases} \tag{4.112}$$

[17] cf. (Phan-Thien, 2002, p. 1)

where r, s is a re-parametrization of the edges in the local ξ, η coordinate system.

Without loss of generality we focus on the second case (edge 3 or 4) and have to evaluate

$$\boldsymbol{n} = \boldsymbol{e}_3 \times \boldsymbol{x}_{,r} = \left\{ \begin{array}{c} -x_{2,r} \\ x_{1,r} \\ 0 \end{array} \right\}. \tag{4.113}$$

As the third dimension perpendicular to the plane is neglected we can describe the re-ordering and sign change with a matrix

$$\boldsymbol{A} = \left\{ \begin{array}{l} \left[\begin{array}{cc} 0 & -1 \\ +1 & 0 \end{array} \right] \quad \text{for edge 3 and 4} \\ \\ \left[\begin{array}{cc} 0 & +1 \\ -1 & 0 \end{array} \right] \quad \text{for edge 1 and 2} \end{array} \right. \tag{4.114}$$

as

$$\boldsymbol{n} = \boldsymbol{A}\boldsymbol{x}_{,r} = \boldsymbol{A}\big(\boldsymbol{\phi}_{,r}(\boldsymbol{\xi}) + \boldsymbol{N}_{,r}\boldsymbol{u}(t)\big) \tag{4.115}$$

where we substituted (4.109) for $\boldsymbol{x}$.

So we can write for the axisymmetric follower load vector (3.50)

$$\begin{aligned} \boldsymbol{f}_{\text{ext}\,t_\text{f}} &= -2\pi\, p \int_{-1}^{1} \boldsymbol{N}^T \boldsymbol{n}\, x_1(r,s)\, \mathrm{d}r \\ &= -2\pi\, p \int_{-1}^{1} \boldsymbol{N}^T \boldsymbol{A}\big(\boldsymbol{\phi}_{,r}(r,s) + \boldsymbol{N}_{,r}\boldsymbol{u}(t)\big)\, x_1(r,s)\, \mathrm{d}r \end{aligned} \tag{4.116}$$

where the differential element $r(\xi)\mathrm{d}r$ from (3.50) transforms to $x_1(r,s)\,\mathrm{d}r$ as $x_1(r,s)$ is the radial distance from the axis of rotation, i.e. the first component of the vector $\boldsymbol{x}$, that is computed as

$$x_1(r,s) = \boldsymbol{X}_1 + \boldsymbol{u}_1^h = [1 \quad 0]\big(\boldsymbol{\phi}(r,s) + \boldsymbol{N}\boldsymbol{u}(t)\big)\,. \tag{4.117}$$

In two dimensions the differential element is simpler and we arrive for (3.48) at

$$\boldsymbol{f}_{\text{ext}\,t_\text{f}} = -p \int_{-1}^{1} \boldsymbol{N}^T \boldsymbol{n}\, \mathrm{d}r = -p \int_{-1}^{1} \boldsymbol{N}^T \boldsymbol{A}\big(\boldsymbol{\phi}_{,r}(r,s) + \boldsymbol{N}_{,r}\boldsymbol{u}(t)\big)\, \mathrm{d}r. \tag{4.118}$$

3. Now we focus on the directional derivatives. First the internal virtual work (3.55) is transferred to matrix/vector form

$$\begin{aligned} \mathsf{D}_{\boldsymbol{u}^h}\, \pi_{\text{int}}(t, \boldsymbol{u}^h, \delta\boldsymbol{u}^h, \boldsymbol{q}^h)[\Delta\boldsymbol{u}^h] &= \int_\Omega \delta\boldsymbol{\Theta}^{hT} \widehat{\widetilde{\boldsymbol{T}}} \Delta\boldsymbol{\Theta}^h\, \mathrm{d}\Omega \quad \text{(geometric stiffness)} \\ &+ \int_\Omega \delta\boldsymbol{E}^{hT} \tilde{\boldsymbol{C}}_L^h \Delta\boldsymbol{E}^h\, \mathrm{d}\Omega \quad \text{(material stiffness)} \end{aligned} \tag{4.119}$$

where the vector representation of the gradients $\operatorname{Grad}\delta\vec{u}, \operatorname{Grad}\Delta\vec{u}$ by $\delta\boldsymbol{\Theta}^h$, $\Delta\boldsymbol{\Theta}^h$ is given in (4.49), of the strains $\delta\boldsymbol{E}^h, \Delta\boldsymbol{E}^h$ in (4.57) and (4.58) respectively and $\tilde{\boldsymbol{C}}_L^h$ is the consistent tangent operator matrix from (4.66).

Introducing the discretizations (4.74), (4.96) and (4.95), again using the arbitrariness of $\delta\boldsymbol{u}$, we arrive at

$$\begin{aligned} \mathsf{D}_{\boldsymbol{u}^h}\tilde{\boldsymbol{g}}_{\text{int}}(t,\boldsymbol{u},\boldsymbol{q})[\Delta\boldsymbol{u}] &= \int_\Omega \boldsymbol{G}^T(\boldsymbol{X},t)\,\widehat{\tilde{\boldsymbol{T}}}\,\boldsymbol{G}(\boldsymbol{X},t)\;\mathrm{d}\Omega\,\Delta\boldsymbol{u} \\ &+ \int_\Omega \boldsymbol{B}^T(\boldsymbol{u}(t),\boldsymbol{X})\,\tilde{\boldsymbol{C}}_L\,\boldsymbol{B}(\boldsymbol{u}(t),\boldsymbol{X})\;\mathrm{d}\Omega\,\Delta\boldsymbol{u} \quad . \end{aligned} \tag{4.120}$$

The global tangential stiffness matrices valid at the state ($\boldsymbol{u}$,$\boldsymbol{q}$) are introduced as the stiffness from geometric nonlinearity

$$\boldsymbol{K}_G = \int_\Omega \boldsymbol{G}^T(\boldsymbol{X},t)\,\widehat{\tilde{\boldsymbol{T}}}\,\boldsymbol{G}(\boldsymbol{X},t)\;\mathrm{d}\Omega \tag{4.121}$$

and the stiffness as a result of the nonlinear constitutive model

$$\boldsymbol{K}_C = \int_\Omega \boldsymbol{B}^T(\boldsymbol{u}(t),\boldsymbol{X})\,\tilde{\boldsymbol{C}}_L\,\boldsymbol{B}(\boldsymbol{u}(t),\boldsymbol{X})\;\mathrm{d}\Omega\,. \tag{4.122}$$

The matrix $\widehat{\tilde{\boldsymbol{T}}} \in \mathbb{R}^{n_{\text{gr}}\times n_{\text{gr}}}$ holds the components of the stress tensor $\tilde{\mathbf{T}}$ in a special arrangement so that the tensorial expression $\operatorname{Grad}\Delta\vec{u}\,\tilde{\mathbf{T}}\cdot\operatorname{Grad}\delta$ from Eq. (3.55) is represented by the matrix/vector notation in Eq. (4.121). We obtain the following matrices,
3D:

$$\widehat{\tilde{\boldsymbol{T}}} = \begin{bmatrix} \overline{\tilde{\boldsymbol{T}}} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \overline{\tilde{\boldsymbol{T}}} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \overline{\tilde{\boldsymbol{T}}} \end{bmatrix} \quad \text{with} \quad \overline{\tilde{\boldsymbol{T}}} = \begin{bmatrix} \tilde{T}_{11} & \tilde{T}_{12} & \tilde{T}_{13} \\ \tilde{T}_{12} & \tilde{T}_{22} & \tilde{T}_{23} \\ \tilde{T}_{13} & \tilde{T}_{23} & \tilde{T}_{33} \end{bmatrix} \tag{4.123}$$

AXS:

$$\widehat{\tilde{\boldsymbol{T}}} = \begin{bmatrix} \overline{\tilde{\boldsymbol{T}}} & \boldsymbol{0}_{2\times2} & 0 \\ \boldsymbol{0}_{2\times2} & \overline{\tilde{\boldsymbol{T}}} & 0 \\ 0 & 0 & \tilde{T}_{33} \end{bmatrix} \quad \text{with} \quad \overline{\tilde{\boldsymbol{T}}} = \begin{bmatrix} \tilde{T}_{11} & \tilde{T}_{12} \\ \tilde{T}_{12} & \tilde{T}_{22} \end{bmatrix} \tag{4.124}$$

2D:

$$\widehat{\tilde{\boldsymbol{T}}} = \begin{bmatrix} \overline{\tilde{\boldsymbol{T}}} & \boldsymbol{0}_{2\times2} \\ \boldsymbol{0}_{2\times2} & \overline{\tilde{\boldsymbol{T}}} \end{bmatrix} \quad \text{with} \quad \overline{\tilde{\boldsymbol{T}}} = \begin{bmatrix} \tilde{T}_{11} & \tilde{T}_{12} \\ \tilde{T}_{12} & \tilde{T}_{22} \end{bmatrix}. \tag{4.125}$$

4. Finally, we address the matrix/vector representation of the directional derivative of follower loading, Eq. (3.56).

In the **three-dimensional case**, (3.47), the vector form is

$$\mathsf{D}_{\boldsymbol{u}^h}\pi_{\text{ext}}(t,\boldsymbol{u}^h,\delta\boldsymbol{u}^h)[\Delta\boldsymbol{u}^h] = -p\int_{\Omega_\xi}\delta\boldsymbol{u}^h\cdot\left(\Delta\boldsymbol{u}^h_{,1}\times\boldsymbol{x}_{,2}+\boldsymbol{x}_{,1}\times\Delta\boldsymbol{u}^h_{,2}\right)\,\mathrm{d}\xi_1\mathrm{d}\xi_2 \quad (4.126)$$

with $\boldsymbol{x}_{,i}\quad i=1,2$ as in Eq. (4.109). For the evaluation of the cross products we start with

$$\Delta\boldsymbol{u}^h_{,1}\times\boldsymbol{x}_{,2} = \begin{vmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \boldsymbol{e}_3 \\ \Delta\boldsymbol{u}^h{}_{1,1} & \Delta\boldsymbol{u}^h{}_{2,1} & \Delta\boldsymbol{u}^h{}_{3,1} \\ \boldsymbol{x}_{1,2} & \boldsymbol{x}_{2,2} & \boldsymbol{x}_{3,2} \end{vmatrix} = \begin{Bmatrix} \Delta\boldsymbol{u}^h{}_{2,1}\boldsymbol{x}_{3,2}-\Delta\boldsymbol{u}^h{}_{3,1}\boldsymbol{x}_{2,2} \\ \Delta\boldsymbol{u}^h{}_{3,1}\boldsymbol{x}_{1,2}-\Delta\boldsymbol{u}^h{}_{1,1}\boldsymbol{x}_{3,2} \\ \Delta\boldsymbol{u}^h{}_{1,1}\boldsymbol{x}_{2,2}-\Delta\boldsymbol{u}^h{}_{2,1}\boldsymbol{x}_{1,2} \end{Bmatrix} \quad (4.127)$$

To be able to pull out $\Delta\boldsymbol{u}^h_{,1}$ we rewrite the vector product with a skew symmetric matrix

$$\Delta\boldsymbol{u}^h_{,1}\times\boldsymbol{x}_{,2} = \begin{bmatrix} 0 & \boldsymbol{x}_{3,2} & -\boldsymbol{x}_{2,2} \\ -\boldsymbol{x}_{3,2} & 0 & \boldsymbol{x}_{1,2} \\ \boldsymbol{x}_{2,2} & -\boldsymbol{x}_{1,2} & 0 \end{bmatrix} \begin{Bmatrix} \Delta\boldsymbol{u}^h{}_{1,1} \\ \Delta\boldsymbol{u}^h{}_{2,1} \\ \Delta\boldsymbol{u}^h{}_{3,1} \end{Bmatrix} = \boldsymbol{A}_{,2}\Delta\boldsymbol{u}^h_{,1}\,. \quad (4.128)$$

As $a\times b=-(b\times a)$ we can readily obtain a similar expression for the second cross product in (4.126)

$$\boldsymbol{x}_{,1}\times\Delta\boldsymbol{u}^h_{,2} = -(\Delta\boldsymbol{u}^h_{,2}\times\boldsymbol{x}_{,1}) = -\begin{bmatrix} 0 & \boldsymbol{x}_{3,1} & -\boldsymbol{x}_{2,1} \\ -\boldsymbol{x}_{3,1} & 0 & \boldsymbol{x}_{1,1} \\ \boldsymbol{x}_{2,1} & -\boldsymbol{x}_{1,1} & 0 \end{bmatrix}\Delta\boldsymbol{u}^h_{,2} = -\boldsymbol{A}_{,1}\Delta\boldsymbol{u}^h_{,2}\,. \quad (4.129)$$

Spatial discretization using Eq. (4.69), $\Delta\boldsymbol{u}^h_{,i}=\boldsymbol{N}_{,i}\Delta\boldsymbol{u}$, yields

$$\begin{aligned} \mathsf{D}_{\boldsymbol{u}}\tilde{\boldsymbol{g}}_{\text{ext}}(t,\boldsymbol{u},)[\Delta\boldsymbol{u}] &= -p\int_{-1}^{1}\int_{-1}^{1}\boldsymbol{N}^T\left(\boldsymbol{A}_{,2}\Delta\boldsymbol{u}_{,1}-\boldsymbol{A}_{,1}\Delta\boldsymbol{u}_{,2}\right)\mathrm{d}\xi\mathrm{d}\eta \\ &= -p\int_{-1}^{1}\int_{-1}^{1}\boldsymbol{N}^T\left(\boldsymbol{A}_{,2}\boldsymbol{N}_{,1}-\boldsymbol{A}_{,1}\boldsymbol{N}_{,2}\right)\mathrm{d}\xi\mathrm{d}\eta\,\Delta\boldsymbol{u} \end{aligned} \quad (4.130)$$

where we can identify the displacement-depending follower tangent stiffness matrix for three-dimensional problems

$$\boldsymbol{K}_F = -p\int_{-1}^{1}\int_{-1}^{1}\boldsymbol{N}^T\left(\boldsymbol{A}_{,2}\boldsymbol{N}_{,1}-\boldsymbol{A}_{,1}\boldsymbol{N}_{,2}\right)\mathrm{d}\xi\mathrm{d}\eta \quad (4.131)$$

with the matrices $\boldsymbol{A}_{,1}$ (4.129) and $\boldsymbol{A}_{,2}$ (4.128) for the cross-product.

The directional derivative in the **plain stress/strain** case (3.49) is transferred to column matrix notation as (w.l.o.g. for edge 3 and 4)

$$\mathsf{D}_{\boldsymbol{u}^h}\pi_{\text{ext}}(t,\boldsymbol{u}^h,\delta\boldsymbol{u}^h)[\Delta\boldsymbol{u}^h] = -p\int_{\Omega_\xi}\delta\boldsymbol{u}^h\cdot\boldsymbol{n}\,\mathrm{d}\xi = -p\int_{\Omega_\xi}\delta\boldsymbol{u}^h\cdot\left(\boldsymbol{e}_3\times\Delta\boldsymbol{u}^h_{,1}\right)\,\mathrm{d}\xi\,. \quad (4.132)$$

The cross product is evaluated as in Eq. (4.113) with the matrix $\boldsymbol{A}$ (4.114) as

$$\boldsymbol{e}_3 \times \Delta \boldsymbol{u}^h_{,1} = \boldsymbol{A} \Delta \boldsymbol{u}^h_{,1} \,. \tag{4.133}$$

Introducing the Ansatz Eq. (4.69) we obtain the spatial discretization

$$\mathsf{D}_{\boldsymbol{u}} \tilde{\boldsymbol{g}}_{\text{ext}}(t, \boldsymbol{u})[\Delta \boldsymbol{u}] = -p \int_{-1}^{1} \boldsymbol{N}^T \boldsymbol{A} \boldsymbol{N}_{,1} \, \mathrm{d}\xi \, \Delta \boldsymbol{u}, \tag{4.134}$$

and the follower stiffness matrix for the plain strain/stress case is abbreviated as

$$\boldsymbol{K}_F = -p \int_{-1}^{1} \boldsymbol{N}^T \boldsymbol{A} \boldsymbol{N}_{,1} \, \mathrm{d}\xi \tag{4.135}$$

with the matrix $\boldsymbol{A}$ defined in (4.114).

In the **axisymmetric** case we give the column matrix form of the directional derivative (3.51) (w.l.o.g. for edge 3 and 4) as

$$\begin{aligned} \mathsf{D}_{\boldsymbol{u}^h} \pi_{\text{ext}}(t, \boldsymbol{u}^h, \delta \boldsymbol{u}^h)[\Delta \boldsymbol{u}^h] &= -2\pi p \int_{-1}^{1} \delta \boldsymbol{u}^h \cdot \left((\boldsymbol{e}_3 \times \Delta \boldsymbol{u}^h_{,1}) x_1 + \boldsymbol{n} \Delta u^h_1 \right) \mathrm{d}\xi \qquad (4.136) \\ &= -2\pi p \int_{-1}^{1} \delta \boldsymbol{u}^h \cdot \left((\boldsymbol{e}_3 \times \Delta \boldsymbol{u}^h_{,1}) x_1 + (\boldsymbol{e}_3 \times \boldsymbol{x}_{,1}) \Delta u^h_1 \right) \mathrm{d}\xi \end{aligned}$$

and substituting (4.133) for the first cross product and (4.115) for the second we obtain

$$\mathsf{D}_{\boldsymbol{u}^h} \pi_{\text{ext}}(t, \boldsymbol{u}^h, \delta \boldsymbol{u}^h)[\Delta \boldsymbol{u}^h] = -2\pi p \int_{-1}^{1} \delta \boldsymbol{u}^h \cdot \left((\boldsymbol{A} \Delta \boldsymbol{u}^h_{,1}) x_1 + (\boldsymbol{A} \boldsymbol{x}_{,1}) \Delta u^h_1 \right) \mathrm{d}\xi \,. \tag{4.137}$$

Spatial discretization by the Ansatz, Eq. (4.69), and expressing $\Delta u^h_1 = [1 \quad 0] \Delta \boldsymbol{u}^h = [1 \quad 0] \boldsymbol{N} \Delta \boldsymbol{u}$ yields

$$\mathsf{D}_{\boldsymbol{u}} \tilde{\boldsymbol{g}}_{\text{ext}}(t, \boldsymbol{u})[\Delta \boldsymbol{u}] = -2\pi p \int_{-1}^{1} \left(\boldsymbol{N}^T \boldsymbol{A} \boldsymbol{N}_{,1} x_1 + \boldsymbol{N}^T \boldsymbol{A} \boldsymbol{x}_{,1} [1 \quad 0] \boldsymbol{N} \right) \mathrm{d}\xi \, \Delta \boldsymbol{u}, \tag{4.138}$$

where x_1 is the first component of the known placement $\boldsymbol{x}$ (4.108), $\boldsymbol{x}_{,1}$ from (4.109) and matrix $\boldsymbol{A} \in \mathbb{R}^{2\times 2}$ as given in (4.114). The follower stiffness for the axisymmetric case is defined as

$$\begin{aligned} \boldsymbol{K}_F &= -2\pi p \int_{-1}^{1} \left(\boldsymbol{N}^T \boldsymbol{A} \boldsymbol{N}_{,1} x_1 + \boldsymbol{N}^T \boldsymbol{n} [1 \quad 0] \boldsymbol{N} \right) \mathrm{d}\xi \qquad (4.139) \\ &= -2\pi p \int_{-1}^{1} \left(\boldsymbol{N}^T \boldsymbol{A} \boldsymbol{N}_{,1} x_1 + \boldsymbol{N}^T \boldsymbol{A} \boldsymbol{x}_{,1} [1 \quad 0] \boldsymbol{N} \right) \mathrm{d}\xi \,. \qquad (4.140) \end{aligned}$$

Checking the dimensions of the follower stiffness we see

$$\begin{aligned} n_{\text{dof}} \times n_{\text{dof}} &= (n_{\text{dof}} \times 2)(2 \times 2)(2 \times n_{\text{dof}})1 + \\ &\quad +(n_{\text{dof}} \times 2)(2 \times 2)(2 \times 1)(1 \times 2)(2 \times n_{\text{dof}}) \end{aligned} \tag{4.141}$$

that the matrix is as expected $n_{\text{dof}} \times n_{\text{dof}}$.

Summarizing, the global tangential stiffness matrix of the whole domain is composed of the stiffness resulting from physical nonlinearity ('material stiffness') $\boldsymbol{K}_C$ (4.122), the stiffness induced by geometric nonlinearity $\boldsymbol{K}_G$ (4.121) and the follower load stiffness $\boldsymbol{K}_F$ given in equations (4.131), (4.135) and (4.139) for the different cases

$$\boldsymbol{K} = \boldsymbol{K}_C + \boldsymbol{K}_G - \boldsymbol{K}_F \,. \tag{4.142}$$

It is apparent from the linearized form, Eq. (4.101), that the follower stiffness must be subtracted.

The column matrix representation of the linearized form $\mathsf{L}[\tilde{\boldsymbol{g}}]$, Eq. (4.101), corresponding to the state $(\boldsymbol{u}, \boldsymbol{q}, t)$ can now be written as

$$\mathsf{L}[\tilde{\boldsymbol{g}}]_{(\boldsymbol{u},\boldsymbol{q})} = (\boldsymbol{K}_C + \boldsymbol{K}_G - \boldsymbol{K}_F)\Delta\boldsymbol{u} + \boldsymbol{f}_{\text{int}} - \boldsymbol{f}_{\text{ext}} = \boldsymbol{0} \tag{4.143}$$

with the external force vector $\boldsymbol{f}_{\text{ext}}$ as introduced in Eq. (4.105) and the internal forces $\boldsymbol{f}_{\text{int}}$ (4.103).

4.6.3 Element level quantities and their assembly

So far all quantities are functions defined over the whole discretized domain Ω. The splitting of the domain into non-overlapping and continuous elements Ω^{e}, cf. Sec. 4.1, and the construction of the shape functions as introduced in Sec. 4.2 ensures C^0 continuity of the displacements. Furthermore, this constitutes that the global functions have local support in the sense that displacements and strains obtained by the global functions $(\bullet)^h$ living in the FE-space over the element Ω^{e} coincide with the displacements and strains evaluated locally on this element $(\bullet)^{eh}$,

$$\mathbf{u}^{e\,h}\big((\boldsymbol{\phi}^e)^{-1}(\mathbf{X}), t\big) \equiv \boldsymbol{u}^h(\boldsymbol{X}, t) = \boldsymbol{N}_a(\boldsymbol{X})\,\boldsymbol{u}_a(t) = \mathbf{N}^e\big((\boldsymbol{\phi}^e)^{-1}(\mathbf{X})\big)\,\mathbf{u}^e(t) \quad \text{for } \boldsymbol{X} = \mathbf{X} \in \Omega^{\text{e}}, \tag{4.144}$$

where $\boldsymbol{\xi} = (\boldsymbol{\phi}^e)^{-1}(\mathbf{X}) \in \mathbb{R}^{n_{\text{sd}}}$ is the inverse mapping introduced in Eq. (4.31) of the local position vector $\mathbf{X} \in \mathbb{R}^{n_{\text{sd}}}$ to the coordinates in the standard element $\boldsymbol{\xi} \in \Omega^{\text{q}}_{\text{st}}$. $\mathbf{N}^e(\boldsymbol{\xi}) \in \mathbb{R}^{n_{\text{sd}} \times n_{\text{em}}}$ represents the matrix of shape functions defined on the element Ω^{e} and $\mathbf{u}^e(t) \in \mathbb{R}^{n_{\text{em}}}$ is the vector of modal degrees of freedom[18] of element Ω^{e}. The number of local degrees of freedom of an element is denoted as n_{em} ('number of element modes') and solely these degrees of freedom — in the higher order sense: the coefficients of the element modes — are necessary to describe the displacement field in the region of the element Ω^{e}. Furthermore, we have the defining equivalences for the elemental strain-displacement matrices $\mathbf{B}^{*e}(\mathbf{u}^e(t), (\boldsymbol{\phi}^e)^{-1}(\mathbf{X}))$,

[18] As the polynomial degree can vary from element to element and even in the local directions the number of element modes, n_{em}, may be different on each element. For four-noded elements as used here the lower bound is $n_{\text{em}} \geq 4n_{\text{sd}}$.

$\mathbf{B}^e(\mathbf{u}^e(t),(\boldsymbol{\phi}^e)^{-1}(\mathbf{X}))$, the strain column matrices $\mathbf{E}^e((\boldsymbol{\phi}^e)^{-1}(\mathbf{X}),t)$, $\delta\mathbf{E}^e((\boldsymbol{\phi}^e)^{-1}(\mathbf{X}))$, and the virtual displacement $\delta\mathbf{u}^{e\,h}$for $\boldsymbol{X}=\mathbf{X}\in\Omega^{\mathrm{e}}$

$$\begin{aligned}
\text{(4.51),(4.85)}\quad \mathbf{E}^e((\boldsymbol{\phi}^e)^{-1}(\mathbf{X}),t) &\equiv \boldsymbol{E}^h(\boldsymbol{X},t)=\boldsymbol{B}_a^*(\boldsymbol{u}_a(t),\boldsymbol{X})\,\boldsymbol{u}_a(t) && (4.145)\\
&\equiv \mathbf{B}^{*e}(\mathbf{u}^e(t),(\boldsymbol{\phi}^e)^{-1}(\mathbf{X}))\,\mathbf{u}^e(t)\\
\text{(4.57),(4.96)}\quad \delta\mathbf{E}^e((\boldsymbol{\phi}^e)^{-1}(\mathbf{X}),t) &\equiv \delta\boldsymbol{E}^h(\boldsymbol{X},t)=\boldsymbol{B}(\boldsymbol{u}_a(t),\boldsymbol{X})\,\delta\boldsymbol{u}_a && (4.146)\\
&= \mathbf{B}^e(\mathbf{u}^e(t),(\boldsymbol{\phi}^e)^{-1}(\mathbf{X}))\,\delta\mathbf{u}^e\,.\\
\text{(4.68)}\quad \delta\mathbf{u}^{e\,h}((\boldsymbol{\phi}^e)^{-1}(\mathbf{X})) &\equiv \delta\boldsymbol{u}^h(\boldsymbol{X})=\boldsymbol{N}_a(\boldsymbol{X})\delta\,\boldsymbol{u}_a=\mathbf{N}^e((\boldsymbol{\phi}^e)^{-1}(\mathbf{X}))\delta\mathbf{u}^e\,. && (4.147)
\end{aligned}$$

The correlation between element modal displacements $\mathbf{u}^e(t)$ and all displacements $\boldsymbol{u}_a(t)$ is given by

$$\mathbf{u}^e=\boldsymbol{Z}_a^e\boldsymbol{u}_a, \tag{4.148}$$

where the incidence matrix $\boldsymbol{Z}_a^e\in\mathbb{R}^{n_{\mathrm{em}}\times n_{\mathrm{dof}}}$ is introduced. This matrix is partitioned according to the split of the vectors $\boldsymbol{u}_a=\{\boldsymbol{u}\quad\overline{\boldsymbol{u}}\}^T$ and $\delta\boldsymbol{u}_a$, Eq. (4.70), as

$$\boldsymbol{Z}_a^e=\left[\boldsymbol{Z}^e\,\overline{\boldsymbol{Z}}^e\right] \tag{4.149}$$

with $\boldsymbol{Z}^e\in\mathbb{R}^{n_{\mathrm{em}}\times n_{\mathrm{u}}}$ and $\overline{\boldsymbol{Z}}^e\in\mathbb{R}^{n_{\mathrm{em}}\times n_{\mathrm{p}}}$ resulting in

$$\begin{aligned}
\mathbf{u}^e &= \boldsymbol{Z}^e\boldsymbol{u}+\overline{\boldsymbol{Z}}^e\overline{\boldsymbol{u}},\qquad \delta\mathbf{u}^e=\boldsymbol{Z}^e\delta\boldsymbol{u},\qquad \Delta\mathbf{u}^e=\boldsymbol{Z}^e\Delta\boldsymbol{u} && (4.150)\\
\mathbf{E}^e &= \mathbf{B}^{*e}\left[\boldsymbol{Z}^e\,\overline{\boldsymbol{Z}}^e\right]\left\{\begin{array}{c}\boldsymbol{u}\\ \overline{\boldsymbol{u}}\end{array}\right\}=\mathbf{B}^{*e}\left\{\boldsymbol{Z}^e\boldsymbol{u}+\overline{\boldsymbol{Z}}^e\overline{\boldsymbol{u}}\right\} && (4.151)\\
\delta\mathbf{E}^e &= \mathbf{B}^e\boldsymbol{Z}^e\delta\boldsymbol{u} && (4.152)\\
\Delta\mathbf{E}^e &= \mathbf{B}^e\boldsymbol{Z}^e\Delta\boldsymbol{u}. && (4.153)
\end{aligned}$$

Remark 11 *The incidence matrices holding only* 0 *and* 1 *are introduced here to formalize and keep track of the assembly operations implemented in program codes and show the transition from elemental to global (assembled) quantities. The benefits of this notation together with the split in unknown and prescribed degrees of freedom was shown to provide insight for questions of displacement control (Hartmann, 2003b) and the computation of reaction forces (Hartmann, 2003a). In (Belytschko et al., 2000, p. 39) the incidence matrices are called 'connectivity matrices', (Hughes, 2000, p.92) also describes the process in detail. Of course the large sparse matrix* $\boldsymbol{Z}_a^e$ *does not have to be stored in memory. In the p-version code AdhoC (Düster et al., 2004) for example the element's data structure holds the inverse relation, a 'location matrix'* $\in\mathbb{R}^{n_{em}\times 1}$ *storing the corresponding global degrees of freedom for the local modes, see (Düster, 2005, Sec. 5.2.1).*

We now apply the decomposition into elements Ω^{e} and obtain exemplarily for the discretized internal virtual work column matrix $\tilde{\boldsymbol{g}}_{\mathrm{int}}$, Eq. (4.103), the so called internal force vector $\boldsymbol{f}_{\mathrm{int}}$,

$$\begin{aligned}
\tilde{\boldsymbol{g}}(t,\boldsymbol{u},\boldsymbol{q}) &= \sum_{e=1}^{n_{\mathrm{e}}}\boldsymbol{Z}^{eT}\underbrace{\left\{\int_{\Omega^{\mathrm{e}}}\mathbf{B}^{eT}\big(\mathbf{u}^e(t),\mathbf{X}\big)\tilde{\boldsymbol{\Phi}}\Big(\mathbf{C}^e\big(\mathbf{E}^e(\mathbf{X},t)\big),\boldsymbol{q}^h\big(\mathbf{C}^e(\mathbf{E}^e(\mathbf{X},t))\big)\Big)\,\mathrm{d}\Omega^{\mathrm{e}}\right\}}_{\mathbf{f}_{\mathrm{int}}^e}\\
&\quad -\boldsymbol{f}_{\mathrm{ext}}(t,\boldsymbol{u})=\boldsymbol{0}\,. && (4.154)
\end{aligned}$$

The abbreviation $\mathbf{f}^e_{\text{int}}$ for the elemental contribution in curly brackets is introduced and the global internal force vector is given as

$$\boldsymbol{f}_{\text{int}} = \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \mathbf{f}^e_{\text{int}} \,. \tag{4.155}$$

For the external force vector (4.105) we obtain by the same reasoning the element level counterparts

$$\boldsymbol{f}_{\text{ext}} = \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \mathbf{f}^e_{\text{ext}} = \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \left[\mathbf{f}^e_{\text{ext k}} + \mathbf{f}^e_{\text{ext } t_{\text{R}}} + \mathbf{f}^e_{\text{ext } t_{\text{f}}} \right] . \tag{4.156}$$

For the assembly of the stiffness matrices stemming from a Gâteaux derivative we take Eq. $(4.150)_3$ into account and arrive at a representation where the local element matrix is multiplied with the incidence matrix from the right and its transpose from the left

$$\boldsymbol{K} = \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \mathbf{k}^e \boldsymbol{Z}^e = \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \left[\mathbf{k}^e_C + \mathbf{k}^e_G - \mathbf{k}^e_F \right] \boldsymbol{Z}^e \tag{4.157}$$

(a forward reference: the matrix $\mathbf{k}^e_G$ is given on Gauss-point level in (4.202), $\mathbf{k}^e_C$ in (4.207)).

The global nonlinear equation system (4.100)

$$\tilde{\boldsymbol{g}} = \tilde{\boldsymbol{g}}_{\text{int}} - \tilde{\boldsymbol{g}}_{\text{ext}} = \boldsymbol{0} \tag{4.158}$$

and its linearization (4.143) can now be expressed as the assembly of elemental contributions

$$\mathsf{L}[\tilde{\boldsymbol{g}}]_{(\boldsymbol{u},\boldsymbol{q})} = \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \left[\mathbf{k}^e_C + \mathbf{k}^e_G - \mathbf{k}^e_F \right] \boldsymbol{Z}^e \Delta\boldsymbol{u} + \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \left[\mathbf{f}^e_{\text{int}} - \mathbf{f}^e_{\text{ext}} \right] = \boldsymbol{0}. \tag{4.159}$$

4.6.4 Numerical integration

The volume integrals hidden in (4.154), (4.156) and (4.157) to be evaluated on element Ω^{e} can be written in the generic form

$$I = \int_{\Omega^{\text{e}}} F(x, y, z) \, \mathrm{d}\Omega^{\text{e}} \,. \tag{4.160}$$

We introduce a coordinate transformation to the standard element, Eq. (4.30),

$$I = \int_{-1}^{1} \int_{-1}^{1} \int_{-1}^{1} F\Big(\boldsymbol{\phi}^e(\xi, \eta, \zeta) \Big) \det\mathbf{J} \, \mathrm{d}\xi \mathrm{d}\eta \mathrm{d}\zeta \tag{4.161}$$

where the determinant of the Jacobian, Eq. (4.36), accounts for the mapping from the standard element to the reference configuration.

For the numerical integration different quadrature schemes can be used, see (Schwarz, 2004, Ch. 7), (Szabó and Babuška, 1991, Sec. 3.3), (Solín et al., 2003, Ch. 4). One important

class interpolates the integrand F by a polynomial that is sampled at certain points $\boldsymbol{\xi}_{jkl}$ to calculate an approximation of the integral by a weighted sum

$$I \approx \sum_j \sum_k \sum_l w_j w_k w_l \, F\Big(\boldsymbol{\phi}^e(\boldsymbol{\xi}_{jkl})\Big) \, \mathrm{det}\mathbf{J} \, \mathrm{d}\xi \mathrm{d}\eta \mathrm{d}\zeta \,. \tag{4.162}$$

Exploiting the tensor-product structure of the Ansatz functions in two and three dimensions the volume integral is reduced to one-dimensional integration for each spatial direction

$$\int_{-1}^{+1} f(x) \mathrm{d}x \approx \sum_{i=n}^{n} w_i f(x_i) \quad \text{with} \quad x_i \in [-1;1] \,. \tag{4.163}$$

If the integrand is known at equally-spaced points, integration formulae based on Newton-Cotes rules can be used. Higher accuracy can be obtained with non-equidistant points. In the finite element context Gauss-Legendre[19]quadrature is commonly applied.[20] For the Gauss-Legendre scheme there exists exactly one set of quadrature points and weights to be used in Eq. (4.163) with n sampling points x_i in the interior of the interval $[-1;1]$ that exactly integrate a polynomial of order

$$p = 2n - 1 \,, \tag{4.164}$$

i.e. to exactly integrate a polynomial with order p we need

$$n \geq \frac{p+1}{2} \tag{4.165}$$

sampling points.

The Gauss points or sampling points x_i are the roots of the n-th Legendre polynomial, Eq. (4.16), and the corresponding weights are given as

$$w_i = \int_{-1}^{1} \prod_{\substack{j=1 \\ j \neq i}}^{n} \left(\frac{x - x_j}{x_i - x_j} \right)^2 \mathrm{d}x > 0, \quad i = 1, 2, \ldots, n. \tag{4.166}$$

[19]While for the Gauss-Legendre quadrature the n abscissae and n locations of n sampling points in the interior of the unit interval give optimal accuracy and interpolate with $2n$ parameters a polynomial with $2n$ coefficients, i.e. a polynomial of order $p_{\mathrm{GLeg}} = 2n - 1$ exactly, it may sometimes be useful to include the end points of the interval as sampling points. If just one end point is used the Gauss-Radau scheme is obtained, if both endpoints are sampling points the so called Gauss-Lobatto scheme is recovered. Prescribing the location of sampling points is at the expense of accuracy. For n sampling points the Gauss-Radau scheme exactly integrates a polynomial of order $p_{\mathrm{GRad}} = 2n - 2$, while the Gauss-Lobatto quadrature is only exact for polynomials up to order $p_{\mathrm{GLob}} = 2n - 3$ (Überhuber, 2001, p. 108).

[20]While Gauss quadrature expands the integrand in terms of polynomials, the Clenshaw-Curtis quadrature is based on the expansion of the integrand in terms of Chebyshev polynomials, see (Trefethen, 2007). This construction leads to nested quadrature points allowing adaptive integration. The Clenshaw-Curtis schemes integrate with n points only polynomials up to order $n - 1$. However, if the integrand is no polynomial there is a wide class of functions where the approximation in terms of Chebyshev polynomials converges rapidly and Clenshaw-Curtis quadrature is advantageous. One variant of the Gauss quadrature with nested sampling points is the Gauss-Kronrod scheme.

A numerically stable method to compute these quantities in given in (Schwarz, 2004, Ch. 7.4). The location of the Gauss points in the interval $[-1;1]$ is symmetric to the origin and from Eq. (4.166) it is obvious that a symmetric pair of integration points share the same weight.[21]

The extension to higher dimensions is straightforward by the tensor product of the one-dimensional Gauss points and weights as indicated in Eq. (4.162). The tensor product construction allows us to choose the integration orders n_ξ, n_η, n_ζ in each local direction $\xi, 2, \zeta$ differently yielding corresponding weights w_j, w_k, w_l for the Gauss point $\boldsymbol{\xi}_{jkl}^T = \{\xi_j, \eta_k, \zeta_l\}$.

An alternative integration approach well suited for high-order shape functions is the vector integration introduced by Hinnant (Hinnant, 1994). This method was implemented in the p-version code AdhoC and augmented by an adaptive component, cf. (Nübel et al., 2000, 2001), (Nübel et al., 2007, Sec. 4.6.2). Melenk et al. introduced a different approach in (Melenk et al., 2001) for the fast quadrature of hp elements.

Accuracy A general demand, cf. (Szabó and Babuška, 1991, p. 116), is to select the quadrature rule such that the integration error is not greater than the discretization error, compare Sec. 4.4. When computing quasi static problems, we must scrutinize the integration of the element stiffness matrix. We focus on the expression for the elemental stiffness matrix given on Gauss-point level in its geometric, Eq. (4.202), and material component, Eq. (4.207) that will be derived later. For the current investigation of integration error we can distinguish in symbolic notation four components at a given Gauss point

$$\mathbf{k}^e = \sum w \times dN \times C \times dN \times J\,, \tag{4.167}$$

the scalar weights w and a scalar entry C either representing an entry in the consistent tangent operator $\tilde{\mathbf{C}}_{\mathrm{L}}^e(\boldsymbol{\xi}_{jkl})$ or stress value in $\widehat{\tilde{\mathbf{T}}}(\boldsymbol{\xi}_{jkl})$. Here we assume that the entries of the tangent operator describing the material model evaluate the scalars for the Gauss point in question. As scalars do not change the polynomial order we now discuss the integration of $dN \times dN \times J$, the product of the Jacobian determinant with two derivatives of shape functions (representing either an entry in the element strain-displacement matrix $\mathbf{B}$ (4.86) or the gradient-displacement matrix $\mathbf{G}$ (4.76). As those derivatives are with respect to global, world space coordinates, the inverse of the Jacobian matrix is present in each component, compare Eq. (4.78).

Only under the far-reaching restriction of *affine mapping* we can develop rules of thumb for exact integration. Affine mapping from the standard element, ξ, η-coordinates, to the element in global X_1, X_2-coordinates (reference configuration) has the property, that

- parallel lines remain parallel and
- the partition ratio of three points is invariant under affine mapping,

compare (Li et al., 1993, p. 6) and (Weisstein, 2004). It is interesting to note that only a subset of the maps that are possible applying bilinear mapping, see Eq. (4.33), are affine maps. Affine maps in the defined sense have the property, that the Jacobian matrix and its determinant is constant, i.e. is no function of the coordinates ξ, η. Fig. 4.16 shows some shapes the standard quadrilateral $\Omega_{\mathrm{st}}^{\mathrm{q}}$ can transform to under bilinear mapping. Only the grey

[21] Abscissas and weight factors up to order $n = 10$ for Gauss-Legendre integration are compiled in (Szabó and Babuška, 1991, p. 356) and up to $n = 50$ in the source code of AdhoC in the file `Fem/set_Gauss.c`.

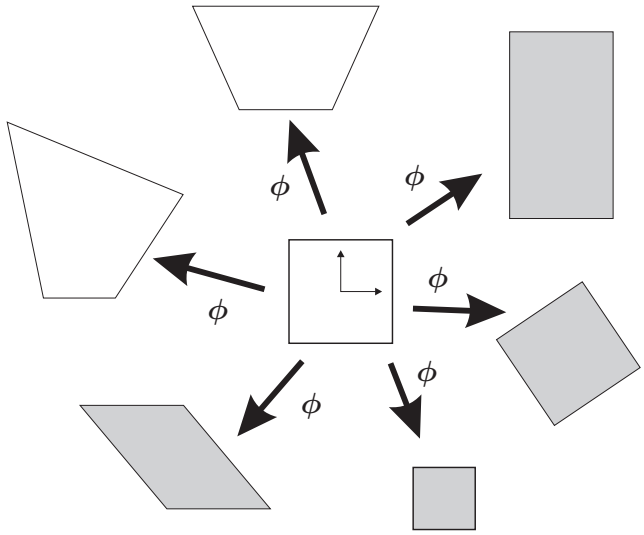


Figure 4.16: Bilinear mapping is a superset of affine maps

shapes are a product of an affine map with constant Jacobian, the white ones do not preserve the parallelism of lines. There the Jacobian is a linear function of the local coordinates ξ, η as can be easily verified if the Jacobian for Eq. (4.33) is calculated.

If an affine map transforms the standard element to the element in the reference configuration, the influence of the inverse of the Jacobian, cf. Eq. (4.78), results only in scalar factors that can be interpreted as stretching. Then, the derivatives of the mapped shape functions with respect to *global* coordinates are still polynomials of the given order. As shape functions of order p are used, the matrices represented by dN in Eq. (4.167) contain derivatives of order $p-1$ in both coordinates directions. The product $dN \times dN$ is the individual product resulting from the matrix pairing $\mathbf{B}^T$ meeting $\mathbf{B}$ or $\mathbf{G}^T$ to $\mathbf{G}$ in Eq. (4.202) and Eq. (4.207) respectively. There, derivatives in different coordinate directions may meet. In the 'worst' case an entry that is derived in X_1-direction is multiplied with an entry derived in X_2-direction, i.e. the product is of order $2p$, in the 'best' case of order $2(p-1) = 2p-2$. According to Eq. (4.165) the Gauss quadrature needs to be of order

$$n \geq \frac{(2p-2)+1}{2} = p - \frac{1}{2} \tag{4.168}$$

for exact integration in the best case and

$$n \geq \frac{(2p)+1}{2} = p + \frac{1}{2} \tag{4.169}$$

in the worst case.

Both conditions are fulfilled if the integration order is chosen to be $n = p+1$ and integration is exact in this case of affine mapping and constant material data, cf. (Demkowicz, 2006, Sec. 10.5.2). In the higher dimensional case with affine mapping it is also enough to use $p+1$ integration points for each spatial direction as the shape functions are the tensor product of one-dimensional shape functions of order p, even in the case of the tensor product space as can be seen from Pascal's triangle, Fig. 4.6.

If, however, the mapping is *not affine*, the transformation from the local ξ, η coordinate system to the global X_1, X_2 system may introduce via the inverse of the Jacobian, see Eq. (4.78), rational fractions or general non-polynomial contributions like trigonometric functions. Thus, these integrands can not be integrated exactly by Gauss quadrature any more. Then, there exists no general rule which integration order to choose for sufficient accuracy as the smoothness of the problem has to be taken into account. From numerical experiments it is suggested to use an integration order of $n = p + 2$, or better $n = p + 3$, in each spatial direction (Bröker, 2001, Ch. 3.4.3) (Graeff-Weinberg, 1995). The commercial p-version code StressCheck (Engineering Software Research & Development, 2006) uses an integration order of $p + 3$. A mathematical investigation of this issue was carried out by Banerjee and Suri, (Banerjee and Suri, 1992a,b) and numerical one- and two-dimensional examples are discussed by Kim and Suri (Kim, 1998; Kim and Suri, 1993) indicating that in pathological cases the margin of safety provided by an integration order of $p+3$ is not sufficient to reach the expected order of convergence in L_2 and H_1 norm.

After this excursion we perform the numerical integration exemplarily for the internal virtual work, Eq. (4.103),

$$\boldsymbol{g}(t,\boldsymbol{u}(t),\boldsymbol{q}(t)) \approx \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}^{eT} \left\{ \sum_{j=1}^{n_\xi} \sum_{k=1}^{n_\eta} \sum_{l=1}^{n_\zeta} w_j w_k w_l \, \mathbf{B}^{eT}\big(\mathbf{u}^e(t), \boldsymbol{\xi}_{jkl}\big) \right. \tag{4.170}$$
$$\left. \tilde{\Phi}\Big(\mathbf{C}^e\big(\mathbf{E}^e(\boldsymbol{\xi}_{jkl},t)\big), \mathbf{q}^e\big(\mathbf{C}^e(\mathbf{E}^e(\boldsymbol{\xi}_{jkl},t))\big)\Big) \det\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right\} - \boldsymbol{f}_\mathrm{ext}(t,\boldsymbol{u}) = \boldsymbol{0}\,.$$

The term in curly brackets is the numerical approximation of the elemental internal force vector $\mathbf{f}^e_\mathrm{int}$. The number of integration points in element e is $n^e_\mathrm{i} = n_\xi \times n_\eta \times n_\zeta$ and the number of integration points of the whole structure amounts to $n_\mathrm{i} = \sum_{e=1}^{n_\mathrm{e}} n^e_\mathrm{i}$, as the number of integration points may vary from element to element. Analogous derivations can be carried out for all contributions in the linearized weak form Eq. (4.101) by evaluating the quantities at the Gauss points $\boldsymbol{\xi}_{jkl}$ and summing with the corresponding weights.

4.6.5 DAE system and time discretization

As a consequence of the numerical integration the internal variables $\mathbf{q}^{e(jkl)}(t) \in \mathbb{R}^{n_\mathrm{q}}$, $\mathbf{q}^{e(jkl)}(t) = \mathbf{q}^e\big(\mathbf{C}^e(\mathbf{E}^e(\boldsymbol{\xi}_{jkl},t))\big)$ need to be available at the quadrature points $\boldsymbol{\xi}_{jkl}$. This necessitates the evaluation of the evolution equations (4.64) of the constitutive model at the n_i integration points of the structure

$$\mathbf{A}\dot{\mathbf{q}}^{e(jkl)}(t) - \mathbf{r}(\mathbf{E}^e(\boldsymbol{\xi}_{jkl},t), \mathbf{q}^{e(jkl)}(t)) = \mathbf{0}, \quad \mathbf{q}^{e(jkl)}(t_0) = \mathbf{q}_0^{e(jkl)}. \tag{4.171}$$

The dependency of the internal variables on the unknown displacements $\boldsymbol{u}(t)$ and the time-dependent prescribed displacements $\overline{\boldsymbol{u}}(t)$ can be clearly seen from equation (4.94).

As we introduced coincidence matrices for the transition from elemental to global displacements in Eq. (4.148) we do a similar process for the internal variables

$$\mathbf{q}^{e(jkl)}(t) = \boldsymbol{Z}_{\boldsymbol{q}}^{e(jkl)} \boldsymbol{q}(t), \quad \mathbf{q}^{e(jkl)} \in \mathbb{R}^{n_\mathrm{q}} \tag{4.172}$$

where the incidence matrix $\boldsymbol{Z}_{\boldsymbol{q}}^{e(jkl)} \in \mathbb{R}^{n_\mathrm{q} \times n_\mathrm{Q}}$ is used to obtain the n_q internal variables $\mathbf{q}^{e(jkl)}$ at Gauss point $\boldsymbol{\xi}_{jkl}$ in element e. We note that the internal variables and their evolution equations

are independent of each other and depend only on quantities at the specific integration point. Mathematically this is described by the property

$$\boldsymbol{Z}_{\boldsymbol{q}}^{e(jkl)}\boldsymbol{Z}_{\boldsymbol{q}}^{\hat{e}(\hat{j}\hat{k}\hat{l})T} = \begin{cases} \mathbf{I}_{n_\mathrm{q}} & \text{if } e=\hat{e} \text{ and } j=\hat{j} \text{ and } k=\hat{k} \text{ and } l=\hat{l} \\ \mathbf{0} & \text{else} \end{cases} \tag{4.173}$$

with the identity matrix $\mathbf{I}_{n_\mathrm{q}} \in \mathbb{R}^{n_\mathrm{q}\times n_\mathrm{q}}$. Contrary to the displacements where multiple element degrees of freedom (nodal and edge modes) may correspond to the same entry in the global displacement vector, for the internal variables we have a one-to-one relation as a consequence of property (4.173) and we can assemble all evolution equations of all internal variables into a system of ordinary differential equations, ODE, (if $\mathbf{A} = \mathbf{I}$, see (2.48)) or a differential algebraic system, DAE, for constitutive models with yield condition. With a number of n_q internal variables (for example 7) of the chosen constitutive model at each quadrature point all internal variables $n_\mathrm{Q} = n_\mathrm{i} \times n_\mathrm{q}$ at time t can be stored in a global vector of internal variables $\boldsymbol{q}(t) \in \mathbb{R}^{n_\mathrm{Q}}$. Then Eq. (4.171) transforms to an n_Q-dimensional initial value problem for the complete structure

$$\boldsymbol{A}\dot{\boldsymbol{q}}(t) - \boldsymbol{r}\big(t, \boldsymbol{u}(t), \boldsymbol{q}(t)\big) = \boldsymbol{0}, \qquad \boldsymbol{q}(t_0) = \boldsymbol{q}_0\,. \tag{4.174}$$

To obtain the unknown displacements $\boldsymbol{u}(t) \in \mathbb{R}^{n_\mathrm{u}}$ and all internal variables $\boldsymbol{q}(t) \in \mathbb{R}^{n_\mathrm{Q}}$ we have to solve the discretized variational principle (4.170) and the evolution equations (4.174). Together they form a system of differential algebraic equations

$$\boldsymbol{F}(t, \boldsymbol{y}(t), \dot{\boldsymbol{y}}(t)) \equiv \left\{ \begin{array}{c} \boldsymbol{g}(t, \boldsymbol{u}(t), \boldsymbol{q}(t)) \\ \boldsymbol{A}\dot{\boldsymbol{q}}(t) - \boldsymbol{r}\big(t, \boldsymbol{u}(t), \boldsymbol{q}(t)\big) \end{array} \right\} = \boldsymbol{0}, \quad \boldsymbol{F} \in \mathbb{R}^{n_\mathrm{u}+n_\mathrm{Q}} \tag{4.175}$$

where the functions $\boldsymbol{y}(t) \in \mathbb{R}^{n_\mathrm{u}+n_\mathrm{Q}}$

$$\boldsymbol{y}(t) \equiv \left\{ \begin{array}{c} \boldsymbol{u}(t) \\ \boldsymbol{q}(t) \end{array} \right\} \quad \text{with the initial conditions } \boldsymbol{y}(t_0) \equiv \left\{ \begin{array}{c} \boldsymbol{u}(t_0) \\ \boldsymbol{q}(t_0) \end{array} \right\} = \left\{ \begin{array}{c} \boldsymbol{u}_0 \\ \boldsymbol{q}_0 \end{array} \right\} \equiv \boldsymbol{y}_0 \tag{4.176}$$

are introduced.

Usually this formal derivation leading to the DAE-system (4.175), compare the work of (Ellsiepen and Hartmann, 2001) based on (Fritzen, 1997), is not carried out in the literature of finite elements. However, it clarifies the mathematical structure of the problem and gives the possibility to identify methods developed in numerical mathematics that have not necessarily an intuitive equivalent. Higher order methods for time integration are naturally suited for time adaptivity and error estimation as more accurate higher order results can be compared to lower order values. The intuitive approach to time integration is to increase the external loads or prescribed displacements step by step and to solve the evolution of the internal variables at the Gauss points, Eq. (4.171), by more or less simple time integration methods like backward Euler. For the solution of semi-explicit nonlinear block-structured DAE-systems like Eq. (4.175) resulting from a space discretization (for example by finite elements) there exist more sophisticated possibilities. The time discretization can be done with one-step or multi-step methods. One-step methods only rely on the information of one previous time step to determine the current value whereas multi-step methods take the information of more than the last time step into account. An example of an one-step method is the stiffly accurate diagonally implicit Runge-Kutta (DIRK) method that contains the traditional backward

Euler method as special case. DIRK methods in the context of finite elements are for example discussed in (Hartmann, 2003a, Sec. 2.3),(Ellsiepen and Hartmann, 2001),(Hartmann, 2002), (Hartmann and Bier, 2008). Another approach is to use Rosenbrock-type methods as investigated in (Hartmann and Wensch, 2007).

For the remainder of the text and the implementation we apply the backward Euler scheme for time discretization, i.e. a DIRK method with only one stage, $s = 1$. Here the boundary conditions $\overline{\boldsymbol{u}}(t), \boldsymbol{f}_{\text{ext}}(t)$ are prescribed for the time $t = t_{n+1}$ and after time-discretizing, the nonlinear equation system Eq. (4.175) for the time step from t_n to t_{n+1} takes the form

$$\boldsymbol{R}_{n+1}\left(\boldsymbol{u}_{n+1}, \boldsymbol{q}_{n+1}\right) = \boldsymbol{R}_{n+1}\left(\boldsymbol{y}_{n+1}\right) = \left\{ \begin{array}{c} \boldsymbol{G}_{n+1}(\boldsymbol{u}_{n+1}, \boldsymbol{q}_{n+1}) \\ \boldsymbol{L}_{n+1}(\boldsymbol{u}_{n+1}, \boldsymbol{q}_{n+1}) \end{array} \right\} = \boldsymbol{0}, \tag{4.177}$$

$\boldsymbol{R} \in \mathbb{R}^{n_{\text{u}}+n_{\text{Q}}}$, with the unknowns

$$\boldsymbol{y}_{n+1} = \left\{ \begin{array}{c} \boldsymbol{u}_{n+1} \\ \boldsymbol{q}_{n+1} \end{array} \right\} \tag{4.178}$$

and the abbreviations for the equations (4.170) and (4.171) that are discretized and evaluated at time t_{n+1}

$$\begin{aligned} \boldsymbol{G}_{n+1}(\boldsymbol{u}_{n+1}, \boldsymbol{q}_{n+1}) \;\equiv\; & \sum_{e=1}^{n_{\text{e}}} \boldsymbol{Z}^{eT} \left\{ \sum_{j=1}^{n_\xi} \sum_{k=1}^{n_\eta} \sum_{l=1}^{n_\zeta} w_j w_k w_l \mathbf{B}^{eT}\left(\mathbf{u}^e_{n+1}, \boldsymbol{\xi}_{jkl}\right) \right. \\ & \left. \tilde{\Phi}\left(\mathbf{E}^{e(jkl)}_{n+1}, \mathbf{q}^{e(jkl)}_{n+1}\right) \det \mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right\} - \boldsymbol{f}_{\text{ext}}(t_{n+1}, \boldsymbol{u}_{n+1}) = \boldsymbol{0} \end{aligned} \tag{4.179}$$

$$\boldsymbol{L}_{n+1}(\boldsymbol{u}_{n+1}, \boldsymbol{q}_{n+1}) \;\equiv\; \sum_{e=1}^{n_{\text{e}}} \left\{ \sum_{j=1}^{n_\xi} \sum_{k=1}^{n_\eta} \sum_{l=1}^{n_\zeta} \boldsymbol{Z}^{e(jkl)T}_{\boldsymbol{q}} \, \mathbf{L}^{e(jkl)}_{n+1}\left(\mathbf{E}^{e(jkl)}_{n+1}, \mathbf{q}^{e(jkl)}_{n+1}\right) \right\} = \boldsymbol{0}\,. \tag{4.180}$$

The local equation system at Gauss point $\boldsymbol{\xi}_{jkl}$ is given as

$$\mathbf{L}^{e(jkl)}_{n+1}\left(\mathbf{E}^{e(jkl)}_{n+1}(\mathbf{u}^e_{n+1}), \mathbf{q}^{e(jkl)}\right) \equiv \mathbf{A} \left\{ \frac{\mathbf{q}^{e(jkl)}_{n+1} - \mathbf{q}^{e(jkl)}_{n}}{\Delta t_n} \right\} - \mathbf{r}\left(\mathbf{E}^{e(jkl)}_{n+1}(\mathbf{u}^e_{n+1}), \mathbf{q}^{e(jkl)}\right) = \mathbf{0} \tag{4.181}$$

and the strain at the quadrature point depends via the $\mathbf{B}^{*e}$ matrix (4.151) also on the unknown and prescribed displacements

$$\mathbf{E}^{e(jkl)}_{n+1} = \mathbf{B}^{*e(jkl)} \mathbf{U}^e_{n+1} \text{ where } \mathbf{U}^e_{n+1} = \boldsymbol{Z}^e \boldsymbol{u}_{n+1} + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}}(t_{n+1}) \tag{4.182}$$

with $\mathbf{U}^e_{n+1} \in \mathbb{R}^{n_{\text{em}}}, \boldsymbol{U}_{n+1} \in \mathbb{R}^{n_{\text{modes}}}$.

4.7 Solution of the global system

Common methods for the solution of nonlinear equation systems in the context of the finite element method are, (Wriggers, 2001, Ch. 5),

- fixed point iteration,
- Newton-Raphson procedures,

- quasi-Newton procedures,
- dynamic relaxation,
- path following methods.

One of the most popular algorithms for the solution of coupled nonlinear systems of the type Eq. (4.177) is the Newton-Raphson method in its different flavors with the property of quadratic convergence in the vicinity of the solution. In the finite element literature following (Simo and Taylor, 1985) it is generally claimed that the equation system (4.177) resulting from the space and time discretization of a constitutive model with internal variables is solved by the Newton-Raphson method. Connected to this is a local iteration ('stress algorithm') and the so called consistent tangent operator necessary to attain quadratic convergence.

Science as a system of knowledge relies on clear notions, (Keller, 1989), so it is advisable to follow (Hartmann, 2005) and distinguish for problems with internal variables two solution procedures of Newton type:

- the classical Newton-Raphson Method (NRM) that yields a consistent tangent operator but no local iteration,

- and the Multi-Level Newton algorithm (MLNA) that is precisely what is usually applied in current finite element codes with a local stress algorithm. An example is the radial return method in elastoplasticity, that iteratively solves the evolution equations for the updated internal variables needed for the stress computation and a consistent tangent operator.

The consistent tangent operator is similar but different in detail for both variants. For problems of nonlinear *elasticity*, even for complex hyperelastic constitutive models, where the stress state is solely determined by the strain state (no internal variables) there is no local iteration and indeed the classical NRM is used. However, finite element codes used for the solution of constitutive equations of evolutionary type exhibit a two-level structure and implicitly apply the MLNA. In the FE community the pioneering work of (Simo and Taylor, 1985) numerically showed that quadratic convergence can be achieved by a concept they called 'consistent linearization'. It is interesting to note that the quadratic convergence of the multi-level Newton algorithm was already proved analytically by (Rabbat et al., 1979) in the context of the computation of nonlinear electric networks some years before.

The classical Newton method that was introduced in Sec. 3.2 based on the linearization (3.22) of a function is adapted here to the system (4.177), $\boldsymbol{R}(\boldsymbol{y}) = \boldsymbol{0}, \boldsymbol{y} \in \mathbb{R}^n$, (we omit the index $n+1$),

$$\mathsf{L}\left[\boldsymbol{R}(\boldsymbol{y})\right]_{\boldsymbol{y}=\boldsymbol{y}_0} = \boldsymbol{R}(\boldsymbol{y}_0) + \mathsf{D}_{\boldsymbol{y}}(\boldsymbol{R}(\boldsymbol{y}_0))[\Delta\boldsymbol{y}] = \boldsymbol{R}(\boldsymbol{y}_0) + \left.\frac{\mathrm{d}\boldsymbol{R}}{\mathrm{d}\boldsymbol{y}}\right|_{\boldsymbol{y}=\boldsymbol{y}_0} \Delta\boldsymbol{y} \tag{4.183}$$

with the increment $\Delta\boldsymbol{y} = \boldsymbol{y} - \boldsymbol{y}_0$. From an initial state $\boldsymbol{y}_0$ we start an iterative process with iteration index $(\bullet)^{(m)}$ to obtain the increment $\Delta\boldsymbol{y} = \boldsymbol{y}^{(m+1)} - \boldsymbol{y}^{(m)}$ that, we hope, brings us nearer to the solution

$$\boldsymbol{R}\left(\boldsymbol{y}^{(m)}\right) + \left.\frac{\mathrm{d}\boldsymbol{R}}{\mathrm{d}\boldsymbol{y}}\right|_{\boldsymbol{y}=\boldsymbol{y}^{(m)}} \Delta\boldsymbol{y} = \boldsymbol{0}. \tag{4.184}$$

Moving the residuum $\boldsymbol{R}\left(\boldsymbol{y}^{(m)}\right)$ to the right-hand side we have to solve a system of linear equations in each iteration to obtain $\Delta\boldsymbol{y}$

$$\left[\frac{\mathrm{d}\boldsymbol{R}}{\mathrm{d}\boldsymbol{y}}\right]\Bigg|_{\boldsymbol{y}=\boldsymbol{y}^{(m)}} \Delta\boldsymbol{y} = -\boldsymbol{R}\left(\boldsymbol{y}^{(m)}\right). \tag{4.185}$$

The procedure can be stopped if the norm of the increment $\|\Delta\boldsymbol{y}\|$ or the norm of the residuum $\|\boldsymbol{R}\left(\boldsymbol{y}^{(m)}\right)\|$ is below a certain threshold, then we have obtained an approximate solution to $\boldsymbol{R}(\boldsymbol{y}) \approx \boldsymbol{0}$. A discussion of convergence criteria in the finite element context can be found in (Zienkiewicz and Taylor, 2000a, Sec. 2.2.7) and (Belytschko et al., 2000, Sec. 6.3.9).

In the following we discuss two distinct methods for solving the DAE system. For a compact notation of the block structured system Eq. (4.177) we drop the index $n+1$ and introduce the abbreviations $\boldsymbol{U} \equiv \boldsymbol{u}_{n+1}, \boldsymbol{Q} \equiv \boldsymbol{q}_{n+1}, \boldsymbol{G} \equiv \boldsymbol{G}_{n+1}, \boldsymbol{L} \equiv \boldsymbol{L}_{n+1},$ and $\boldsymbol{R} \equiv \boldsymbol{R}_{n+1}$

$$\boldsymbol{R}(\boldsymbol{y}) = \left\{ \begin{array}{c} \boldsymbol{G}(\boldsymbol{U},\boldsymbol{Q}) \\ \boldsymbol{L}(\boldsymbol{U},\boldsymbol{Q}) \end{array} \right\} = \boldsymbol{0}, \quad \boldsymbol{y} = \left\{ \begin{array}{c} \boldsymbol{U} \\ \boldsymbol{Q} \end{array} \right\}. \tag{4.186}$$

4.7.1 Newton-Raphson method (NRM)

Applying the classical Newton-Raphson procedure (4.185) to the block structured system (4.186) we obtain for iteration m

$$\left[\begin{array}{cc} \dfrac{\partial \boldsymbol{G}}{\partial \boldsymbol{U}} & \dfrac{\partial \boldsymbol{G}}{\partial \boldsymbol{Q}} \\ \dfrac{\partial \boldsymbol{L}}{\partial \boldsymbol{U}} & \dfrac{\partial \boldsymbol{L}}{\partial \boldsymbol{Q}} \end{array} \right]\Bigg|_{\boldsymbol{y}} \left\{ \begin{array}{c} \Delta\boldsymbol{U} \\ \Delta\boldsymbol{Q} \end{array} \right\} = - \left\{ \begin{array}{c} \boldsymbol{L}(\boldsymbol{y}) \\ \boldsymbol{G}(\boldsymbol{y}) \end{array} \right\}, \text{with } \boldsymbol{y} \equiv \left\{ \begin{array}{c} \boldsymbol{U}^{(m)} \\ \boldsymbol{Q}^{(m)} \end{array} \right\} \tag{4.187}$$

and the increments $\Delta\boldsymbol{U} = \boldsymbol{U}^{(m+1)} - \boldsymbol{U}^{(m)}$, $\Delta\boldsymbol{Q} = \boldsymbol{Q}^{(m+1)} - \boldsymbol{Q}^{(m)}$.

The iterative solution of the system (4.187) corresponds to the application of the classical Newton-Raphson method (NRM). This may be solved in one step for the vector of all unknowns $\{\Delta\boldsymbol{U}, \Delta\boldsymbol{Q}\}^T$, see e.g. (Christensen, 2000), or the block-structure is exploited by solving the second equation in (4.187) for $\Delta\boldsymbol{Q}$ and inserting in the first equation, so $\Delta\boldsymbol{U}$ is the remaining unknown and the associated coefficient matrix is a tangential stiffness matrix. The numerical implications of this approach yielding a consistent tangent operator but no local iteration are described in detail in (Hartmann, 2005).

4.7.2 Multi-level Newton algorithm (MLNA)

We now focus on the multi-level Newton algorithm that is used in current finite element implementations for the solution of the coupled system (4.186). To this end the implicit function theorem[22], (Krantz and Parks, 2003), is applied to Eq. $(4.186)_2$. This theorem states that in the neighborhood of a solution $\boldsymbol{y}^* = \{\boldsymbol{U}^*, \boldsymbol{Q}^*\}$ of $\boldsymbol{L}(\boldsymbol{y}) = \boldsymbol{0}$ there exists an implicitly defined differentiable function $\boldsymbol{Q}(\boldsymbol{U})$ in $\boldsymbol{U}$ satisfying $(4.186)_2$. This function is inserted into Eq. $(4.186)_1$ and we obtain the system

$$\boldsymbol{G}\big(\boldsymbol{U}, \boldsymbol{Q}(\boldsymbol{U})\big) = \boldsymbol{0}. \tag{4.188}$$

[22]For a demonstrative, yet detailed presentation see (Implicit Function Theorem, 2007).

For the solution of this nonlinear equation we linearize in view of the Newton-Raphson method, viz Eq. (4.183), and shift the known terms to the right-hand side to obtain

$$\mathsf{D}_{\boldsymbol{U}}\boldsymbol{G}\big(\boldsymbol{U},\boldsymbol{Q}(\boldsymbol{U})\big)[\Delta\boldsymbol{U}] = -\boldsymbol{G}(\boldsymbol{y})\,, \tag{4.189}$$

where the directional derivative defines the tangential stiffness matrix $\boldsymbol{K}|_{\boldsymbol{y}}$ of the MLNA

$$\mathsf{D}_{\boldsymbol{U}}\boldsymbol{G}\big(\boldsymbol{U},\boldsymbol{Q}(\boldsymbol{U})\big)[\Delta\boldsymbol{U}] = \left[\frac{\partial\boldsymbol{G}}{\partial\boldsymbol{U}} + \frac{\partial\boldsymbol{G}}{\partial\boldsymbol{Q}}\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}\right]\Bigg|_{\boldsymbol{y}} \Delta\boldsymbol{U} = \boldsymbol{K}|_{\boldsymbol{y}}\,\Delta\boldsymbol{U}\,. \tag{4.190}$$

This is explicated at Gauss point level in (4.197) – (4.208) yielding also the consistent tangent operator or material matrix (4.208).

Consequently the equation system corresponding to Eq. (4.185) is

$$\left[\frac{\partial\boldsymbol{G}}{\partial\boldsymbol{U}} + \frac{\partial\boldsymbol{G}}{\partial\boldsymbol{Q}}\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}\right]\Bigg|_{\boldsymbol{y}} \Delta\boldsymbol{U} = -\boldsymbol{G}(\boldsymbol{y}) \tag{4.191}$$

where $\Delta\boldsymbol{U} = \boldsymbol{U}^{(m+1)} - \boldsymbol{U}^{(m)}$ is the vector of modal displacement increments and $\boldsymbol{y} \equiv \{\boldsymbol{U}^{(m)}, \boldsymbol{Q}^{(m)}\}$ the state corresponding to iteration m.

The implicit function theorem states the existence of the function $\boldsymbol{Q}(\boldsymbol{U})$ and rules for the derivative, but no explicit formulation. To evaluate $\boldsymbol{G}(\boldsymbol{y})$ in (4.191), given in (4.179), we need to calculate the internal variables $\boldsymbol{Q}^{(m)} = \boldsymbol{Q}(\boldsymbol{U}^{(m)})$ iteratively for given displacements $\boldsymbol{U}^{(m)}$ from the evolution equations. This is the so-called local integration step

$$\boldsymbol{L}(\boldsymbol{U}^{(m)}, \boldsymbol{Q}^{(m)}) = \boldsymbol{0} \qquad \rightsquigarrow \boldsymbol{Q}^{(m)}\,. \tag{4.192}$$

Besides $\boldsymbol{Q}^{(m)}$ the derivative $\mathrm{d}\boldsymbol{Q}/\mathrm{d}\boldsymbol{U}$ of the implicit function $\boldsymbol{Q}(\boldsymbol{U})$ is needed in the tangential stiffness matrix (4.190). The implicit function $\boldsymbol{Q}(\boldsymbol{U})$ is defined in a way that the equation $(4.186)_2$ is identically satisfied, hence,

$$\boldsymbol{L}\big(\boldsymbol{U},\boldsymbol{Q}(\boldsymbol{U})\big) = \boldsymbol{0}. \tag{4.193}$$

Deriving this identity with respect to $\boldsymbol{U}$, see e.g. (Meyberg and Vachenauer, 2003, p. 399), the chain rule gives

$$\frac{\partial\boldsymbol{L}}{\partial\boldsymbol{U}}\bigg|_{\boldsymbol{y}} + \frac{\partial\boldsymbol{L}}{\partial\boldsymbol{Q}}\bigg|_{\boldsymbol{y}} \frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}\bigg|_{\boldsymbol{y}} = \boldsymbol{0} \tag{4.194}$$

and we can rearrange to obtain an equation system with several right-hand sides to evaluate the derivative $\mathrm{d}\boldsymbol{Q}/\mathrm{d}\boldsymbol{U}$

$$\left[\frac{\partial\boldsymbol{L}}{\partial\boldsymbol{Q}}\right]\Bigg|_{\boldsymbol{y}} \frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}\bigg|_{\boldsymbol{y}} = -\frac{\partial\boldsymbol{L}}{\partial\boldsymbol{U}}\bigg|_{\boldsymbol{y}}\,. \tag{4.195}$$

The MLNA procedure is summarized in Tab. 4.2 taken from (Hartmann, 2005) and adapted (simplified) to the backward Euler time integration procedure, i.e. an implicit Runge-Kutta method with one stage.

Table 4.2: Multilevel-Newton algorithm at time t_{n+1}, taken from (Hartmann, 2005) and simplified for backward-Euler time integration

Given: $\boldsymbol{U}_{n+1}^{(0)} = \boldsymbol{u}_n$, $\boldsymbol{Q}_{n+1}^{(0)} = \boldsymbol{q}_n$, Δt_n		
Repeat	$m = 0, \ldots$	
	local level (given: $\boldsymbol{U}_{n+1}^{(m)}$, argument vector $\boldsymbol{y} := (\boldsymbol{U}_{n+1}^{(m)}, \boldsymbol{Q}_{n+1}^{(m)})$)	
	local integration step	
	$\boldsymbol{L}(\boldsymbol{U}_{n+1}^{(m)}, \boldsymbol{Q}_{n+1}^{(m)}) = \boldsymbol{0}$	$\rightsquigarrow \boldsymbol{Q}_{n+1}^{(m)}$
	consistent linearization	
	$\left[\left.\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{Q}}\right\|_{\boldsymbol{y}}\right] \left.\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}\right\|_{\boldsymbol{y}} = -\left.\frac{\partial \boldsymbol{L}}{\partial \boldsymbol{U}}\right\|_{\boldsymbol{y}}$	$\rightsquigarrow \left.\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}\right\|_{\boldsymbol{y}}$
	global level	
	solve linear system of equations	
	$\left[\left.\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{U}}\right\|_{\boldsymbol{y}} + \left.\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Q}}\right\|_{\boldsymbol{y}} \left.\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}\right\|_{\boldsymbol{y}}\right] \Delta \boldsymbol{U}_{n+1} = -\boldsymbol{G}(\boldsymbol{y})$	$\rightsquigarrow \Delta \boldsymbol{U}_{n+1}$
	Update of global variables	
	$\boldsymbol{U}_{n+1}^{(m+1)} \leftarrow \boldsymbol{U}_{n+1}^{(m)} + \Delta \boldsymbol{U}_{n+1}$	$\rightsquigarrow \boldsymbol{U}_{n+1}^{(m+1)}$
Until	the convergence criterion is fulfilled	

4.7.3 Evaluation on element level

The global representation of the MLNA for the solution of the coupled nonlinear DAE-system, as given in Tab. 4.2, has to be completed with the actual representation on element and Gauss point level.

First we focus on the calculation of the internal variables, Eq. (4.193). As the evolution equations are decoupled, cf. (4.173), the local nonlinear equation (4.181) at Gauss point $\boldsymbol{\xi}_{jkl}$,

$$\mathbf{L}^{e(jkl)}(\mathbf{U}^e, \mathbf{Q}^{e(jkl)}) = \mathbf{0} \tag{4.196}$$

for given $\mathbf{U}^e$ is iteratively solved to obtain the internal variables $\mathbf{Q}^{e(jkl)}$. This local iteration is also called stress algorithm as the stresses can be evaluated from Eq. (4.62) once the internal variables are known. As this iterative procedure has to be carried out for all n_i integration points of the structure this can be quite expensive. Therefore, it is very beneficial if the equation system can be simplified or even solved analytically. These problem adapted optimized stress algorithms (see for example in the case of elastoplasticity (Simo and Taylor, 1985), (Nübel, 2005), (Düster, 2001) and visco-plasticity (Hartmann et al., 1997)) improve efficiency but disguise the general structure of the global multi-level procedure.

Now the tangential stiffness matrix is traced to element and Gauss point level. Restricting

ourself to the contribution of the internal virtual work (the follower stiffness is treated accordingly) we can build on the derivations leading to the representation of the internal virtual work on Gauss point level, Eq. (4.179). The directional derivative of this part

$$\mathbf{G}_{\text{int}}^{e(jkl)} = \mathbf{B}^{e(jkl)T}(\mathbf{U}^e)\,\tilde{\mathbf{\Phi}}^{(jkl)}\Big(\mathbf{E}^{e(jkl)}(\mathbf{U}^e), \mathbf{Q}^{e(jkl)}(\mathbf{E}^{e(jkl)}(\mathbf{U}^e))\Big) \tag{4.197}$$

in the direction of the displacements needs to utilize the product rule as the $\mathbf{B}^e$ matrix in finite strain analysis also depends on the deformation and the chain rule for the dependency of the local displacements $\mathbf{U}^e$ on the global $\boldsymbol{U}$, (4.150). For the global picture see (4.190). So we arrive at

$$\begin{aligned}
&\mathsf{D}_{\boldsymbol{U}}\mathbf{G}_{\text{int}}^{e(jkl)}[\Delta\boldsymbol{U}] = && (4.198)\\
&\quad = \mathsf{D}_{\mathbf{U}^e}\mathbf{B}^{e(jkl)T}(\mathbf{U}^e)[\mathsf{D}_{\boldsymbol{U}}\mathbf{U}^e[\Delta\boldsymbol{U}]]\,\tilde{\mathbf{\Phi}}^{(jkl)} && (4.199)\\
&\quad + \mathbf{B}^{e(jkl)T}\Big(\mathsf{D}_{\boldsymbol{U}}\tilde{\mathbf{\Phi}}^{(jkl)}\big(\mathbf{E}^{e(jkl)}(\mathbf{U}^e), \mathbf{Q}^{e(jkl)}(\mathbf{E}^{e(jkl)}(\mathbf{U}^e))\big)[\Delta\boldsymbol{U}]\Big). && (4.200)
\end{aligned}$$

For the evaluation of the first term, (4.199), we introduce the definition of $\mathbf{B}^e$, (4.86) together with (4.83)-(4.84)

$$\mathsf{D}_{\boldsymbol{U}}\Big(\mathbf{H}\mathbf{G} + \mathbf{M}\big(\mathbf{G}(\mathbf{U}^e)\big)\mathbf{G}\Big)^T[\mathsf{D}_{\boldsymbol{U}}\mathbf{U}^e[\Delta\boldsymbol{U}]]\,\tilde{\mathbf{\Phi}}^{(jkl)} = \Big(\mathbf{M}\big(\mathbf{G}(\Delta\boldsymbol{U})\big)\mathbf{G}\Big)^T\boldsymbol{Z}^e\Delta\boldsymbol{U}\,\tilde{\mathbf{\Phi}}^{(jkl)} \tag{4.201}$$

where the matrix $\boldsymbol{Z}^e$ results from $\mathsf{D}_{\boldsymbol{U}}\mathbf{U}^e[\Delta\boldsymbol{U}]$, cf. (4.150). The part $\mathbf{M}\big(\mathbf{G}(\Delta\boldsymbol{U})\big)\mathbf{G}$ is the discretization of $\mathsf{D}\delta\mathbf{E}$, (3.29). Written in matrix-vector form using the same arguments as for (4.121) we obtain the familiar representation

$$\mathbf{k}_G^e = \sum_{j=1}^{n_\xi}\sum_{k=1}^{n_\eta}\sum_{l=1}^{n_\zeta} w_j w_k w_l\,\mathbf{G}^T(\boldsymbol{\xi}_{jkl})\,\widehat{\widetilde{\mathbf{T}}}(\boldsymbol{\xi}_{jkl})\,\mathbf{G}(\boldsymbol{\xi}_{jkl})\,\det\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \tag{4.202}$$

where $\widehat{\widetilde{\mathbf{T}}}$ is the matrix holding the stresses $\tilde{\mathbf{\Phi}}^{(jkl)}$ in the order of $\widehat{\widetilde{\boldsymbol{T}}}$, see (4.123)-(4.125), and the element gradient-displacement matrix $\mathbf{G}$ at point $\boldsymbol{\xi}_{jkl}$ is the counterpart of the global matrix $\boldsymbol{G}$, (4.76) with elemental shape functions. At the end of this derivation we obtained the geometric stiffness $\mathbf{k}_G^e$ as needed in (4.159).

To explicate the stiffness resulting from the nonlinear constitutive model, $\mathbf{k}_C^e$ in (4.159), we evaluate the second term of the directional derivative, (4.200), using the chain rule

$$\begin{aligned}
&\mathbf{B}^{e(jkl)T}\mathsf{D}_{\mathbf{E}^e}\tilde{\mathbf{\Phi}}^{jkl}\Big(\mathbf{E}^{e(jkl)}(\mathbf{U}^e), \mathbf{Q}^{e(jkl)}(\mathbf{E}^{e(jkl)})\Big)\Big[\mathsf{D}_{\mathbf{U}^e}\mathbf{E}^{e(jkl)}\big[\mathsf{D}_{\boldsymbol{U}}\mathbf{U}^e[\Delta\boldsymbol{U}]\big]\Big] && (4.203)\\
+\;&\mathbf{B}^{e(jkl)T}\mathsf{D}_{\mathbf{Q}}\tilde{\mathbf{\Phi}}^{jkl}\Big(\mathbf{E}^{e(jkl)}(\mathbf{U}^e), \mathbf{Q}^{e(jkl)}(\mathbf{E}^{e(jkl)})\Big)\Big[\mathsf{D}_{\mathbf{E}^e}\mathbf{Q}\Big[\mathsf{D}_{\mathbf{U}^e}\mathbf{E}^{e(jkl)}\big[\mathsf{D}_{\boldsymbol{U}}\mathbf{U}^e[\Delta\boldsymbol{U}]\big]\Big]\Big] && (4.204)\\
=\;&\mathbf{B}^{e(jkl)T}\left[\frac{\partial\tilde{\mathbf{\Phi}}}{\partial\mathbf{E}}\mathbf{B}^{e(jkl)}\boldsymbol{Z}^e\Delta\boldsymbol{U}\right] + \mathbf{B}^{e(jkl)T}\left[\frac{\partial\tilde{\mathbf{\Phi}}}{\partial\mathbf{Q}}\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}\mathbf{E}}\mathbf{B}^{e(jkl)}\boldsymbol{Z}^e\Delta\boldsymbol{U}\right] && (4.205)\\
=\;&\mathbf{B}^{e(jkl)T}\left[\frac{\partial\tilde{\mathbf{\Phi}}}{\partial\mathbf{E}} + \frac{\partial\tilde{\mathbf{\Phi}}}{\partial\mathbf{Q}}\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}\mathbf{E}}\right]\Bigg|_{\mathbf{y},\boldsymbol{\xi}_{jkl}}\mathbf{B}^{e(jkl)}\boldsymbol{Z}^e\Delta\boldsymbol{U} && (4.206)
\end{aligned}$$

where the vector $\mathbf{y} = \{\mathbf{U}^{e(m)}, \mathbf{Q}^{e(m)}\}$ corresponds to the known quantities at iteration (m). Finally, we can give the element stiffness matrix corresponding to the material nonlinearity

$$\mathbf{k}_C^e = \left[\sum_{j=1}^{n_\xi} \sum_{k=1}^{n_\eta} \sum_{l=1}^{n_\varsigma} w_j w_k w_l \, \mathbf{B}^{e(jkl)T} \, \tilde{\mathbf{C}}_{\mathrm{L}}^e(\boldsymbol{\xi}_{jkl}) \, \mathbf{B}^{e(jkl)} \, \mathrm{det}\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right] \Bigg|_{\mathbf{y}} \tag{4.207}$$

with the consistent tangent operator (material matrix) at Gauss point $\boldsymbol{\xi}_{jkl}$ and state $\mathbf{y}$ as derived in Eq. (4.206)

$$\tilde{\mathbf{C}}_{\mathrm{L}}^e(\boldsymbol{\xi}_{jkl}) \Big|_{\mathbf{y}} = \left[\frac{\partial \tilde{\boldsymbol{\Phi}}}{\partial \mathbf{E}^{e(jkl)}} + \frac{\partial \tilde{\boldsymbol{\Phi}}}{\partial \mathbf{Q}^{e(jkl)}} \frac{\mathrm{d}\mathbf{Q}^{e(jkl)}}{\mathrm{d}\mathbf{E}^{e(jkl)}} \right] \Bigg|_{\boldsymbol{y}} . \tag{4.208}$$

The term $\dfrac{\mathrm{d}\mathbf{Q}^{e(jkl)}}{\mathrm{d}\mathbf{E}^{e(jkl)}}$ is obtained from the solution of the equation system (4.195) where it is the result of the discretization of $\dfrac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}}$ at Gauss point level.

Remark 12 *For elastic materials the second part of the matrix (4.208) vanishes and the consistent tangent operator is given by*

$$\tilde{\mathbf{C}}_{L}^e(\boldsymbol{\xi}_{jkl}) \Big|_{\mathbf{y}} = \left[\frac{\partial \tilde{\boldsymbol{\Phi}}}{\partial \mathbf{E}^{e(jkl)}} \right] \Bigg|_{\boldsymbol{y}} . \tag{4.209}$$

Compare for example (Bonet and Wood, 1997, p. 119).

4.8 Elimination of interior degrees of freedom

In this section, we exploit the distinction of hierarchical high order shape functions into interface modes $\mathbf{u}^{\mathrm{ei}}$ and bubble modes $\mathbf{u}^{\mathrm{eb}}$ on element level. As the bubble modes are purely local, i.e. independent from element to element, they can be condensed on local level and only the reduced element matrices (and corresponding right-hand side) need to be assembled to the global equation system. After its solution, the remaining bubble modes can be evaluated locally. This Schur complement approach is attractive for parallelization and is also known to improve the conditioning of the global equation system, (Ainsworth, 1996; Mandel, 1990a,b).

First, we distinguish in the global vector of unknowns $\boldsymbol{u}_a$ three components: the degrees of freedom that are prescribed $\overline{\boldsymbol{u}}$, and among those remaining the interface $\boldsymbol{u}_i$ and bubble $\boldsymbol{u}_b$ degrees of freedom. Correspondingly, the incidence matrices $\boldsymbol{Z}_a^e \in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{dof}}}$, $\overline{\boldsymbol{Z}}^e \in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{p}}}$, $\boldsymbol{Z}_i^e \in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{u}}}$, $\boldsymbol{Z}_b^e \in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{b}}}$ are introduced,

$$\mathbf{u}^e = \boldsymbol{Z}_a^e \boldsymbol{u}_a = [\boldsymbol{Z}_i^e \, \boldsymbol{Z}_b^e \, \overline{\boldsymbol{Z}}^e] \left\{ \begin{array}{c} \boldsymbol{u}_i \\ \boldsymbol{u}_b \\ \overline{\boldsymbol{u}} \end{array} \right\} = \boldsymbol{Z}_i^e \boldsymbol{u}_i + \boldsymbol{Z}_b^e \boldsymbol{u}_b + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}} \tag{4.210}$$

$$\delta\mathbf{u}^e = \boldsymbol{Z}_a^e \delta\boldsymbol{u}_a = [\boldsymbol{Z}_i^e \, \boldsymbol{Z}_b^e] \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\} = \boldsymbol{Z}_i^e \delta\boldsymbol{u}_i + \boldsymbol{Z}_b^e \delta\boldsymbol{u}_b \,. \tag{4.211}$$

Furthermore, on element level the interface degrees of freedom $\mathbf{u}^{\mathrm{ei}}$ and the bubble degrees of freedom $\mathbf{u}^{\mathrm{eb}}$ are distinguished,

$$\mathbf{u}^e = \left\{ \begin{array}{c} \mathbf{u}^{\mathrm{ei}} \\ \mathbf{u}^{\mathrm{eb}} \end{array} \right\}, \quad \delta\mathbf{u}^e = \left\{ \begin{array}{c} \delta\mathbf{u}^{\mathrm{ei}} \\ \delta\mathbf{u}^{\mathrm{eb}} \end{array} \right\} \tag{4.212}$$

and incidence matrices for the selection of the interface d.o.f., $\mathbf{Z}^{\mathrm{ei}} \in \mathbb{R}^{n_{\mathrm{ei}} \times n_{\mathrm{em}}}$, and bubble d.o.f., $\mathbf{Z}^{\mathrm{eb}} \in \mathbb{R}^{n_{\mathrm{eb}} \times n_{\mathrm{em}}}$ from the element mode vector $\mathbf{u}^e$ are defined,

$$\mathbf{u}^{\mathrm{ei}} = \mathbf{Z}^{\mathrm{ei}} \mathbf{u}^e = \mathbf{Z}^{\mathrm{ei}} (\boldsymbol{Z}_i^e \boldsymbol{u}_i + \boldsymbol{Z}_b^e \boldsymbol{u}_b + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}}) = \mathbf{Z}^{\mathrm{ei}} (\boldsymbol{Z}_i^e \boldsymbol{u}_i + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}}) \tag{4.213}$$

$$\mathbf{u}^{\mathrm{eb}} = \mathbf{Z}^{\mathrm{eb}} \mathbf{u}^e = \mathbf{Z}^{\mathrm{eb}} (\boldsymbol{Z}_i^e \boldsymbol{u}_i + \boldsymbol{Z}_b^e \boldsymbol{u}_b + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}}) = \mathbf{Z}^{\mathrm{eb}} (\boldsymbol{Z}_b^e \boldsymbol{u}_i + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}}) \tag{4.214}$$

$$\delta\mathbf{u}^{\mathrm{ei}} = \mathbf{Z}^{\mathrm{ei}} \delta\mathbf{u}^e = \mathbf{Z}^{\mathrm{ei}} (\boldsymbol{Z}_i^e \delta\boldsymbol{u}_i + \boldsymbol{Z}_b^e \delta\boldsymbol{u}_b) = \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e \delta\boldsymbol{u}_i \tag{4.215}$$

$$\delta\mathbf{u}^{\mathrm{eb}} = \mathbf{Z}^{\mathrm{eb}} \delta\mathbf{u}^e = \mathbf{Z}^{\mathrm{eb}} (\boldsymbol{Z}_i^e \delta\boldsymbol{u}_i + \boldsymbol{Z}_b^e \delta\boldsymbol{u}_b) = \mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_b^e \delta\boldsymbol{u}_b \tag{4.216}$$

where we used the fact that interface and bubble modes are independent by construction, i.e.

$$\mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_b^e = \boldsymbol{0}\,, \tag{4.217}$$

$$\mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_i^e = \boldsymbol{0}\,. \tag{4.218}$$

For the explication of the principle of virtual work these distinctions allow the partition of the shape function and strain displacement matrices, both on global as well as on local level:

$$\boldsymbol{u}^h = \boldsymbol{N}_a \boldsymbol{u}_a = \left[\boldsymbol{N}_i \, \boldsymbol{N}_b \, \overline{\boldsymbol{N}}\right] \left\{ \begin{array}{c} \boldsymbol{u}_i \\ \boldsymbol{u}_b \\ \overline{\boldsymbol{u}} \end{array} \right\} = \mathbf{N}^e \mathbf{u}^e = \left[\mathbf{N}^{\mathrm{ei}} \, \mathbf{N}^{\mathrm{eb}}\right] \left\{ \begin{array}{c} \mathbf{u}^{\mathrm{ei}} \\ \mathbf{u}^{\mathrm{eb}} \end{array} \right\} = \mathbf{u}^{e\,h} \tag{4.219}$$

$$\delta\boldsymbol{u}^h = \boldsymbol{N}_a \delta\boldsymbol{u}_a = \left[\boldsymbol{N}_i \, \boldsymbol{N}_b\right] \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\} = \mathbf{N}^e \delta\mathbf{u}^e = \left[\mathbf{N}^{\mathrm{ei}} \, \mathbf{N}^{\mathrm{eb}}\right] \left\{ \begin{array}{c} \delta\mathbf{u}^{\mathrm{ei}} \\ \delta\mathbf{u}^{\mathrm{eb}} \end{array} \right\} = \delta\mathbf{u}^{e\,h}\,. \tag{4.220}$$

For example, the transpose of the virtual displacement vector needed later evaluates to

$$\delta\mathbf{u}^{e\,h\,T} = \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\}^T \left[\begin{array}{c} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T}\mathbf{N}^{\mathrm{ei}\,T} \\ \boldsymbol{Z}_b^{e\,T}\mathbf{Z}^{\mathrm{eb}\,T}\mathbf{N}^{\mathrm{eb}\,T} \end{array} \right] . \tag{4.221}$$

For the strains we obtain a similar transition from global to local quantities

$$\begin{aligned} \boldsymbol{E}^h = \boldsymbol{B}_a^* \boldsymbol{u}_a = \left[\boldsymbol{B}_i^*\, \boldsymbol{B}_b^*\right] \left\{ \begin{array}{c} \boldsymbol{u}_i \\ \boldsymbol{u}_b \end{array} \right\} &= \mathbf{B}^{e*}\mathbf{u}^e = \left[\mathbf{B}^{\mathrm{ei}*}\, \mathbf{B}^{\mathrm{eb}*}\right] \left\{ \begin{array}{c} \mathbf{Z}^{\mathrm{ei}}(\boldsymbol{Z}_i^e \boldsymbol{u}_i + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}}) \\ \mathbf{Z}^{\mathrm{eb}}(\boldsymbol{Z}_b^e \boldsymbol{u}_i + \overline{\boldsymbol{Z}}^e \overline{\boldsymbol{u}}) \end{array} \right\} = \mathbf{E}^h \\ \delta\boldsymbol{E}^h = \boldsymbol{B}_a \boldsymbol{u}_a = \left[\boldsymbol{B}_i\, \boldsymbol{B}_b\right] \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\} &= \mathbf{B}^e \delta\mathbf{u}^e = \left[\mathbf{B}^{\mathrm{ei}}\, \mathbf{B}^{\mathrm{eb}}\right] \left\{ \begin{array}{c} \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e \delta\boldsymbol{u}_i \\ \mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_b^e \delta\boldsymbol{u}_b \end{array} \right\} = \delta\mathbf{E}^h \end{aligned} \tag{4.222}$$

and for later use the transpose of the virtual strains is stated

$$\delta\mathbf{E}^{\,h\,T} = \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\}^T \left[\begin{array}{c} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T}\mathbf{B}^{\mathrm{ei}\,T} \\ \boldsymbol{Z}_b^{e\,T}\mathbf{Z}^{\mathrm{eb}\,T}\mathbf{B}^{\mathrm{eb}\,T} \end{array} \right] . \tag{4.223}$$

Using these relations in the principle of virtual work, (4.102),(4.104) we arrive at

$$\begin{aligned} \pi(t, \boldsymbol{u}_i, \boldsymbol{u}_b, \delta\boldsymbol{u}_i, \delta\boldsymbol{u}_b, \boldsymbol{q}) &= \left\{ \begin{array}{c} \pi_i(t, \boldsymbol{u}_i, \delta\boldsymbol{u}_i, \boldsymbol{q}) \\ \pi_b(t, \boldsymbol{u}_b, \delta\boldsymbol{u}_b, \boldsymbol{q}) \end{array} \right\} = \\ &= \sum_{e=1}^{n_\mathrm{e}} \int_{\Omega^\mathrm{e}} \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\}^T \left[\begin{array}{c} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T}\mathbf{B}^{\mathrm{ei}\,T} \\ \boldsymbol{Z}_b^{e\,T}\mathbf{Z}^{\mathrm{eb}\,T}\mathbf{B}^{\mathrm{eb}\,T} \end{array} \right] \tilde{\boldsymbol{\Phi}}\Big(\mathbf{E}^e, \mathbf{q}^e\Big)\, \mathrm{d}\Omega^\mathrm{e} \\ &- \sum_{e=1}^{n_\mathrm{e}} \int_{\Omega^\mathrm{e}} \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\}^T \left[\begin{array}{c} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T}\mathbf{N}^{\mathrm{ei}\,T} \\ \boldsymbol{Z}_b^{e\,T}\mathbf{Z}^{\mathrm{eb}\,T}\mathbf{N}^{\mathrm{eb}\,T} \end{array} \right] \rho_\mathrm{R}\mathbf{k}\, \mathrm{d}\Omega^\mathrm{e} \\ &- \sum_{e=1}^{n_\mathrm{e}} \int_{\Gamma_{\mathrm{tR}}} \left\{ \begin{array}{c} \delta\boldsymbol{u}_i \\ \delta\boldsymbol{u}_b \end{array} \right\}^T \left[\begin{array}{c} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T}\mathbf{N}^{\mathrm{ei}\,T} \\ \boldsymbol{Z}_b^{e\,T}\mathbf{Z}^{\mathrm{eb}\,T}\mathbf{N}^{\mathrm{eb}\,T} \end{array} \right] \mathbf{t}_\mathrm{R}\, \mathrm{d}\Gamma_{\mathrm{tR}} = 0\,. \end{aligned} \tag{4.224}$$

Exploiting the arbitrariness of the virtual displacements and performing numerical integration by Gauss quadrature we obtain

$$\begin{aligned} \tilde{\boldsymbol{g}}_i(t, \boldsymbol{u}_i, \boldsymbol{q}) &= \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T} \sum_{jkl} w^{jkl} \mathbf{B}^{ei(jkl)\,T} \tilde{\boldsymbol{\Phi}}\Big(\mathbf{E}^e(\boldsymbol{u}_a), \mathbf{q}^e\big(\mathbf{E}^e(\boldsymbol{u}_a)\big)\Big) \det\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \\ &- \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T} \sum_{jkl} w^{jkl} \mathbf{N}^{\mathrm{ei}\,T} \rho_\mathrm{R}\mathbf{k} \det\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \\ &- \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T} \sum_{jkl} w^{jkl} \mathbf{N}^{\mathrm{ei}\,T} \mathbf{t}_\mathrm{R} \det\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \\ &= \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{e\,T}\mathbf{Z}^{\mathrm{ei}\,T} \mathbf{g}_i^e\,, \end{aligned} \tag{4.225, 4.226}$$

$$\begin{aligned}
\tilde{\boldsymbol{g}}_b(t,\boldsymbol{u}_b,\boldsymbol{q}) &= \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT}\mathbf{Z}^{\mathrm{eb}T}\sum_{jkl} w^{jkl}\mathbf{B}^{eb(jkl)T}\tilde{\Phi}\Big(\mathbf{E}^e(\boldsymbol{u}_a),\mathbf{q}^e\big(\mathbf{E}^e(\boldsymbol{u}_a)\big)\Big)\det\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \quad (4.227)\\
&- \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT}\mathbf{Z}^{\mathrm{eb}T}\sum_{jkl} w^{jkl}\mathbf{N}^{\mathrm{eb}T}\rho_\mathrm{R}\mathbf{k}\det\mathbf{J}^e(\boldsymbol{\xi}_{jkl})\\
&- \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT}\mathbf{Z}^{\mathrm{eb}T}\sum_{jkl} w^{jkl}\mathbf{N}^{\mathrm{eb}T}\mathbf{t}_\mathrm{R}\det\mathbf{J}^e(\boldsymbol{\xi}_{jkl})\\
&= \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT}\mathbf{Z}^{\mathrm{eb}T}\mathbf{g}_b^e\,, \quad (4.228)
\end{aligned}$$

where the local residual vector is denoted by $\mathbf{g}_i^e$ and $\mathbf{g}_b^e$ respectively.

In the context of the Newton-type methods derivatives are necessary (in this section we omit the change of notation from $\boldsymbol{u}$ to $\boldsymbol{U}$ that is introduced for the time integration of the evolution equations). The equation system we encounter is on global level

$$\begin{bmatrix} \frac{\partial\tilde{\boldsymbol{g}}_i}{\partial\boldsymbol{u}_i} & \frac{\partial\tilde{\boldsymbol{g}}_i}{\partial\boldsymbol{u}_b} \\ \frac{\partial\tilde{\boldsymbol{g}}_b}{\partial\boldsymbol{u}_i} & \frac{\partial\tilde{\boldsymbol{g}}_b}{\partial\boldsymbol{u}_b} \end{bmatrix} \begin{Bmatrix} \Delta\boldsymbol{u}_i \\ \Delta\boldsymbol{u}_b \end{Bmatrix} = -\begin{Bmatrix} \tilde{\boldsymbol{g}}_i \\ \tilde{\boldsymbol{g}}_b \end{Bmatrix} \rightarrow \begin{bmatrix} \boldsymbol{K}_{ii} & \boldsymbol{K}_{ib} \\ \boldsymbol{K}_{bi} & \boldsymbol{K}_{bb} \end{bmatrix} \begin{Bmatrix} \Delta\boldsymbol{u}_i \\ \Delta\boldsymbol{u}_b \end{Bmatrix} = -\begin{Bmatrix} \tilde{\boldsymbol{g}}_i \\ \tilde{\boldsymbol{g}}_b \end{Bmatrix}. \quad (4.229)$$

For explicit calculation of the terms we pre-evaluate

$$\frac{\partial\tilde{\Phi}\Big(\mathbf{E}^e(\boldsymbol{u}_a),\mathbf{q}^e\big(\mathbf{E}^e(\boldsymbol{u}_a)\big)\Big)}{\partial\boldsymbol{u}_i} = \tilde{\mathbf{C}}_\mathrm{L}^e\frac{\partial\mathbf{E}^e(\boldsymbol{u}_a)}{\partial\boldsymbol{u}_i}\,, \quad (4.230)$$

$$\frac{\partial\tilde{\Phi}\Big(\mathbf{E}^e(\boldsymbol{u}_a),\mathbf{q}^e\big(\mathbf{E}^e(\boldsymbol{u}_a)\big)\Big)}{\partial\boldsymbol{u}_a} = \tilde{\mathbf{C}}_\mathrm{L}^e\frac{\partial\mathbf{E}^e(\boldsymbol{u}_a)}{\partial\boldsymbol{u}_b}\,, \quad (4.231)$$

where use is made of the abbreviation for the consistent tangent operator

$$\tilde{\mathbf{C}}_\mathrm{L}^e = \left[\frac{\partial\tilde{\Phi}}{\partial\mathbf{E}^e} + \frac{\partial\tilde{\Phi}}{\partial\mathbf{q}^e}\frac{\mathrm{d}\mathbf{q}^e}{\mathrm{d}\mathbf{E}^e}\right]. \quad (4.232)$$

The derivative of the strains with respect to the displacements evaluate to

$$\frac{\partial\mathbf{E}^e(\boldsymbol{u}_a)}{\partial\boldsymbol{u}_i} = \mathbf{B}^{e\mathrm{i}*}\mathbf{Z}^{e\mathrm{i}}\boldsymbol{Z}_i^e \quad (4.233)$$

$$\frac{\partial\mathbf{E}^e(\boldsymbol{u}_a)}{\partial\boldsymbol{u}_b} = \mathbf{B}^{e\mathrm{b}*}\mathbf{Z}^{e\mathrm{b}}\boldsymbol{Z}_b^e\,. \quad (4.234)$$

Now, the terms of the stiffness matrix are given as assembly of the element stiffness matrices (assuming the the external loadings are independent of the deformation). For example, $\boldsymbol{K}_{ii}$ is

the assembly of all element stiffness matrices $\mathbf{k}_{ii}^e$,

$$\boldsymbol{K}_{ii} = \frac{\partial \tilde{\boldsymbol{g}}_i}{\partial \boldsymbol{u}_i} = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}^T} \underbrace{\left\{ \sum_{jkl} w^{jkl} \mathbf{B}^{ei(jkl)T} \tilde{\mathbf{C}}_\mathrm{L}^e(\boldsymbol{\xi}_{jkl}) \mathbf{B}^{ei*(jkl)} \det \mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right\}}_{\mathbf{k}_{ii}^e} \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e \quad (4.235)$$

$$\boldsymbol{K}_{ib} = \frac{\partial \tilde{\boldsymbol{g}}_i}{\partial \boldsymbol{u}_b} = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}^T} \underbrace{\left\{ \sum_{jkl} w^{jkl} \mathbf{B}^{ei(jkl)T} \tilde{\mathbf{C}}_\mathrm{L}^e(\boldsymbol{\xi}_{jkl}) \mathbf{B}^{eb*(jkl)} \det \mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right\}}_{\mathbf{k}_{ib}^e} \mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_b^e \quad (4.236)$$

$$\boldsymbol{K}_{bi} = \frac{\partial \tilde{\boldsymbol{g}}_b}{\partial \boldsymbol{u}_i} = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT} \mathbf{Z}^{\mathrm{eb}^T} \underbrace{\left\{ \sum_{jkl} w^{jkl} \mathbf{B}^{eb(jkl)T} \tilde{\mathbf{C}}_\mathrm{L}^e(\boldsymbol{\xi}_{jkl}) \mathbf{B}^{ei*(jkl)} \det \mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right\}}_{\mathbf{k}_{bi}^e} \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e \quad (4.237)$$

$$\boldsymbol{K}_{bb} = \frac{\partial \tilde{\boldsymbol{g}}_b}{\partial \boldsymbol{u}_b} = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT} \mathbf{Z}^{\mathrm{eb}^T} \underbrace{\left\{ \sum_{jkl} w^{jkl} \mathbf{B}^{eb(jkl)T} \tilde{\mathbf{C}}_\mathrm{L}^e(\boldsymbol{\xi}_{jkl}) \mathbf{B}^{eb*(jkl)} \det \mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right\}}_{\mathbf{k}_{bb}^e} \mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_b^e \,. \quad (4.238)$$

For the static condensation of the global equation system (4.229) the second equation is solved for $\Delta \boldsymbol{u}_b$ and substituted in the first equation to obtain an equation only in the unknown interface degrees of freedom $\Delta \boldsymbol{u}_i$

$$\underbrace{(\boldsymbol{K}_{ii} - \boldsymbol{K}_{ib} \boldsymbol{K}_{bb}^{-1} \boldsymbol{K}_{bi})}_{\boldsymbol{K}^*} \Delta \boldsymbol{u}_i = \underbrace{-\tilde{\boldsymbol{g}} + \boldsymbol{K}_{ib} \boldsymbol{K}_{bb}^{-1} \tilde{\boldsymbol{g}}_b}_{\boldsymbol{g}^*} \,. \quad (4.239)$$

The condensed matrix $\boldsymbol{K}^*$ and the condensed residual vector $\boldsymbol{g}^*$ can be traced to local level and it can be shown that the static condensation can be carried out in the element,

$$\begin{aligned} \boldsymbol{K}^* &= \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}^T} \mathbf{k}_{ii}^e \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e - \quad (4.240) \\ &- \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}^T} \mathbf{k}_{ib}^e \mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_b^e \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{e-1} \mathbf{Z}^{\mathrm{eb}^{-1}} \mathbf{k}_{bb}^{e\ -1} \mathbf{Z}^{\mathrm{eb}^{-T}} \boldsymbol{Z}_b^{e-T} \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT} \mathbf{Z}^{\mathrm{eb}^T} \mathbf{k}_{bi}^e \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e \\ &= \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}^T} \left[\mathbf{k}_{ii}^e - \mathbf{k}_{ib}^e \mathbf{k}_{bb}^{e\ -1} \mathbf{k}_{bi}^e \right] \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}^T} \left[\mathbf{k}^{e*} \right] \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e \end{aligned}$$

where we used the fact that bubble modes of different elements e have no interaction, i.e.

$$(\mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_b^e) (\boldsymbol{Z}_b^{\hat{e}-1} \mathbf{Z}^{\hat{e}\mathrm{b}-1}) = \begin{cases} \boldsymbol{I} & \text{for} \quad e = \hat{e} \\ \boldsymbol{0} & \text{for} \quad e \neq \hat{e} \end{cases} \,. \quad (4.241)$$

For the right-hand side $\boldsymbol{g}^*$ we can also show that the static condensation can be carried out

on element level,

$$\begin{aligned} \boldsymbol{g}^* &= -\sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}T} \mathbf{g}_i^e + \\ &+ \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}T} \mathbf{k}_{ib}^e \mathbf{Z}^{\mathrm{eb}} \boldsymbol{Z}_b^e \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{e-1} \mathbf{Z}^{e\mathrm{b}-1} \mathbf{k}_{bb}^{e\ -1} \mathbf{Z}^{e\mathrm{b}-T} \boldsymbol{Z}_b^{e-T} \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_b^{eT} \mathbf{Z}^{\mathrm{eb}T} \mathbf{g}_b^e \\ &= \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}T} \Big[-\mathbf{g}_i^e + \mathbf{k}_{ib}^e \mathbf{k}_{bb}^{e\ -1} \mathbf{g}_b^e \Big] = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}_i^{eT} \mathbf{Z}^{\mathrm{ei}T} \Big[\mathbf{g}^* \Big] . \end{aligned} \tag{4.242}$$

In the actual implementation the inverse matrix $\mathbf{k}_{bb}^{e\ -1}$ on element level does not have to be calculated explicitly. The Schur complement $\mathbf{k}^{e*}|\mathbf{g}^*$ can be obtained as result of a incomplete Gauss elimination on element level that is stopped after all bubble degrees of freedom are eliminated,

$$\begin{bmatrix} \mathbf{k}^{e*} & \mathbf{0} \\ \mathbf{k}_{bi}^{e\ *} & \mathbf{L} \end{bmatrix} \begin{Bmatrix} \Delta \mathbf{u}^{\mathrm{ei}} \\ \Delta \mathbf{u}^{\mathrm{eb}} \end{Bmatrix} = - \begin{Bmatrix} \mathbf{g}^* \\ \mathbf{g}_b^* \end{Bmatrix} . \tag{4.243}$$

A detailed discussion with efficient use of block elimination (BLAS and LAPACK routines) is given in (Niggl, 2007, Sec. 3.3.1). After the global displacement increment of the interface modes $\Delta \boldsymbol{u}_i$ has been calculated, the local interface modes are obtained via $\Delta \mathbf{u}^{\mathrm{ei}} = \mathbf{Z}^{\mathrm{ei}} \boldsymbol{Z}_i^e \Delta \boldsymbol{u}_i$. The last step is to evaluate the bubble degrees of freedom,

$$\Delta \mathbf{u}^{\mathrm{eb}} = -\mathbf{k}_{bb}^{e\ -1} (\mathbf{g}_b^e + \mathbf{k}_{ii}^e \Delta \mathbf{u}^{\mathrm{ei}}) . \tag{4.244}$$

In an actual implementation this can effectively be done by backsubstitution in $(4.243)_2$, where $\mathbf{L}$ is a lower left triangular matrix.

4.9 Reaction Forces

In Sec. 4.6.2 the principle of virtual displacement is used to derive the weak form Eq. (3.9) in the Galerkin formulation. After discretization the unknown displacements can be computed. However, by construction the principle of virtual displacements can not give information about reaction forces because the virtual displacement must vanish where a displacement is prescribed, $\delta \overline{\boldsymbol{u}} = \boldsymbol{0}$ on Γ_{ud}, and consequently no work is performed by the reaction forces. To compute reaction forces different variational principles must be applied. Here we investigate the method of Lagrangian multipliers and the closely related penalty function method.

4.9.1 Computing reaction forces

Outline of the argumentation. The Lagrange multiplier method and the penalty method for the imposition of constraints can be motivated in a very simple context, cf. (Hughes, 2000, pp. 194), (Bathe, 1996, pp. 143). Here this approach is applied to the DAE-system (4.174) and the Langrange multiplier is identified with the reaction forces pertaining to those displacement degrees of freedom where a constraint is enforced. This argumentation is based on (Hartmann, 2003a, Sec. 5.1.2) and (Hartmann et al., 2008). Next, a procedure for the calculation of the

Lagrange multiplier, viz the reaction force, from the internal force vector is discussed. The entries in question can be also obtained by the penalty method in a postprocessing step, i.e. no Lagrange multipliers are needed for the calculation, only for the interpretation. Finally, we argue that in the context of hierarchic Ansatz functions, see Sec. 4.2.2, the reaction force is the assembly only of those entries of the internal force vector corresponding to the linear, nodal shape functions. The method to calculate the reaction forces from the internal force vector is verified in Sec. 4.9.1.5 by comparing the result with the integration of the stresses along the boundaries where displacements are prescribed.

4.9.1.1 Motivation

The method of Lagrange multipliers and the penalty function for treating constraints are closely related and are important concepts for optimization theory and mathematical programming, compare for example (Luenberger, 2003). There, the task is to

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{subject to} && \mathbf{x} \in S, \text{ where } S = \{\mathbf{x} : c_i(\mathbf{x}) = 0, \quad i = 1, 2, ..., p\} . \end{aligned} \tag{4.245}$$

For further treatment the constrained problem (4.245) is transferred to an unconstrained formulation where the constraint is incorporated. In the case of the penalty function method we use

$$\text{minimize } f(\mathbf{x}) + \kappa P(\mathbf{x}) \tag{4.246}$$

where κ is a positive constant, the penalty parameter, and P is a continuous function that only vanishes if the constraint is satisfied, if not $P(\mathbf{x}) > 0$. Among the many possible functions, a very useful penalty function is

$$P(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^{p} c_i(\mathbf{x})^2 . \tag{4.247}$$

In continuum mechanics Lagrange multipliers and penalty function methods are adapted to impose constraints in the finite element method, compare (Hughes, 2000, pp. 194), (Bathe, 1996, pp. 143), (Belytschko et al., 2000, pp. 325), be it prescribed displacements, the treatment of contact problems (Wriggers, 2006) and more generally other constraints like incompressibility in mixed methods.[23]

For the *motivation* of those methods the existence of a potential is assumed, however, also for nonconservative systems weak forms using Lagrange multipliers and the penalty function method can be constructed with the 'guidance' from the conservative case, see (Belytschko et al., 2000, Eq. (6.3.42) and Eq. (6.3.50)).

4.9.1.2 Lagrange multiplier method for the DAE-system

This argument is based on (Hartmann, 2003a, Sec. 5.1.2) and (Hartmann et al., 2008). As introduced in Eq. (4.70) we distinguish in the vector of all modal degrees of freedom $\boldsymbol{u}_a(t) \in \mathbb{R}^{n_{\text{dof}}}$

[23] Besides to the penalty and Lagrange multiplier method linear displacement constraints can of course be enforced by direct elimination in the linear equation system. This is described in (Dhondt, 2004, Sec. 3.6).

the modal displacements $\boldsymbol{u} \in \mathbb{R}^{n_\mathrm{u}}$ without constraints and the modal displacements $\hat{\boldsymbol{u}} \in \mathbb{R}^{n_\mathrm{p}}$ where the displacement constraint $\overline{\boldsymbol{u}}(t)$ is prescribed,

$$\boldsymbol{u}_a = \left\{ \begin{array}{c} \boldsymbol{u} \\ \hat{\boldsymbol{u}} \end{array} \right\} . \tag{4.248}$$

Here also $\hat{\boldsymbol{u}}$ is unknown and will be obtained by the solution that satisfies the constraint equation $\boldsymbol{C}_c \in \mathbb{R}^{n_\mathrm{p}}$ stating the Dirichlet boundary conditions

$$\boldsymbol{C}_c(t, \boldsymbol{u}_a(t)) = \hat{\boldsymbol{u}} - \overline{\boldsymbol{u}}(t) = \boldsymbol{M}^T \boldsymbol{u}_a(t) - \overline{\boldsymbol{u}}(t) = \boldsymbol{0}, \qquad \boldsymbol{M} = \left[\begin{array}{c} \boldsymbol{0}_{n_\mathrm{u} \times n_\mathrm{p}} \\ \boldsymbol{I}_{n_\mathrm{p}} \end{array} \right] \tag{4.249}$$

with the filter matrix $\boldsymbol{M} \in \mathbb{R}^{n_\mathrm{dof} \times n_\mathrm{p}}$ and the prescribed modal displacements $\overline{\boldsymbol{u}}(t) \in \mathbb{R}^{n_\mathrm{p}}$.

Remark 13 *The Dirichlet b.c. have to be prescribed in the space of the hierarchic shape functions, see Sec. 4.2, as $\hat{\boldsymbol{u}}$ holds the coefficients of the hierarchic shape functions pertaining to constrained modes, cf. (Demkowicz, 2006, p. 177). For example, along an edge in a two-dimensional analysis homogenous ($\overline{\boldsymbol{u}} = \boldsymbol{0}$), constant and linear prescribed displacements can be readily enforced by setting the coefficients of the quadratic and higher modes (see Eq. (4.22)$_i, i > 2$) along this edge to zero and determining the value of the prescribed displacement at the end points of the edge in global x_1, x_2 direction corresponding to the linear modes Eq. (4.22)$_1$ and (4.22)$_2$. These values are the prescribed coefficients of the linear shape functions N_1, N_2.*

If more complicated displacements need to be prescribed they have to be projected in the space of the Ansatz functions to determine the coefficients of the shape functions giving the best approximation of $\overline{\boldsymbol{u}}(t)$. This operation is important for hp-adaptivity and is denoted as projection based interpolation. It is explained in (Demkowicz, 2006, Eq. (5.1), p. 74).

The variational equation π (3.9) is augmented by Lagrange multipliers to account for the Dirichlet b.c.

$$\begin{aligned} \pi_\mathrm{La}(t, \boldsymbol{u}_a, \delta\boldsymbol{u}_a, \boldsymbol{\lambda}, \delta\boldsymbol{\lambda}, \boldsymbol{q}) &= \delta\boldsymbol{u}_a^T \boldsymbol{g}_a(t, \boldsymbol{u}_a(t), \boldsymbol{q}(t)) + \delta\Big(\boldsymbol{\lambda}^T \boldsymbol{C}_c(t, \boldsymbol{u}_a)\Big) && (4.250) \\ &= \delta\boldsymbol{u}_a^T \boldsymbol{g}_a(t, \boldsymbol{u}_a(t), \boldsymbol{q}(t)) + \delta\boldsymbol{\lambda}^T(t) \boldsymbol{C}_c + \boldsymbol{\lambda}^T(t) \boldsymbol{M}^T \delta\boldsymbol{u}_a && (4.251) \\ &= \delta\boldsymbol{u}_a^T \Big(\boldsymbol{g}_a(t, \boldsymbol{u}_a(t), \boldsymbol{q}(t)) + \boldsymbol{M}\boldsymbol{\lambda}(t)\Big) + \delta\boldsymbol{\lambda}^T(t) \boldsymbol{C}_c = 0 && (4.252) \end{aligned}$$

with the Lagrange multipliers $\boldsymbol{\lambda}(t) \in \mathbb{R}^{n_\mathrm{p}}$. To obtain (4.251) the first variation $\delta(\boldsymbol{\lambda}^T \boldsymbol{C}_c)$ in (4.250) is carried out according to the definition, Eq. (3.26),

$$\begin{aligned} \delta\Big(\boldsymbol{\lambda}^T \boldsymbol{C}_c(t, \boldsymbol{u}_a)\Big) &= \delta\boldsymbol{\lambda}^T \boldsymbol{C}_c(t, \boldsymbol{u}_a) + \boldsymbol{\lambda}^T \delta\boldsymbol{C}_c(t, \boldsymbol{u}_a) && (4.253) \\ &= \frac{\mathrm{d}}{\mathrm{d}\epsilon}(\boldsymbol{\lambda} + \epsilon\delta\boldsymbol{\lambda})^T \Big|_{\epsilon=0} \boldsymbol{C}_c(t, \boldsymbol{u}_a) + \boldsymbol{\lambda}^T \frac{\mathrm{d}}{\mathrm{d}\epsilon}\Big(\boldsymbol{M}^T(\boldsymbol{u}_a + \epsilon\delta\boldsymbol{u}_a) - \overline{\boldsymbol{u}}(t)\Big)\Big|_{\epsilon=0} \\ &= \delta\boldsymbol{\lambda}^T \boldsymbol{C}_c(t, \boldsymbol{u}_a) + \boldsymbol{\lambda}^T \boldsymbol{M}^T \delta\boldsymbol{u}_a \,, && (4.254) \end{aligned}$$

where $\boldsymbol{C}_c$ is expanded as in Eq. (4.249) .

Exploiting the arbitrariness of $\delta\boldsymbol{u}_a \in \mathbb{R}^{n_{\rm dof}}$ and $\delta\boldsymbol{\lambda} \in \mathbb{R}^{n_{\rm p}}$ in Eq. (4.252) we arrive at, compare (Belytschko et al., 2000, Eq. (6.3.45), p. 327),

$$\begin{aligned} \boldsymbol{g}_a(t,\boldsymbol{u}_a(t),\boldsymbol{q}(t)) + \boldsymbol{M}\boldsymbol{\lambda}(t) &= \boldsymbol{0} \\ \boldsymbol{C}_c(t,\boldsymbol{u}_a(t)) &= \boldsymbol{0}\,. \end{aligned} \tag{4.255}$$

The modified DAE-system is obtained by similar arguments that lead to Eq. (4.175)

$$\boldsymbol{F}(t,\boldsymbol{y}(t),\dot{\boldsymbol{y}}(t)) \equiv \left\{ \begin{array}{c} \boldsymbol{g}_a(t,\boldsymbol{u}_a(t),\boldsymbol{q}(t)) + \boldsymbol{M}\boldsymbol{\lambda}(t) \\ \boldsymbol{C}_c(t,\boldsymbol{u}_a(t)) \\ \boldsymbol{A}\dot{\boldsymbol{q}}(t) - \boldsymbol{r}\big(t,\boldsymbol{u}_a(t),\boldsymbol{q}(t)\big) \end{array} \right\} = \boldsymbol{0}, \quad \boldsymbol{F} \in \mathbb{R}^{n_{\rm dof}+n_{\rm p}+n_{\rm Q}} \tag{4.256}$$

where

$$\boldsymbol{y}(t) \equiv \left\{ \begin{array}{c} \boldsymbol{u}_a(t) \\ \boldsymbol{\lambda}(t) \\ \boldsymbol{q}(t) \end{array} \right\} \quad \text{and the initial conditions } \boldsymbol{y}(t_0) \equiv \left\{ \begin{array}{c} \boldsymbol{u}_a(t_0) \\ \boldsymbol{\lambda}(t_0) \\ \boldsymbol{q}(t_0) \end{array} \right\} = \left\{ \begin{array}{c} \boldsymbol{u}_{a0} \\ \boldsymbol{\lambda}_0 \\ \boldsymbol{q}_0 \end{array} \right\} \equiv \boldsymbol{y}_0\,. \tag{4.257}$$

The constraints $\boldsymbol{C}_c$ are introduced in (4.249), the evolution equations in (4.174). The discretized variational principle (4.170) is now denoted as $\boldsymbol{g}_a$ and modified to account for all degrees of freedom explaining the $\boldsymbol{0}_{n_{\rm p}}$ in the vector of external loads corresponding to the constrained degrees of freedom $\hat{\boldsymbol{u}}$

$$\begin{aligned} \boldsymbol{g}_a(t,\boldsymbol{u}_a(t),\boldsymbol{q}(t)) \;\equiv\; & \sum_{e=1}^{n_{\rm e}} \boldsymbol{Z}_a^{eT} \left\{ \sum_{j=1}^{n_\xi}\sum_{k=1}^{n_\eta}\sum_{l=1}^{n_\zeta} w_j w_k w_l \mathbf{B}^{eT}\big(\mathbf{u}^e(t),\boldsymbol{\xi}_{jkl}\big) \right. \\ & \left. \tilde{\Phi}\Big(\mathbf{C}^e\big(\mathbf{E}^e(\boldsymbol{\xi}_{jkl},t)\big),\mathbf{q}^e\big(\mathbf{C}^e(\mathbf{E}^e(\boldsymbol{\xi}_{jkl},t))\big)\Big)\det\mathbf{J}^e(\boldsymbol{\xi}_{jkl}) \right\} \\ & - \left\{ \begin{array}{c} \boldsymbol{f}_{\rm ext}(t,\boldsymbol{u}_a(t)) \\ \boldsymbol{0}_{n_{\rm p}} \end{array} \right\} = \boldsymbol{0}, \end{aligned} \tag{4.258}$$

$\boldsymbol{g}_a \in \mathbb{R}^{n_{\rm dof}}$, with the incidence matrix $\boldsymbol{Z}_a^e$ according to (4.148) and the external force vector $\boldsymbol{f}_{\rm ext}$ given in Eq. (4.105). The local $\mathbf{B}^e$ matrix at Gauss point $\boldsymbol{\xi}_{jkl}$ depends on the local displacement vector $\mathbf{u}^e$ that may contain contributions to both global vectors $\boldsymbol{u}$ and $\hat{\boldsymbol{u}}$.

For time discretization the simplest DIRK method to solve the DAE system (4.256) is the backward Euler method that was already applied to the analogous Eq. (4.177). As for Eq. (4.186) we drop the subscript $n+1$ and introduce the abbreviations

$$\begin{aligned} T \;&\equiv\; t_{n+1}, \boldsymbol{U}_a \equiv \boldsymbol{u}_{a\,n+1}, \boldsymbol{C}_c \equiv \boldsymbol{C}_{cn+1}, \boldsymbol{\Lambda} \equiv \boldsymbol{\lambda}_{n+1}, \boldsymbol{Q} \equiv \boldsymbol{q}_{n+1}, \\ \hat{\boldsymbol{G}} \;&\equiv\; \hat{\boldsymbol{G}}_{n+1}, \boldsymbol{G} \equiv \boldsymbol{G}_{n+1}, \boldsymbol{L} \equiv \boldsymbol{L}_{n+1}, \boldsymbol{R} \equiv \boldsymbol{R}_{n+1}\,. \end{aligned} \tag{4.259}$$

Then, Eq. (4.256) is rendered as

$$\boldsymbol{R}\,(\boldsymbol{U}_a,\boldsymbol{\Lambda},\boldsymbol{Q}) = \boldsymbol{R}\,(\boldsymbol{y}) = \left\{ \begin{array}{c} \hat{\boldsymbol{G}}(\boldsymbol{U}_a,\boldsymbol{\Lambda},\boldsymbol{Q}) \\ \boldsymbol{L}(\boldsymbol{U}_a,\boldsymbol{Q}) \end{array} \right\} = \boldsymbol{0}, \quad \boldsymbol{R} \in \mathbb{R}^{n_{\rm dof}+n_{\rm p}+n_{\rm Q}}, \tag{4.260}$$

with

$$\hat{\boldsymbol{G}}(\boldsymbol{U}_a, \boldsymbol{\Lambda}, \boldsymbol{Q}) \equiv \left\{ \begin{array}{c} \boldsymbol{G}(\boldsymbol{U}_a, \boldsymbol{\Lambda}, \boldsymbol{Q}) \\ \boldsymbol{C}_c(\boldsymbol{U}_a) \end{array} \right\} = \left\{ \begin{array}{c} \boldsymbol{g}_a(T, \boldsymbol{U}_a, \boldsymbol{Q}) + \boldsymbol{M}\boldsymbol{\Lambda} \\ \boldsymbol{M}^T \boldsymbol{U}_a - \overline{\boldsymbol{u}}(T) \end{array} \right\} \tag{4.261}$$

$$\boldsymbol{L}(\boldsymbol{U}_a, \boldsymbol{Q}) \equiv \boldsymbol{A} \left\{ \frac{\boldsymbol{Q} - \boldsymbol{q}_n}{\Delta t_n} \right\} - \boldsymbol{r}(\boldsymbol{U}_a, \boldsymbol{Q}) \,. \tag{4.262}$$

To apply the multi-level Newton algorithm, cf. Sec. 4.7.2, the internal variables $\boldsymbol{Q}$ are calculated on local level for a given modal displacement $\boldsymbol{U}_a$, compare (4.192). On the global level the directional derivative of $\hat{\boldsymbol{G}}$, Eq. (4.261), according to (4.189) (substitute $\boldsymbol{G}$ by $\hat{\boldsymbol{G}}$) yields the system of linear equations for iteration $m \rightsquigarrow m+1$

$$\left\{ \begin{array}{l} \mathsf{D}_{\boldsymbol{U}_a} \boldsymbol{G}(\boldsymbol{y})[\Delta \boldsymbol{U}_a] + \mathsf{D}_{\boldsymbol{\Lambda}} \boldsymbol{G}(\boldsymbol{y})[\Delta \boldsymbol{\Lambda}] \\ \mathsf{D}_{\boldsymbol{U}_a} \boldsymbol{C}_c(\boldsymbol{y})[\Delta \boldsymbol{U}_a] + \mathsf{D}_{\boldsymbol{\Lambda}} \boldsymbol{C}_c(\boldsymbol{y})[\Delta \boldsymbol{\Lambda}] \end{array} \right\} = - \left\{ \begin{array}{c} \boldsymbol{G}(\boldsymbol{y}) \\ \boldsymbol{C}_c(\boldsymbol{y}) \end{array} \right\} \tag{4.263}$$

with $\boldsymbol{y}^T = \{ {\boldsymbol{U}_a}^{(m)^T} \boldsymbol{\Lambda}^{(m)^T} \boldsymbol{Q}^{(m)^T} \}$ and the increments $\Delta \boldsymbol{U}_a = {\boldsymbol{U}_a}^{(m+1)} - {\boldsymbol{U}_a}^{(m)}$ as well as $\Delta \boldsymbol{\Lambda} = \boldsymbol{\Lambda}^{(m+1)} - \boldsymbol{\Lambda}^{(m)}$. Evaluating the individual terms in (4.263) we obtain

$$\mathsf{D}_{\boldsymbol{U}_a} \boldsymbol{G}(\boldsymbol{y}))[\Delta \boldsymbol{U}_a] = \boldsymbol{K}_a|_{\boldsymbol{y}} \, \Delta \boldsymbol{U}_a = \left[\frac{\partial \boldsymbol{G}}{\partial \boldsymbol{U}_a} + \frac{\partial \boldsymbol{G}}{\partial \boldsymbol{Q}} \frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}\boldsymbol{U}_a} \right] \Big|_{\boldsymbol{y}} \Delta \boldsymbol{U}_a, \quad \text{cf. Eq. (4.190)} \tag{4.264}$$

$$\mathsf{D}_{\boldsymbol{\Lambda}} \boldsymbol{G}[\Delta \boldsymbol{\Lambda}] = \boldsymbol{M} \, \Delta \boldsymbol{\Lambda} \tag{4.265}$$

$$\mathsf{D}_{\boldsymbol{U}_a} \boldsymbol{C}_c[\Delta \boldsymbol{U}_a] = \boldsymbol{M}^T \, \Delta \boldsymbol{U}_a \tag{4.266}$$

$$\mathsf{D}_{\boldsymbol{\Lambda}} \boldsymbol{C}_c[\Delta \boldsymbol{\Lambda}] = \boldsymbol{0}_{n_\mathrm{p} \times n_\mathrm{p}} \, \Delta \boldsymbol{\Lambda}. \tag{4.267}$$

The tangential stiffness matrix $\boldsymbol{K}_a|_{\boldsymbol{y}}$ is singular because no rigid body modes are suppressed, mathematically representing an under-determined system. Inserting (4.264) – (4.267) in the linearized equation system (4.263) we obtain for the global equation system

$$\left[\begin{array}{cc} \boldsymbol{K}_a|_{\boldsymbol{y}} & \boldsymbol{M} \\ \boldsymbol{M}^T & \boldsymbol{0}_{n_\mathrm{p} \times n_\mathrm{p}} \end{array} \right] \left\{ \begin{array}{c} \Delta \boldsymbol{U}_a \\ \Delta \boldsymbol{\Lambda} \end{array} \right\} = - \left\{ \begin{array}{c} \boldsymbol{G}(\boldsymbol{y}) \\ \boldsymbol{C}_c(\boldsymbol{y}) \end{array} \right\} . \tag{4.268}$$

Partitioning the system to distinguish the free and prescribed degrees of freedom as in the displacement vector (4.248) yields

$$\left[\begin{array}{ccc} \boldsymbol{K} & \boldsymbol{K}_{up} & \boldsymbol{0}_{n_\mathrm{u} \times n_\mathrm{p}} \\ \boldsymbol{K}_{pu} & \boldsymbol{K}_{pp} & \boldsymbol{I}_{n_\mathrm{p}} \\ \boldsymbol{0}_{n_\mathrm{p} \times n_\mathrm{u}} & \boldsymbol{I}_{n_\mathrm{p}} & \boldsymbol{0}_{n_\mathrm{p} \times n_\mathrm{p}} \end{array} \right] \Big|_{\boldsymbol{y}} \left\{ \begin{array}{c} \Delta \boldsymbol{U} \\ \Delta \hat{\boldsymbol{U}} \\ \Delta \boldsymbol{\Lambda} \end{array} \right\} = \left\{ \begin{array}{c} -\boldsymbol{f}(\boldsymbol{y}) \\ -\bar{\boldsymbol{f}}(\boldsymbol{y}) - \boldsymbol{\Lambda}^{(m)} \\ \overline{\boldsymbol{u}}(T) - \hat{\boldsymbol{U}}^{(m)} \end{array} \right\} = \left\{ \begin{array}{c} -\boldsymbol{f}_{\mathrm{int}}(\boldsymbol{y}) + \boldsymbol{f}_{\mathrm{ext}}(\boldsymbol{y}) \\ -\bar{\boldsymbol{f}}_{\mathrm{int}}(\boldsymbol{y}) - \boldsymbol{\Lambda}^{(m)} \\ \overline{\boldsymbol{u}}(T) - \hat{\boldsymbol{U}}^{(m)} \end{array} \right\} \tag{4.269}$$

where we introduced

$$\boldsymbol{U}_a = \left\{ \begin{array}{c} \boldsymbol{U} \\ \hat{\boldsymbol{U}} \end{array} \right\}, \quad \Delta \boldsymbol{U}_a = \left\{ \begin{array}{c} \Delta \boldsymbol{U} \\ \Delta \hat{\boldsymbol{U}} \end{array} \right\} , \tag{4.270}$$

as well as the sub-matrices that are assembled from the elemental stiffness matrices $\mathbf{k}^e$ according to

$$\boldsymbol{K}|_{\boldsymbol{y}} = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}^{eT} \mathbf{k}^e|_{\boldsymbol{y}} \boldsymbol{Z}^e, \quad \boldsymbol{K}_{up}|_{\boldsymbol{y}} = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}^{eT} \mathbf{k}^e|_{\boldsymbol{y}} \overline{\boldsymbol{Z}}^e, \tag{4.271}$$

$$\boldsymbol{K}_{pu}|_{\boldsymbol{y}} = \sum_{e=1}^{n_\mathrm{e}} \overline{\boldsymbol{Z}}^{eT} \mathbf{k}^e|_{\boldsymbol{y}} \boldsymbol{Z}^e, \quad \boldsymbol{K}_{pp}|_{\boldsymbol{y}} = \sum_{e=1}^{n_\mathrm{e}} \overline{\boldsymbol{Z}}^{eT} \mathbf{k}^e|_{\boldsymbol{y}} \overline{\boldsymbol{Z}}^e, \tag{4.272}$$

compare (4.157), and (4.148) – (4.149) for the definition of the incidence matrices $\boldsymbol{Z}^e, \overline{\boldsymbol{Z}}^e$. Furthermore, the residual vector $(4.261)_1$ belonging to the state $\boldsymbol{y}$, cf. Eq. (4.258), is partitioned as[24]

$$\begin{aligned} \boldsymbol{G}(\boldsymbol{y}) &= \boldsymbol{g}_a(t_{n+1}, \boldsymbol{U}_a^{(m)}, \Lambda^{(m)}) + \boldsymbol{M}\Lambda^{(m)} \qquad (4.273)\\ &= \left\{ \begin{array}{c} \boldsymbol{f}(\boldsymbol{y}) \\ \overline{\boldsymbol{f}}(\boldsymbol{y}) \end{array} \right\} + \left\{ \begin{array}{c} \boldsymbol{0}_{n_\mathrm{u}} \\ \Lambda^{(m)} \end{array} \right\} = \left\{ \begin{array}{cc} \boldsymbol{f}_{\mathrm{int}}(\boldsymbol{y}) & - \quad \boldsymbol{f}_{\mathrm{ext}}(\boldsymbol{y}) \\ \overline{\boldsymbol{f}}_{\mathrm{int}}(\boldsymbol{y}) & \end{array} \right\} + \left\{ \begin{array}{c} \boldsymbol{0}_{n_\mathrm{u}} \\ \Lambda^{(m)} \end{array} \right\} \end{aligned}$$

$$= \left\{ \begin{array}{cc} \displaystyle\sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}^{eT} \left\{ \sum_{(jkl)} w^{jkl} \mathbf{B}^{e(jkl)T}(\mathbf{U}^e) \tilde{\Phi}\Big(\mathbf{C}^{e(jkl)}, \mathbf{Q}^{e(jkl)}\big(\mathbf{C}^{e(jkl)}\big)\Big) \det \mathbf{J}^{e(jkl)} \right\} & - \quad \boldsymbol{f}_{\mathrm{ext}} \\ \displaystyle\sum_{e=1}^{n_\mathrm{e}} \overline{\boldsymbol{Z}}^{eT} \left\{ \sum_{(jkl)} w^{jkl} \mathbf{B}^{e(jkl)T}(\mathbf{U}^e) \tilde{\Phi}\Big(\mathbf{C}^{e(jkl)}, \mathbf{Q}^{e(jkl)}\big(\mathbf{C}^{e(jkl)}\big)\Big) \det \mathbf{J}^{e(jkl)} \right\} & \end{array} \right\}$$
$$+ \left\{ \begin{array}{c} \boldsymbol{0}_{n_\mathrm{u}} \\ \Lambda^{(m)} \end{array} \right\}$$

where $\overline{\boldsymbol{f}}_{\mathrm{int}}$ is the internal force vector containing the entries belonging to prescribed displacement degrees of freedom $\hat{\boldsymbol{u}}$ assembled by $\overline{\boldsymbol{Z}}^{eT}$, while $\boldsymbol{f}_{\mathrm{int}}$ corresponds to the 'free' degrees of freedom $\boldsymbol{u}$, cf. (4.154) – (4.155), assembled by $\boldsymbol{Z}^{eT}$ as

$$\boldsymbol{f}_{\mathrm{int}} = \sum_{e=1}^{n_\mathrm{e}} \boldsymbol{Z}^{eT} \mathbf{f}_{\mathrm{int}}^e \qquad (4.274)$$

$$\overline{\boldsymbol{f}}_{\mathrm{int}} = \sum_{e=1}^{n_\mathrm{e}} \overline{\boldsymbol{Z}}^{eT} \mathbf{f}_{\mathrm{int}}^e \qquad (4.275)$$

with $\mathbf{f}_{\mathrm{int}}^e$ given in (4.154).

After this technical work we proceed with the interpretation in the framework of the MLNA. It is natural to use the prescribed displacements as initial guess for $\hat{\boldsymbol{U}}$ in the first iteration, i.e. $\hat{\boldsymbol{U}}^{(0)} = \overline{\boldsymbol{u}}(T) = \overline{\boldsymbol{u}}(t_{n+1})$. Then evaluating $(4.269)_3$ gives that the increment $\Delta\hat{\boldsymbol{U}} = \boldsymbol{0}$ and therefore $\hat{\boldsymbol{U}}^{(0)} = \hat{\boldsymbol{U}}^{(m)} = \overline{\boldsymbol{u}}(T) = \overline{\boldsymbol{u}}(t_{n+1})$ for all iterations m. Consequently the remaining equations of (4.269) now state

$$\boldsymbol{K}|_{\boldsymbol{y}} \Delta\boldsymbol{U} = -\boldsymbol{f}(\boldsymbol{y}) \qquad (4.276)$$

$$\boldsymbol{K}_{pu}|_{\boldsymbol{y}} \Delta\boldsymbol{U} + \underbrace{\left\{ \Lambda^{(m+1)} - \Lambda^{(m)} \right\}}_{\Delta\Lambda} = -\Lambda^{(m)} - \overline{\boldsymbol{f}}_{\mathrm{int}}(\boldsymbol{y}) \qquad (4.277)$$

$$\Rightarrow \boldsymbol{K}_{pu}|_{\boldsymbol{y}} \Delta\boldsymbol{U} = -\Lambda^{(m+1)} - \overline{\boldsymbol{f}}_{\mathrm{int}}(\boldsymbol{y})\,. \qquad (4.278)$$

Eq. (4.276) can be solved independently, using the internal variables obtained by the local stress algorithm $(4.256)_3$. After the global iteration is converged the increment $\Delta\boldsymbol{U} \approx \boldsymbol{0}$ and Eq. (4.278) now reads

$$\boldsymbol{0} = -\Lambda^{(m+1)} - \overline{\boldsymbol{f}}_{\mathrm{int}}(\boldsymbol{y}) \quad \Rightarrow \qquad (4.279)$$

$$\Lambda^{(m+1)} = -\overline{\boldsymbol{f}}_{\mathrm{int}}(\boldsymbol{y}) = -\overline{\boldsymbol{f}}_{\mathrm{int}}(\boldsymbol{U}_a, \boldsymbol{Q}) \quad \text{for } \overline{\boldsymbol{f}}_{\mathrm{int}} \text{ see Eq. } (4.273)_2. \qquad (4.280)$$

[24] The summation over the Gauss points from Eq. (4.258) is abbreviated here to obtain a shorter notation.

This means, the Lagrange multiplier can be interpreted as the reaction force

$$\boldsymbol{f}_{\mathrm{RF}} = \boldsymbol{\Lambda} = -\bar{\boldsymbol{f}}_{\mathrm{int}} \tag{4.281}$$

equilibrating the internal forces $\bar{\boldsymbol{f}}_{\mathrm{int}}$ corresponding to the constrained displacement degrees of freedom, cf. Eq. (4.249).

Concerning the implementation we note that Eq. (4.276) is independent of the Lagrange multipliers and analogous to the equation system derived for the pure principle of virtual displacements, Eq. (4.159) in Sec. 4.6. In the case of a converged solution — the Dirichlet b.c. can also be imposed by the penalty function method, see Sec. 4.9.1.3 — the (negative, cf. Eq. (4.281)) reaction force vector of the complete structure is obtained by the assembly operation described in Eq. (4.275) of the internal force vector $\mathbf{f}^e_{\mathrm{int}}$, Eq. (4.154), on *element* level,

$$\begin{aligned} -\boldsymbol{f}_{\mathrm{RF}} &= \bar{\boldsymbol{f}}_{\mathrm{int}} = \sum_{e=1}^{n_{\mathrm{e}}} \overline{\boldsymbol{Z}}^{eT} \mathbf{f}^e_{\mathrm{int}} \qquad (4.282) \\ &= \sum_{e=1}^{n_{\mathrm{e}}} \overline{\boldsymbol{Z}}^{eT} \left\{ \sum_{(jkl)} w^{jkl} \mathbf{B}^{e(jkl)T}\big(\mathbf{u}^e(t)\big) \tilde{\Phi}\Big(\mathbf{C}^{e(jkl)}(t), \mathbf{q}^e\big(\mathbf{C}^{e(jkl)}(t)\big)\Big) \det \mathbf{J}^{e(jkl)} \right\} . \end{aligned}$$

4.9.1.3 Penalty function method

The penalty function method for the imposition of Dirichlet b.c. augments the variational equation π (3.9) with a term that penalizes the violation of the prescribed constraints. Since the advent of computerized structural analysis in the late '50s the penalty approach was used in a physical 'disguise' where the displacement constraint is motivated by a very stiff spring. In the '70s the penalty function method has been described and investigated in formal mathematical terms, compare (Felippa, 1977) and the references therein. A general class of penalty functions is given in (Luenberger, 2003, Eq. (18), p. 372). The most popular penalty function is

$$P(\boldsymbol{u}_a) = \frac{1}{2} \kappa \boldsymbol{C}_c^T \boldsymbol{C}_c \,, \tag{4.283}$$

with a positive constant κ and the constraint equation $\boldsymbol{C}_c$ as set forth in Eq. (4.249).

We now define a slightly different variational equation (compare (4.250) for the Lagrangian form π_{La})

$$\pi_{\mathrm{Pen}}(t, \boldsymbol{u}_a, \delta\boldsymbol{u}_a, \boldsymbol{q}) = \delta\boldsymbol{u}_a^T \boldsymbol{g}_a(t, \boldsymbol{u}_a(t), \boldsymbol{q}(t)) + \delta P(\boldsymbol{u}_a) \,. \tag{4.284}$$

The variational term δP is evaluated[25] invoking the definition Eq. (3.26), where we expand

[25] As δP is a scalar product of column matrices we can use the property $a^T\, b = b^T\, a$, for example to transform $(\boldsymbol{M}^T \delta\boldsymbol{u}_a(t))^T\, \boldsymbol{C}_c$ to $\boldsymbol{C}_c^T\, \boldsymbol{M}^T \delta\boldsymbol{u}_a(t)$ and further $(\boldsymbol{C}_c^T \boldsymbol{M}^T)\, \delta\boldsymbol{u}_a(t)$ to $\delta\boldsymbol{u}_a^T(t)\, (\boldsymbol{C}_c^T \boldsymbol{M}^T)^T = \delta\boldsymbol{u}_a^T(t)\, (\boldsymbol{M}\, \boldsymbol{C}_c)$. Voilá

$\boldsymbol{C}_c$ according to (4.249),

$$\begin{aligned}
\delta P(\boldsymbol{u}_a) &= \delta\Big(\frac{1}{2}\kappa \boldsymbol{C}_c^T(t,\boldsymbol{u}_a)\boldsymbol{C}_c(t,\boldsymbol{u}_a)\Big) \\
&= \frac{1}{2}\kappa\delta\boldsymbol{C}_c^T\,\boldsymbol{C}_c + \frac{1}{2}\kappa\boldsymbol{C}_c^T\,\delta\boldsymbol{C}_c \\
&= \frac{1}{2}\kappa\,\frac{\mathrm{d}}{\mathrm{d}\epsilon}\Big(\boldsymbol{M}^T\big(\boldsymbol{u}_a(t)+\epsilon\delta\boldsymbol{u}_a(t)\big)-\overline{\boldsymbol{u}}(t)\Big)^T\Big|_{\epsilon=0}\boldsymbol{C}_c \\
&+ \frac{1}{2}\kappa\boldsymbol{C}_c^T\,\frac{\mathrm{d}}{\mathrm{d}\epsilon}\Big(\boldsymbol{M}^T\big(\boldsymbol{u}_a(t)+\epsilon\delta\boldsymbol{u}_a(t)\big)-\overline{\boldsymbol{u}}(t)\Big)\Big|_{\epsilon=0} \\
&= \frac{1}{2}\kappa\Big(\boldsymbol{M}^T\delta\boldsymbol{u}_a(t)\Big)^T\boldsymbol{C}_c + \frac{1}{2}\kappa\boldsymbol{C}_c^T\boldsymbol{M}^T\delta\boldsymbol{u}_a(t) \\
&= \frac{1}{2}\kappa\boldsymbol{C}_c^T\boldsymbol{M}^T\delta\boldsymbol{u}_a(t) + \frac{1}{2}\kappa\boldsymbol{C}_c^T\boldsymbol{M}^T\delta\boldsymbol{u}_a(t) \\
&= \delta\boldsymbol{u}_a^T\,\kappa\boldsymbol{M}\boldsymbol{C}_c = \delta\boldsymbol{u}_a^T\,\kappa\boldsymbol{M}\Big\{\boldsymbol{M}^T\boldsymbol{u}_a(t)-\overline{\boldsymbol{u}}(t)\Big\}\,. \qquad (4.285)
\end{aligned}$$

Introducing this result in Eq. (4.284), the variational formulation for the penalty method is obtained as

$$\pi_{\mathrm{Pen}}(t,\boldsymbol{u}_a,\delta\boldsymbol{u}_a,\boldsymbol{q}) = \delta\boldsymbol{u}_a^T\Big\{\boldsymbol{g}_a(t,\boldsymbol{u}_a(t),\boldsymbol{q}(t))+\kappa\boldsymbol{M}\big\{\boldsymbol{M}^T\boldsymbol{u}_a(t)-\overline{\boldsymbol{u}}(t)\big\}\Big\}\,, \qquad (4.286)$$

where $\boldsymbol{g}_a$ is defined in Eq. (4.258).

Exploiting the arbitrariness of $\delta\boldsymbol{u}_a$ we obtain the space discretized DAE-system

$$\boldsymbol{F}(t,\boldsymbol{y}(t),\dot{\boldsymbol{y}}(t)) \equiv \left\{\begin{array}{c} \boldsymbol{g}_a(t,\boldsymbol{u}_a(t),\boldsymbol{q}(t))+\kappa\boldsymbol{M}\Big\{\boldsymbol{M}^T\boldsymbol{u}_a(t)-\overline{\boldsymbol{u}}(t)\Big\} \\ \boldsymbol{A}\dot{\boldsymbol{q}}(t)-\boldsymbol{r}\big(t,\boldsymbol{u}_a(t),\boldsymbol{q}(t)\big) \end{array}\right\} = \boldsymbol{0}, \quad \boldsymbol{F}\in\mathbb{R}^{n_{\mathrm{dof}}+n_{\mathrm{Q}}} \qquad (4.287)$$

where

$$\boldsymbol{y}(t) \equiv \left\{\begin{array}{c}\boldsymbol{u}_a(t)\\ \boldsymbol{q}(t)\end{array}\right\} \quad \text{with the initial conditions } \boldsymbol{y}(t_0) \equiv \left\{\begin{array}{c}\boldsymbol{u}_a(t_0)\\ \boldsymbol{q}(t_0)\end{array}\right\} = \left\{\begin{array}{c}\boldsymbol{u}_{a0}\\ \boldsymbol{q}_0\end{array}\right\} \equiv \boldsymbol{y}_0\,. \qquad (4.288)$$

For the time discretization of the DAE system (4.287) we again employ the Backward-Euler method to solve for time $T = t_{n+1}$ using the abbreviations introduced in (4.259),

$$\boldsymbol{R}\,(\boldsymbol{U}_a,\boldsymbol{Q}) = \boldsymbol{R}\,(\boldsymbol{y}) = \left\{\begin{array}{c}\boldsymbol{G}(\boldsymbol{U}_a,\boldsymbol{Q})\\ \boldsymbol{L}(\boldsymbol{U}_a,\boldsymbol{Q})\end{array}\right\} = \boldsymbol{0}, \quad \boldsymbol{R}\in\mathbb{R}^{n_{\mathrm{dof}}+n_{\mathrm{Q}}}, \qquad (4.289)$$

with

$$\boldsymbol{G}(\boldsymbol{U}_a,\boldsymbol{Q}) \equiv \boldsymbol{g}_a(T,\boldsymbol{U},\boldsymbol{Q})+\kappa\boldsymbol{M}\Big\{\boldsymbol{M}^T\boldsymbol{U}_a-\overline{\boldsymbol{u}}(T)\Big\} \qquad (4.290)$$

$$\boldsymbol{L}(\boldsymbol{U}_a,\boldsymbol{Q}) \equiv \boldsymbol{A}\left\{\frac{\boldsymbol{Q}-\boldsymbol{q}_n}{\Delta t_n}\right\} - \boldsymbol{r}(\boldsymbol{U}_a,\boldsymbol{Q})\,. \qquad (4.291)$$

To apply the Multilevel-Newton algorithm we linearize the differential equation (4.290) according to Eq. (4.189) at a given state $\boldsymbol{y} \equiv (\boldsymbol{U}_a^{(m)},\boldsymbol{Q}^{(m)})$,

$$\mathsf{D}_{\boldsymbol{U}_a}\boldsymbol{G}\big(\boldsymbol{U}_a,\boldsymbol{Q}(\boldsymbol{U}_a)\big)[\Delta\boldsymbol{U}_a] = -\boldsymbol{G}(\boldsymbol{y})\,, \qquad (4.292)$$

with the individual terms

$$\mathsf{D}_{\boldsymbol{U}_a}\boldsymbol{G}\big(\boldsymbol{U}_a,\boldsymbol{Q}(\boldsymbol{U}_a)\big)[\Delta\boldsymbol{U}_a] = \Big[\boldsymbol{K}_a|_{\boldsymbol{y}} + \kappa\boldsymbol{M}\boldsymbol{M}^T\Big]\,\Delta\boldsymbol{U}_a \tag{4.293}$$

$$\boldsymbol{G}(\boldsymbol{y}) = \boldsymbol{g}_a(T,\boldsymbol{U}_a^{(m)},\boldsymbol{Q}^{(m)}) + \kappa\boldsymbol{M}\Big\{\boldsymbol{M}^T\boldsymbol{U}_a^{(m)} - \overline{\boldsymbol{u}}(T)\Big\} \tag{4.294}$$

to arrive at the linear equation system

$$\Big[\boldsymbol{K}_a|_{\boldsymbol{y}} + \kappa\boldsymbol{M}\boldsymbol{M}^T\Big]\,\Delta\boldsymbol{U}_a = -\boldsymbol{g}_a(T,\boldsymbol{y}) - \kappa\boldsymbol{M}\big\{\boldsymbol{M}^T\boldsymbol{U}_a^{(m)} - \overline{\boldsymbol{u}}(T)\big\}, \tag{4.295}$$

where $\boldsymbol{K}_a|_{\boldsymbol{y}}$ is given in Eq. $(4.271)_1$ and $\boldsymbol{g}_a$ in Eq. (4.258).

Partitioning the system (4.295) according to the vector (4.270) we arrive at

$$\begin{aligned}\begin{bmatrix} \boldsymbol{K} & \boldsymbol{K}_{up} \\ \boldsymbol{K}_{pu} & \boldsymbol{K}_{pp}+\kappa\boldsymbol{I}\end{bmatrix}\Big|_{\boldsymbol{y}} \begin{Bmatrix} \Delta\boldsymbol{U} \\ \Delta\hat{\boldsymbol{U}}\end{Bmatrix} &= \begin{Bmatrix} -\boldsymbol{f}(\boldsymbol{y}) \\ -\bar{\boldsymbol{f}}(\boldsymbol{y}) - \kappa\big\{\hat{\boldsymbol{U}}^{(m)} - \overline{\boldsymbol{u}}(T)\big\}\end{Bmatrix} \\ &= \begin{Bmatrix} -\boldsymbol{f}_{\text{int}}(\boldsymbol{y}) + \boldsymbol{f}_{\text{ext}}(\boldsymbol{y}) \\ -\bar{\boldsymbol{f}}_{\text{int}}(\boldsymbol{y}) - \kappa\big\{\hat{\boldsymbol{U}}^{(m)} - \overline{\boldsymbol{u}}(T)\big\}\end{Bmatrix}\end{aligned} \tag{4.296}$$

using the matrices defined in Eq. (4.271) – (4.272) and the load vectors from Eq. (4.273).

Remark 14 *The positive penalty parameter κ has to be chosen large enough that the constraints $\boldsymbol{C}_c$ are observed satisfactorily. However, as the penalty parameter is added to those entries on the* diagonal *of the system stiffness matrix, where constraints are given, the condition number of the global equation system deteriorates with increasing value of the penalty parameter. This is investigated in (Luenberger, 2003, p. 373ff). When iterative solvers like the conjugate gradient method are used this results in additional iterations, when direct solvers are employed, the system can be still solved, however the accuracy may be affected, i.e. the solution is sensitive to small perturbations of the stiffness matrix entries or the right hand side. However, these shortcomings are only theoretically as they can be healed by the simple operation of diagonal preconditioning, a standard first step in solver implementations.*

In the p-FEM code AdhoC the value of the penalty parameter κ is typically chosen as 10^{20}. If dynamic problems using small time steps Δt with direct time integration methods (like the generalized-α method (Chung and Hulbert, 1993)) are calculated, the penalty parameter has to be even higher to satisfy the condition to be much larger than the diagonal entries $\boldsymbol{K}_{ii}$ as the inverse of Δt^2 enters the resulting stiffness matrix, compare (Heisserer, 2001, Eq. (3.171) together with Eq. (3.168)).

4.9.1.4 Reaction force in the context of hierarchic high-order finite elements

We have identified the negative vector $\bar{\boldsymbol{f}}_{\text{int}}$ (4.275) with the reaction force vector $\boldsymbol{f}_{\text{RF}}$, i.e. those forces necessary to maintain the constraint (4.249) and obtained the rule (4.282) to compute the reaction force vector corresponding to *the chosen Ansatz space*.

It is known that for hierarchic shape functions only the linear modes have direct physical meaning. Contrarily, if standard low order shape functions or high-order shape functions like Lagrange functions that constitute a partition of unity are employed, see Eq. (4.11) and Fig. 4.4(a) for the one-dimensional case, we can directly sum the entries of the vector $\bar{\boldsymbol{f}}_{\text{int}}$

belonging to the displacement degrees of freedom along a boundary of interest and obtain the reaction force in the global coordinate system.

If *hierarchic* high-order shape functions constructed from integrated Legendre polynomials as introduced in Eqs. (4.12) – (4.14) are applied, cf. Fig. 4.4(b), only the bilinear nodal modes constitute a partition of unity, see Eq. (4.24). Consequently only those entries from the vector $\overline{\boldsymbol{f}}_{\text{int}}$ must be taken into account that correspond to the nodal modes, Eq. (4.23), to compute a reaction force with physical meaning.

The importance of the linear modes can also be demonstrated by investigating the entries of the external load vector $\boldsymbol{f}_{\text{ext}}$ (4.107). The following analogy justifies the examination of the external force vector: like the external force vector $\boldsymbol{f}_{\text{ext}}$ is upon convergence in equilibrium with the part of the internal force vector $\boldsymbol{f}_{\text{int}}$ corresponding to the 'free' displacement degrees of freedom, in the same way the other part of the internal force vector $\overline{\boldsymbol{f}}_{\text{int}}$ corresponding to constrained d.o.f. is equilibrated by the reaction forces $\boldsymbol{f}_{\text{RF}}$, compare Eq. (4.273).

$$\boldsymbol{G}(\boldsymbol{y}) = \left\{ \begin{array}{c} \boldsymbol{f}_{\text{int}} \\ \overline{\boldsymbol{f}}_{\text{int}} \end{array} \right\} - \left\{ \begin{array}{c} \boldsymbol{f}_{\text{ext}} \\ -\boldsymbol{f}_{\text{RF}} \end{array} \right\} = \boldsymbol{0}\,. \tag{4.297}$$

Therefore, if the physically applied external load is recovered by adding the 'linear' components of $\boldsymbol{f}_{\text{ext}}$, more precise the components corresponding to the linear modes (4.23), that are in equilibrium with the linear entries of $\boldsymbol{f}_{\text{int}}$, we can argue that the linear components of $\overline{\boldsymbol{f}}_{\text{int}}$ give the negative *physical* reaction force.

Assume for simplicity that along a horizontal boundary (global $x-y$ and local $\xi-\eta$ directions may coincide, $\det\mathbf{J} = 1$) of the reference element a normal traction $t_y(\xi)$ and tangential traction $t_x(\xi) = 0$ is applied. Then, the load vector $\boldsymbol{f}_{\text{ext}}$ is according to Eq. (4.107)

$$\boldsymbol{f}_{\text{ext}\,t_{\text{R}}} = \int_{\Gamma_{\text{tR}}} \boldsymbol{N}^T \left\{ \begin{array}{c} t_x \\ t_y \end{array} \right\} \mathrm{d}\Gamma_{\text{tR}} = \int_{\xi=-1}^{\xi=1} \left[\begin{array}{cc} N_1(\xi) & . \\ . & N_1(\xi) \\ N_2(\xi) & . \\ . & N_2(\xi) \\ N_3(\xi) & . \\ . & N_3(\xi) \\ \vdots & \vdots \end{array} \right] \left\{ \begin{array}{c} 0 \\ t_y \end{array} \right\} \mathrm{d}\xi = \left\{ \begin{array}{c} 0 \\ \int_\xi N_1(\xi) t_y \,\mathrm{d}\xi \\ 0 \\ \int_\xi N_2(\xi) t_y \,\mathrm{d}\xi \\ 0 \\ \int_\xi N_3(\xi) t_y \,\mathrm{d}\xi \\ \vdots \end{array} \right\}, \tag{4.298}$$

where the integrals in the load vector were abbreviated by $\int_\xi \equiv \int_{\xi=-1}^{\xi=1}$. Next we compute the total applied load P_y and try to identify the external load in the load vector $\boldsymbol{f}_{\text{ext}\,t_{\text{R}}}$.

$$P_y = \int_{\xi=-1}^{\xi=1} t_y(\xi)\,\mathrm{d}\xi = \int_{\xi=-1}^{\xi=1} 1 \cdot t_y(\xi)\,\mathrm{d}\xi = \int_{\xi=-1}^{\xi=1} \Big[N_1(\xi) + N_2(\xi)\Big] t_y(\xi)\,\mathrm{d}\xi \tag{4.299}$$

$$= \int_{\xi=-1}^{\xi=1} N_1(\xi) t_y(\xi)\,\mathrm{d}\xi + \int_{\xi=-1}^{\xi=1} N_2(\xi) t_y(\xi)\,\mathrm{d}\xi\,, \tag{4.300}$$

where we used the property that the linear modes, i.e. the nodal shape functions constitute a partition of unity, $1 = N_1 + N_2$, cf. Eq. (4.19) and (4.24) for the 2D case. Identifying the terms

of (4.300) giving the external applied load in y-direction in the load vector (4.298), we see that we have to sum the second and the fourth component, i.e. those entries where the linear shape functions N_1 and N_2 are multiplied with t_y. By the same reasoning the total applied load in x direction would be obtained by summing the first and third component. The higher shape functions, $N_i, i \geq 3$ do not contribute to the partition of unity – their 'purpose' is to enrich the Ansatz space and they are responsible for the beneficial numerical properties of the p-version. By analogy we conclude that the physical total reaction force along a boundary with prescribed displacement is obtained by summing the linear components of the vector $\overline{\boldsymbol{f}}_{\text{int}}$ of all nodal degrees of freedom constituting this boundary. For a note concerning the implementation see Remark 15 on p. 106.

This procedure, termed 'method RF', is verified in Sec. 4.9.1.5, where the reaction force is alternatively computed by integration of stresses along the boundary.

4.9.1.5 Verification of reaction forces for an L-shaped axisymmetric domain

This example serves for the verification of the calculation of reaction forces. The reaction forces extracted from the entries of the internal force vector corresponding to the linear (nodal) shape functions of the elements along the boundary ('**method RF**') as described in Sec. 4.9.1.4 are compared to the force obtained by integration of the stresses along cutlines located on the boundaries ('**method** F_σ'). The geometric setting of the *axisymmetric* domain and the boundary conditions are shown in Fig. 4.17. Displacement controlled pressing at the top and bottom introduces an inhomogeneous deformation ($\overline{u}_{\text{bottom}} = 2.4 = -0.8\overline{u}_{\text{top}}$). The setting is

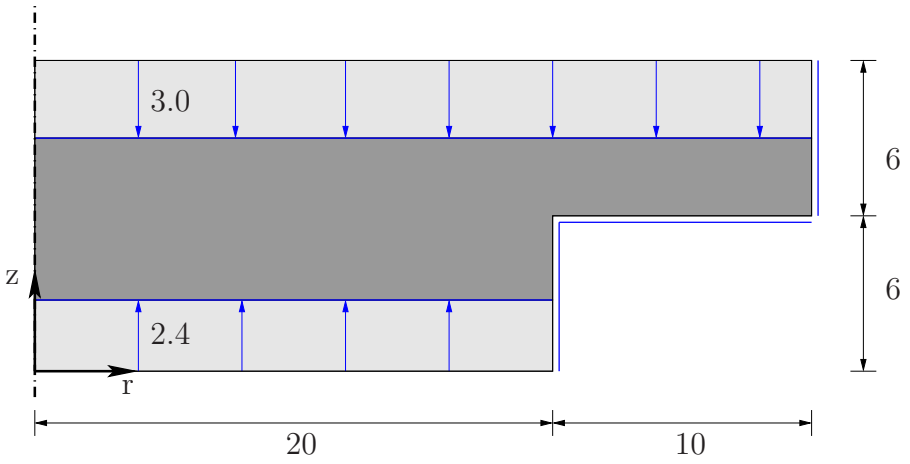
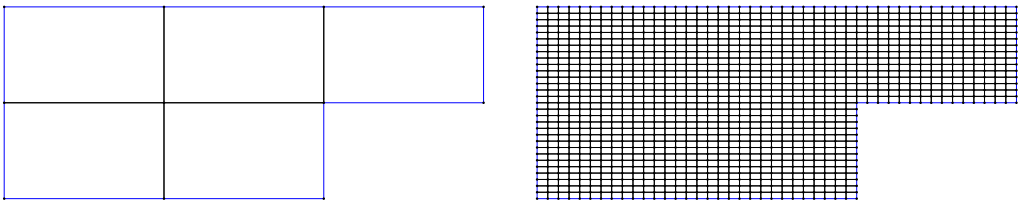


Figure 4.17: System and boundary conditions for the L-shape example with displacement control

the same as in Sec. 5.4.1.2, however here St.Venant material ($E = 1000, \nu = 0$) and linearized kinematics are chosen. These simplifications are justified as we are only interested in comparing the two approaches to obtain the reaction force.

The polynomial degree is uniformly increased from $p = 1$ to $p = 10$ and for each p the elements in the base mesh Fig. 4.18(a) are uniformly subdivided in $n \times n$ sub-elements, $n = 1, 3, 5, 7, 9, 11, 13, 15$ (see Fig. 4.18(b)).

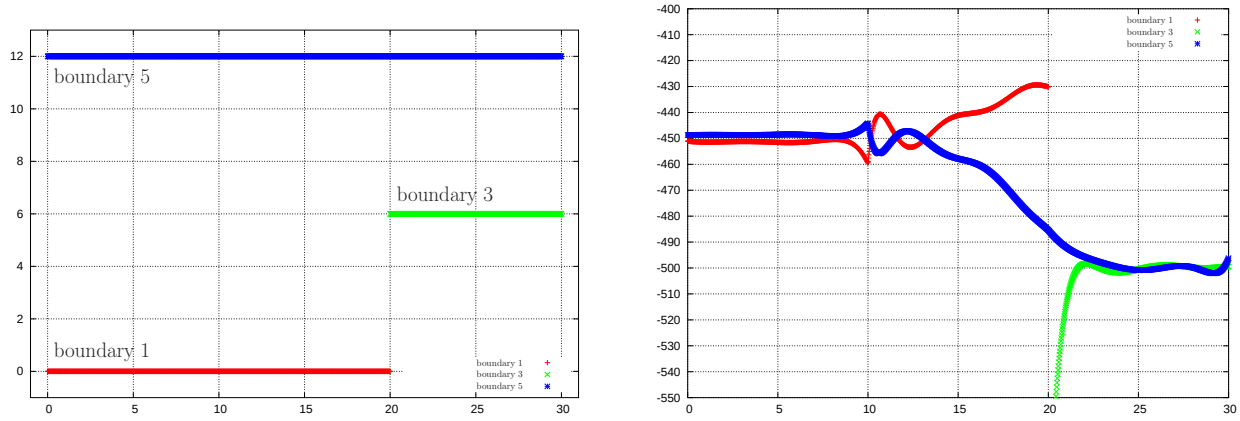
Method F_σ: To be able to calculate the reaction force F_σ from the integration of the stresses along the boundaries of the L-shaped cross section, the stress state is written out



(a) base h-mesh (b) h-refined mesh, e.g. 15×15

Figure 4.18: Base mesh, where each element is uniformly subdivided into $n \times n$, up to $n = 15$ elements

using cutlines with 1500 points along the radial extension of the domain. Fig. 4.19 shows the location of the postprocessing points where the stresses are evaluated and an exemplary distribution of σ_{zz} stresses along these boundaries.



(a) Location of evaluation points (b) σ_{zz} along these lines for $p = 10$, mesh 5x 1x1

Figure 4.19: Location of the boundaries where the stress is plotted

These σ_{zz} stresses are integrated along the horizontal boundaries using a simple trapezoidal rule taking the axisymmetric setting (integration over one radian) into account, compare Fig. 4.20 for the notation. We have a series of points r_i with corresponding values f_i along a cutline. The integral is composed of the sum of the intervals $I = \sum_i I_i$. Over one interval (r_i, r_{i+1}) the mean value is $\overline{f}_i = \frac{1}{2}(f_i + f_{i+1})$. The integral over one radian of one interval is approximated by

$$I_i = \int_{r_i}^{r_{i+1}} \overline{f}_i r \, \mathrm{d}r = \overline{f}_i \frac{1}{2} \Big[r^2 \Big]_{r_i}^{r_{i+1}} = \overline{f}_i \frac{1}{2} \Big(r_{i+1}^2 - r_i^2 \Big) \, . \tag{4.301}$$

Remark 15 *For the calculation of the reaction force with the method RF those entries of the* element *internal force vector* $\mathbf{f}_{int}^e$ *pertaining to linear, i.e. nodal shape functions must be added along the boundary in question, compare Sec. 4.9.1.4. The situation at the reentrant corner warrants a closer look for the computer implementation, compare Fig. 4.21. It is straightforward in the p-FE code AdhoC to obtain all edges that constitute a boundary with a*

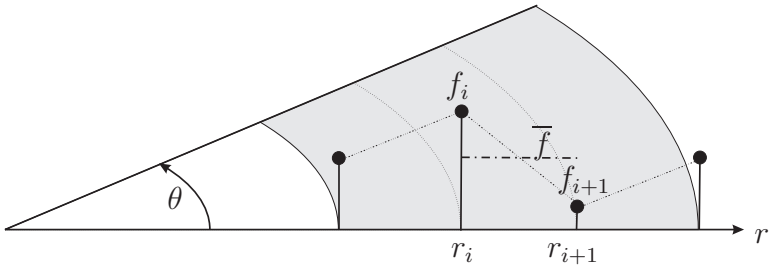


Figure 4.20: Notations for trapezoidal integration in a cylindrical coordinate system

certain boundary condition code (bcc). But if we would only take the internal force vector for the elements containing those edges into account we would solely consider the elements labeled 'A' in Fig. 4.21. The elements that only contribute a node to the boundary with the bcc in question would be omitted. However, as can be seen from the sketch, the elements 'B' and 'C' also contribute to the global nodal shape function associated with the global node N and therefore are found and evaluated in the implementation.

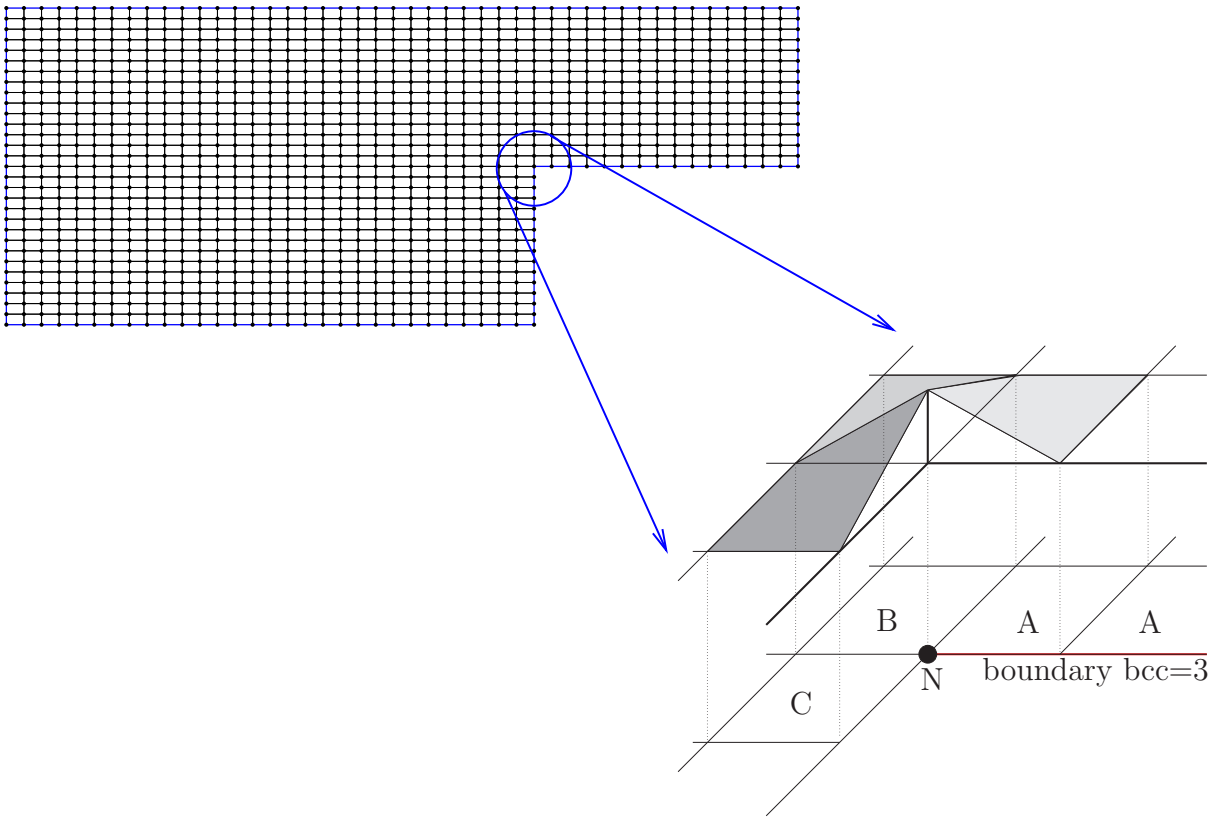


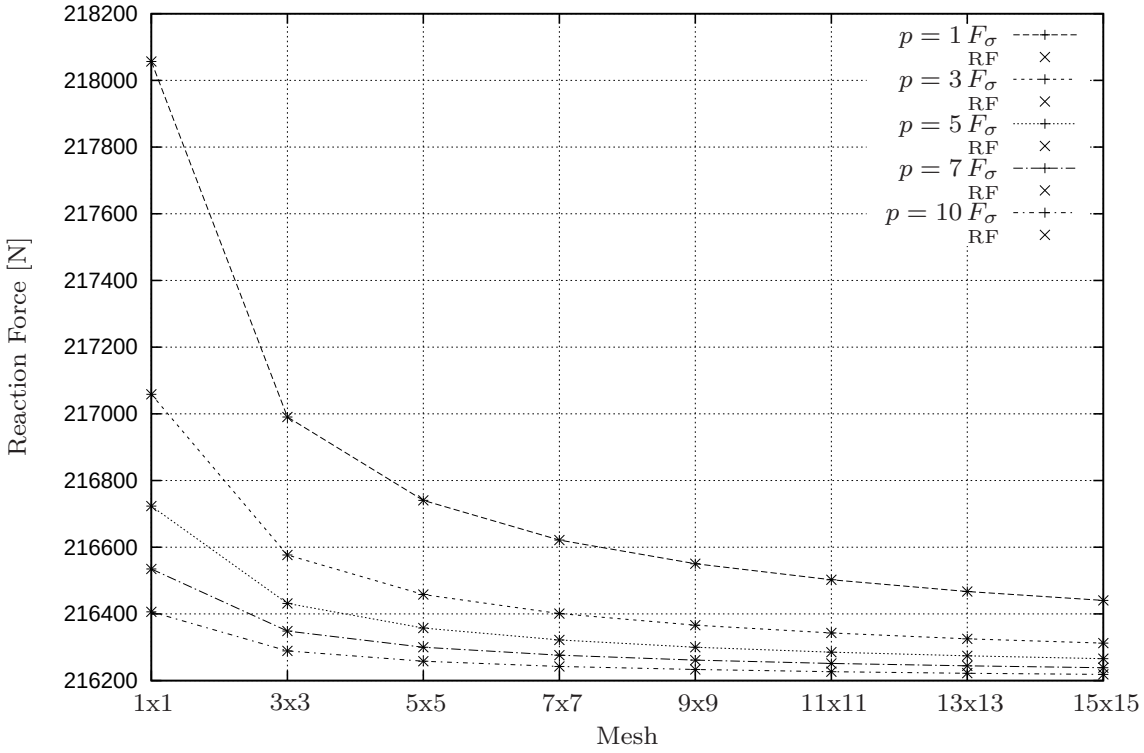
Figure 4.21: Calculation of the reaction force via the internal force vector: elements to be taken into account for boundary 3

The reaction force along the upper boundary (boundary 5, compare Fig. 4.19(a) for the numbers of the boundaries) obtained by the method RF and the method F_σ are compared in Fig. 4.22(a). It can be seen that both methods deliver very good matching results for

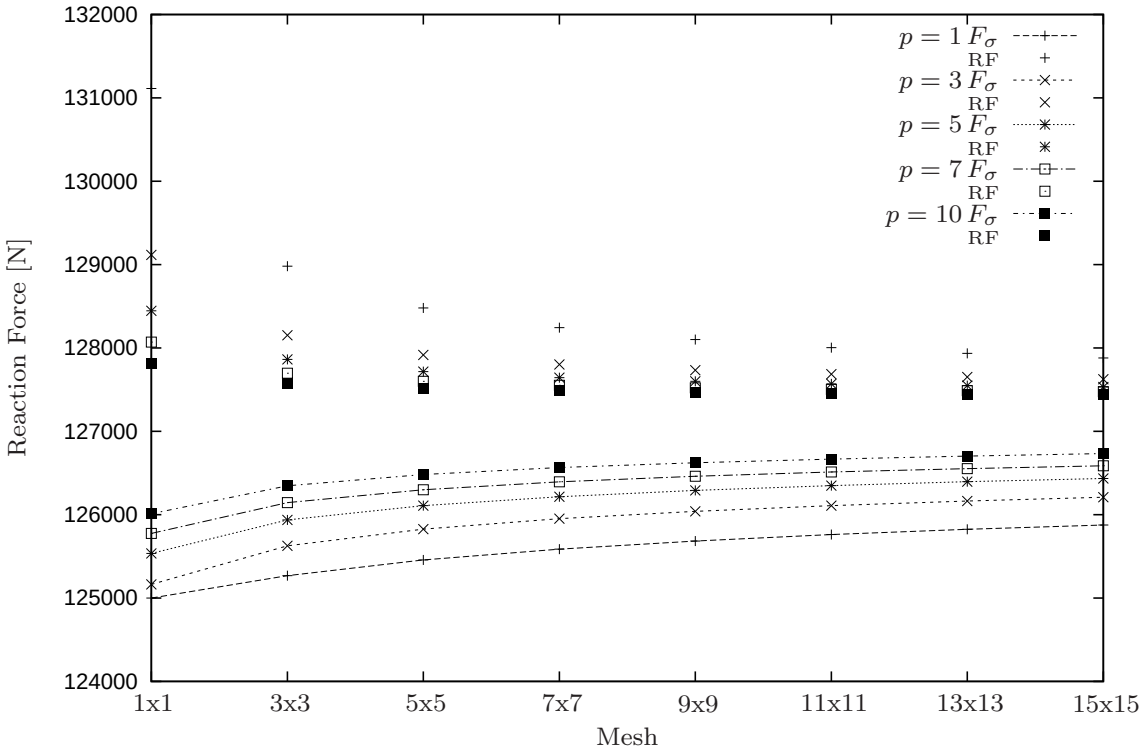
increasing polynomial order and mesh refinement. The same holds true for the reaction force along boundary 1 (bottom left, not plotted).

Along the boundary 3 (bottom right) both methods differ in about 700 N for $p = 10$ on the finest mesh. This relative difference of only about 0.5 % can be explained by the stress-singularity at the reentrant corner. It poses a challenge to any numerical quadrature and the trapezoidal rule is known to be the least exact of Newton-Cotes type integration schemes. The method RF exploits by construction integrals that are used to compute the solution and inherently computes the reaction force to the best accuracy for a present polynomial order and mesh. The method F_σ on the other hand relies on post-processed data and the accuracy can be increased by evaluating the stresses at more and more post-processing points, in this case the boundary 3 was resolved by 500 integration points. Of course a higher accurate integration scheme than the trapezoid rule could be used but the limitation that the result near a singularity depends on the number of integration points remains. When using material models with internal variables another advantage of the method RF becomes obvious. The internal force vector $\mathbf{f}^{\mathrm{e}}_{\mathrm{int}}$ (4.282) is evaluated over the element domain Ω^{e} and using numerical integration the stresses are only needed at the Gauss-Legendre points inside the element. If the stress along a cutline like in the method F_σ need to be evaluated each of the post processing points has to hold an array of internal variables and the internal variables have to be updated after each converged step. With the method RF these additional computations are not needed.

The residual force in axial direction, i.e. the sum of the axial reaction forces of boundary 1,3 and 5 should vanish and is therefore an indicator for the quality of the solution. For the two methods discussed the residual is shown in Fig. 4.23. The first plot shows the absolute value, i.e. the out-of balance force, the second plot the relative residuum, i.e. the residuum is divided by the reaction force along the top boundary (boundary 5). Both figures use a logarithmic scale for the axis of ordinates. The difference induced by the singularity discussed above leaves its traces here also. As the reaction forces along boundary 1 and 5 coincide for both methods the results along boundary 3 cause the discrepancy. It can be noted that the absolute value of the residuum for the method RF is about 10^{-10} for all computations and the relative value, i.e. the residuum over the reaction force of the top boundary is even smaller. As indicated the relative difference for the method F_σ also decreases to less than 1% (0.3% for $p = 10$ and the finest mesh).

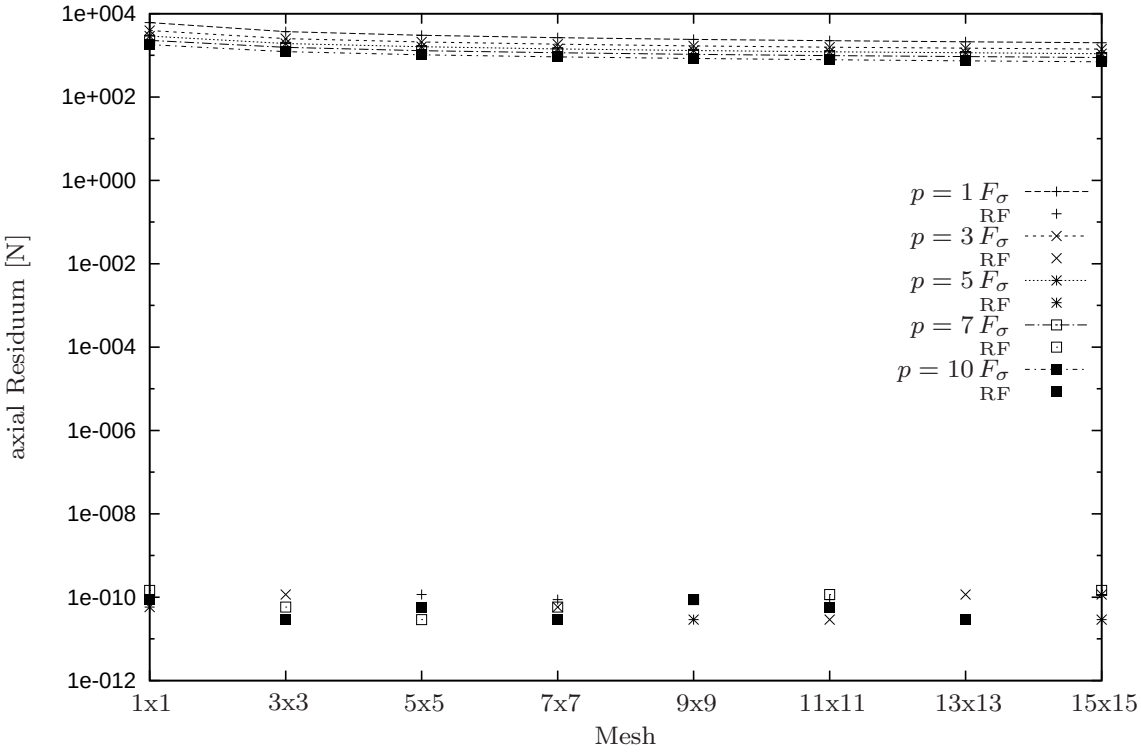


(a) Reaction force along boundary 5 (top)

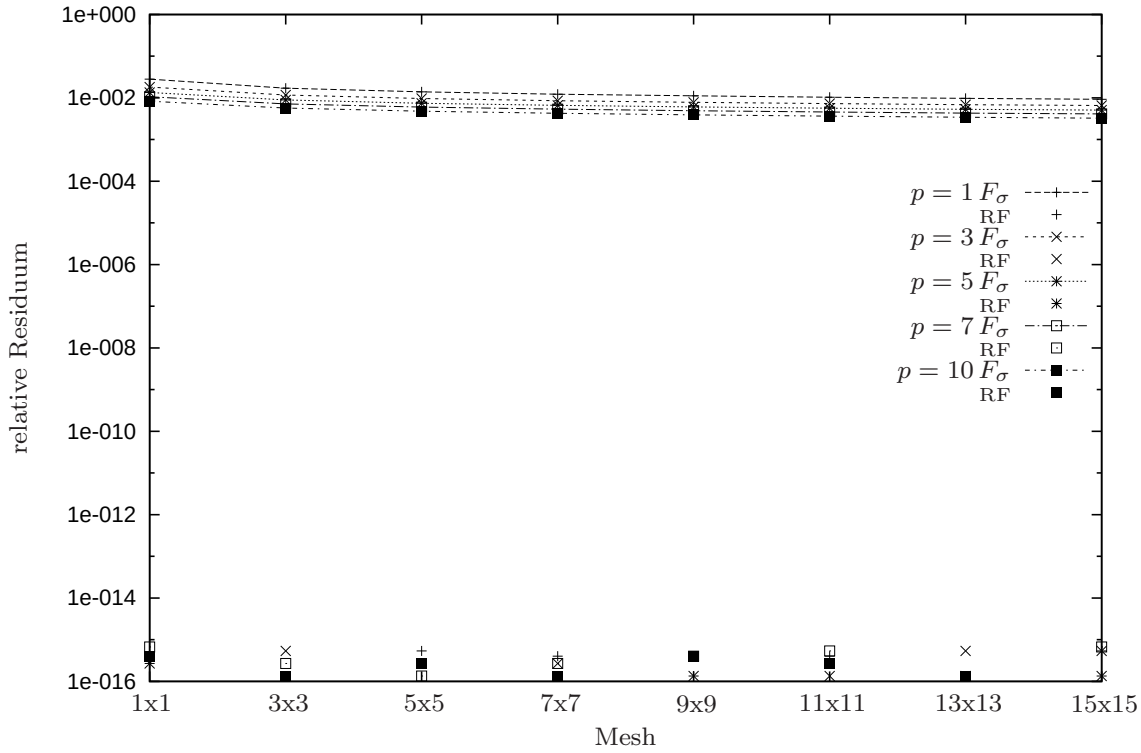


(b) Reaction force along boundary 3 (bottom right)

Figure 4.22: Comparison of the reaction force obtained by the two methods



(a) Absolute value of the axial residuum



(b) Relative axial residuum (normalized by reaction force of top boundary)

Figure 4.23: Residuum obtained by the two methods: absolute value and normalized

4.9.2 Application to the investigation of the St.Venant-Kirchhoff material

The simplest example of a hyperelastic material is the St.Venant-Kirchhoff model, see Remark 3 on p. 17 and (Bonet and Wood, 1997, p. 120), (Wriggers, 2001, pp. 45), (Bathe, 2002, p. 589), (Holzap fel, 2000, pp. 250). It is a classical nonlinear model for compressible hyperelastic materials that is often used for metals. Note that this model is suitable for large displacements but not for large compressive strains. The reason is a horizontal tangent in the stress-strain curve, i.e. zero stiffness in the compression domain.

The ability to compute the reaction forces allows to reproduce this instability within a simple example. In a plain stress analysis a unit square with thickness 1, $E = 100$, $\nu = 0$ is fixed in normal direction on all sides except the upper edge. At the upper edge displacement controlled compression or tension, as well as — in a different investigation — force controlled compression or tension is applied. We plot the displacement versus the reaction force (or the applied force in force control). The displacement-controlled analysis gives the typical picture, Fig. 4.24, of the St.Venant-Kirchhoff material and the two points that prohibit the use of this model in the finite strain compressible range can be clearly made out. The first issue is that as the domain is compressed to zero the reaction forces approach zero also — this makes no physical sense. For a physically reasonable constitutive model the stresses should increase to infinity if the domain is compressed to zero volume. The second point is the minimum at $\overline{u}_y = \frac{1}{3}\sqrt{3} - 1 \approx -0.4226$ with a horizontal tangent and vanishing stiffness. It is no problem to trace the curve in the tension range with force control but in the compression range force control fails to get over the minimum as can be seen in Fig. 4.25 where the automatic step control algorithm reduces the increment resulting in a concentration of points at the minimum.

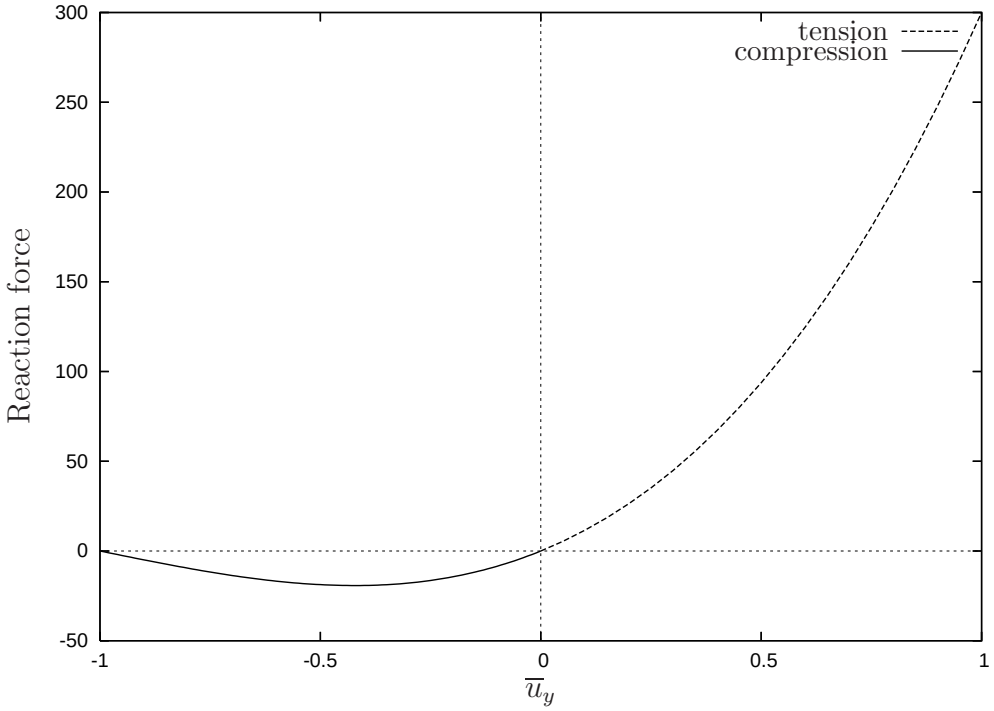


Figure 4.24: Reaction force vs. deformation for compression and tension

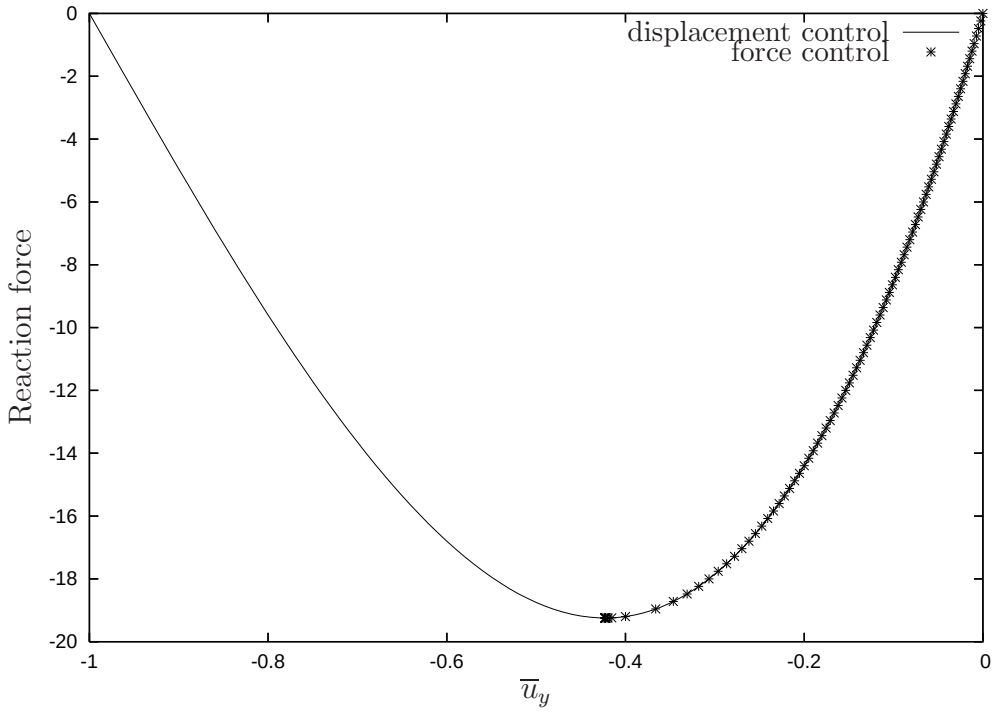


Figure 4.25: Reaction force in compression domain: displacement and force control

Chapter 5

Application, adaption, and numerical studies

After setting out the theoretical foundation in the previous chapters, we now focus on the application of these concepts and the adaptions or enhancements of the p-version code AdhoC (Düster et al., 2004) necessary to perform the numerical studies reported.

First, a finite strain high order axisymmetric element is treated and the implementation is verified by a number of benchmarks. Available analytical solutions for follower loading for hyperelastic material provide the possibility to investigate how displacement based hierarchic finite elements overcome the volumetric locking problem. Finally, the prepared tools are applied for the simulation of powder metallurgy. We start with die compaction processes, move on to cold isostatic pressing and rubber isostatic pressing to end with a validation experiment. It is pointed out, that for materials with evolving internal variables, determined by a nonlinear equation system at each integration point, the concept of (displacement) degrees of freedom must be amended by the number of internal variables present in the structure.

5.1 Finite strain axisymmetric p-version element

A high-order finite strain axisymmetric element was implemented in the p-version code AdhoC to efficiently perform numerical simulations of structures of revolution under axisymmetric boundary conditions.

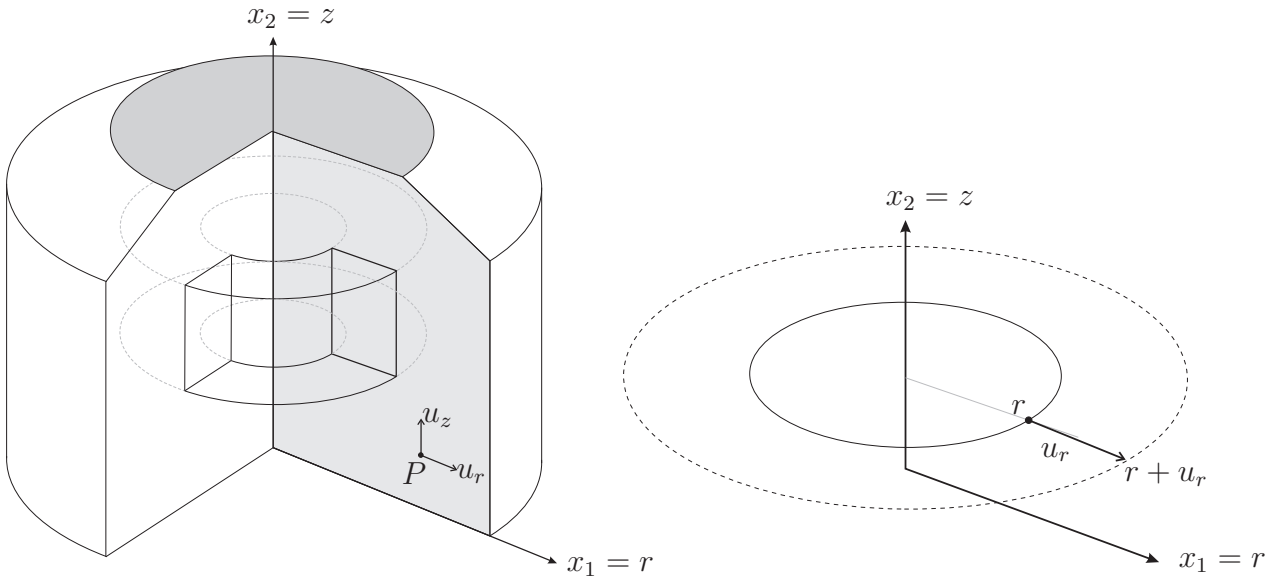
The analysis of structures of revolution by the finite element method can be traced back to the 1960s. As the finite element method was applied in the aerospace industry axisymmetric elements were developed for rocket analysis. The first journal paper on axisymmetric solid elements, by E.L. Wilson, appeared in 1965 (Wilson, 1965) and modeled axisymmetric rocket nozzles and space craft heating shields. There exists an ample literature on axisymmetric finite element analysis. Zienkiewicz and Taylor for example cover some aspects of rotational symmetry in (Zienkiewicz and Taylor, 2000b, Ch. 5) and give examples. Szabó and Babuška (Szabó and Babuška, 1991, Sec. 5.5) state the principle of virtual work and the bilinear form in cylindrical coordinates.

The mathematical structure is very similar to plain strain or plain stress problems as the situation can be reduced to a two-dimensional displacement field. Due to symmetry arguments the two components of displacement in any plane section containing the axis of rotation com-

pletely define the strain and consequently the state of stress. Fig. 5.1(a) shows such a cross section in cylindrical coordinates, the displacement components at point P are denoted as $u_1 = u_r$ in radial and $u_2 = u_z$ in axial direction. The displacement in circumferential direction vanishes by the mentioned symmetry arguments $u_3 = u_\theta = 0$, however, there exist circumferential strains and stresses. A 'quadrilateral element' in the cross section plane corresponds now to a torus as indicated in the figure. As we use cylindrical coordinates and have only two independent displacement components u_r, u_z all volume integrals, e.g. of a function $f(u_r, u_z)$ can be reduced to an integral in the $r - z$ plane by performing the circumferential integration in advance,

$$\int_V f \, \mathrm{d}v = \int_{\theta=0}^{2\pi} \int_{r=r_1}^{r_2} \int_{z=z_1}^{z_2} f(u_r, u_z) \, \mathrm{d}\theta \, r \mathrm{d}r \, \mathrm{d}z = 2\pi \int_{r=r_1}^{r_2} \int_{z=z_1}^{z_2} f(u_r, u_z) \, r \mathrm{d}r \, \mathrm{d}z \, . \tag{5.1}$$

In view of the p-version the displacement components u_r and u_z are discretized by the same high-order shape functions for quadrilaterals introduced in Sec. 4.2.2, using the Ansatz Eq. (4.1). The virtual work for the *plain strain or plain stress* setting depends only on the



(a) Axisymmetric solid element (b) Circumferential strain from u_r

Figure 5.1: Axisymmetric settings

in-plane strains, the component normal to the coordinate plane does not contribute as either the stress or the strain vanishes. This is different in the *axisymmetric* setting. Each radial displacement u_r also causes circumferential strain as can be explained from Fig. 5.1(b). If we assume for simplicity a linear strain measure where the strain is defined by the change in length over the original length, $\varepsilon = \Delta\ell/\ell_0$, a radial displacement u_r corresponds to an increase in circumference of the circle $\Delta\ell = 2\pi(r + u_r - r)$ resulting in the strain induced by a radial displacement u_r as $\varepsilon_\theta = 2\pi u_r/(2\pi r) = u_r/r$. Therefore, the circumferential ('hoop') strain and corresponding stress component has to be considered also. Here, and in the fact that the differential element contains the radius, $r \, \mathrm{d}r \, \mathrm{d}z$, lies the essential difference in the treatment of the axisymmetric situation from plain strain/stress case.

The column vector form of the axisymmetric Green-Lagrange strain tensor is given in Eq. (4.47), of the gradient tensor $\boldsymbol{\Theta}$ in (4.49) and the second Piola-Kirchhoff stress in Eq. (4.60). The resulting strain displacement matrix $\boldsymbol{B}$ (4.88) and gradient displacement matrix $\boldsymbol{G}$ (4.76) contain u_r/r respectively u_1/X_1 terms (X_1 denotes the radial coordinate of a point in reference configuration). These terms are undefined for points on the axis of rotation where $r = X_1 = 0$. If numerical integration using Gauss-Legendre quadrature on quadrilateral elements is used this case can never happen in the pre-processing stage as the Gaussian points always are located in the interior of the element and never on the boundary. This is different if Gauss-Lobatto or Gauss-Radau integration is used, compare 4.6.4 and especially footnote 19 on p. 79.

Remark 16 *In the calculation of stresses for elements that touch the axis of rotation one may encounter during postprocessing the case that the strain is to be evaluated for points on the axis of rotation, $r = 0$. As we do not allow pinholes on the axis of rotation the radial displacement must vanish there, $\lim_{r\to 0} u_r = 0$. For the small strain case we only have the first, linear, term of strains given in (4.47). Consequently we can apply the rule of L'Hospital (Weisstein, 2005) to evaluate the undefined expression*

$$\lim_{r\to 0} \varepsilon_\theta = \lim_{r\to 0} \frac{u_r}{r} \overset{'\frac{0}{0}'}{=} \lim_{r\to 0} \frac{\frac{\mathrm{d}u_r}{\mathrm{d}r}}{\frac{\mathrm{d}r}{\mathrm{d}r}} = \lim_{r\to 0} \frac{\mathrm{d}u_r}{\mathrm{d}r} = \lim_{r\to 0} \varepsilon_r \tag{5.2}$$

and may replace the value of ε_θ by ε_r on the axis of rotation, e.g. for postprocessing evaluations. See (Cook et al., 2001, p. 512) for the 'trick'. For the computer implementation this means, that instead of the third row of the $\boldsymbol{B}$-matrix (4.88) the first row is used in the case that r is smaller than a certain threshold, e.g. 10^{-15}. In the finite strain case this relation does not hold due to the nonlinear terms in (4.47).

5.2 Follower load for the p-version: implementation and verification

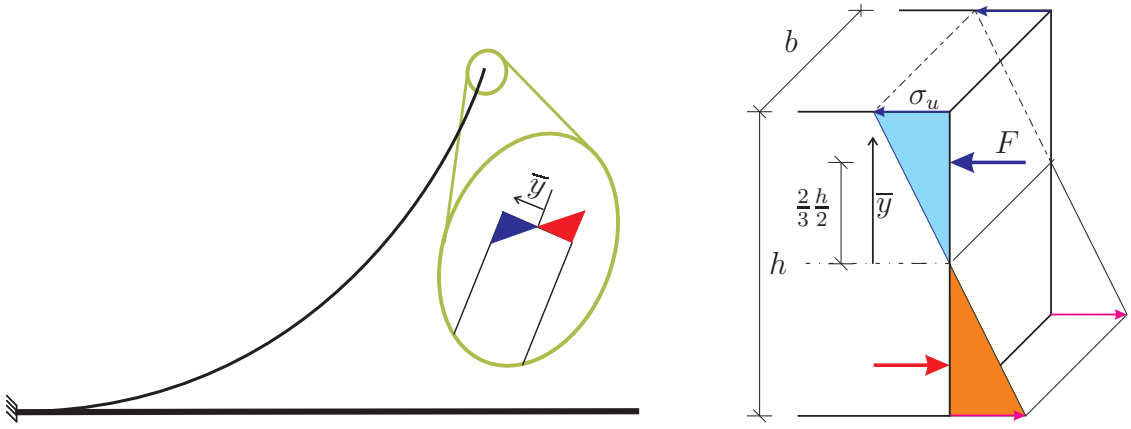
The theory is derived in Ch. 4 leading to the discretized follower load vector (4.118) for 2D problems and (4.116) in the axisymmetric case. As the follower load depends on the deformation this nonlinear dependency is linearized in the view of a Newton-Raphson algorithm. The tangential stiffness matrix for deformation-dependent loading is given for the 2D case in (4.135) and Eq. (4.139) pertains to the axisymmetric setting. All components assembled form the linearized global equation system given in Eq. (4.143).

Once the theory in the framework of the p-version with the characteristic distinction of mapping $\boldsymbol{\phi}$ and deformation $\boldsymbol{u}^h$ is obtained, the implementation in the p-version code AdhoC is straightforward. As in the axisymmetric setting all contributions carry the factor 2π we can omit it, remembering that as a consequence we only integrate over one radian, i.e. $1/2\pi$ of the domain.

5.2.1 Bending strip

The first example uses the St.Venant-Kirchhoff constitutive model. The scope of this constitutive model includes problems with large rotations however only small strains (Wriggers,

2001, pp. 45). It is well known that in the compressible range this constitutive model exhibits non-physical behavior as the strain-stress curve has a horizontal tangent that can lead to instability, i.e. a numerical snap-through, compare Sec. 4.9.2. The critical value for one-dimensional compression with no transversal contraction ($\nu = 0$) is reached when the stretch $\lambda = x/X$ reaches a value of $\frac{1}{3}\sqrt{3} \approx 0.577$. For the bending of the strip reported below this critical value is not reached in the compression region.[1]



(a) Setting for the bending strip example

(b) Calculating the equivalent stress for a given bending moment

Figure 5.2: Bending strip: System and notation

A thin elastic strip is bent to a circle. This example is discussed for example in the context of shells in (Betsch, 1996, p. 67) and p-version setting in (Noel and Szabó, 1997) and (Szabó et al., 2004). The strip is clamped at its left edge, the right edge is subjected to a linearly distributed normal follower traction, cf. Fig. 5.2(a). This traction is chosen equal to a bending moment that distorts the strip to a perfect circle. From the Bernoulli beam theory the curvature ρ is connected to the applied bending moment M, the Young's modulus E and the geometrical moment of inertia I by $1/\rho = M/(EI)$. For a perfect circle formed of a strip with length ℓ the radius is $\rho = \ell/(2\pi)$.

To calculate the normal stress equivalent to the moment that forms the circle we use the variables introduced in Fig. 5.2(b). The resultant force of the compression region for a width b is

$$F = -\frac{1}{2}\sigma_u b \frac{h}{2}\,. \tag{5.3}$$

The total moment of the compression and tension regions relative to the neutral axis

$$M = 2\frac{2}{3}\frac{h}{2}F = -2\frac{h^2 b}{12}\sigma_u \tag{5.4}$$

[1] The stretch in the outer compressed fiber is $\lambda = \frac{2\pi(r-h/2)}{2\pi r} = 1 - \frac{h}{2r}$. To ensure that the critical value of the St.Venant constitutive model is not reached the stretch in the most compressed fiber must fulfill $\lambda = 1 - \frac{h}{2r} > \frac{1}{3}\sqrt{3}$, i.e. $h < 2r(1 - \frac{1}{3}\sqrt{3}) \approx 0.845\, r$. Expressed in the length of the neutral fiber this means, $h < \frac{\ell}{\pi}\frac{1}{3}\sqrt{3} \approx 0.269\,\ell$. For the thin strip investigated here this condition is satisfied.

has to be for a perfect circle

$$M = \frac{EI}{\rho} = \frac{2\pi EI}{\ell}. \tag{5.5}$$

Using $I = bh^3/12$ and equating (5.4) and (5.5) we arrive at

$$\sigma_u = -2\pi \frac{h}{2} \frac{E}{\ell}\,. \tag{5.6}$$

Introducing the coordinate $-\frac{h}{2} \le \overline{y} \le \frac{h}{2}$ perpendicular to the neutral axis we obtain the normal traction equivalent to the moment that bends the strip into a circle

$$t_n(\overline{y}) = -\overline{y}\,\frac{2\pi E}{\ell}\,. \tag{5.7}$$

For the implementation the follower load vector is computed by numerical integration along the loaded edge. At each integration point the local coordinate $-1 \le \xi \le 1$ corresponds to the coordinate $\overline{y}$ and we set the load amplitude in Eq. (4.118) to be $p = \xi\sigma_u$.

This plain stress example with Young's modulus $E = 1$ MPa and Poisson ratio $\nu = 0$ is discretized with 3 plain stress high-order finite elements of height 0.5 mm (lengths of the three elements are $10, 45, 45$ mm) where the aspect ratio of the largest element is length/height $= 90$. The load is applied in 10 equal increments in a total Lagrangian analysis, the final deformation and four intermediate stages are given in Fig. 5.3. This example illustrates that in the p-version very large aspect ratios can be used. The mapping concept can handle large deformations. The edge where the load is applied is rotated 360 degrees relative to its reference configuration. The computations reported were carried out using the trunk space with polynomials of order 8.

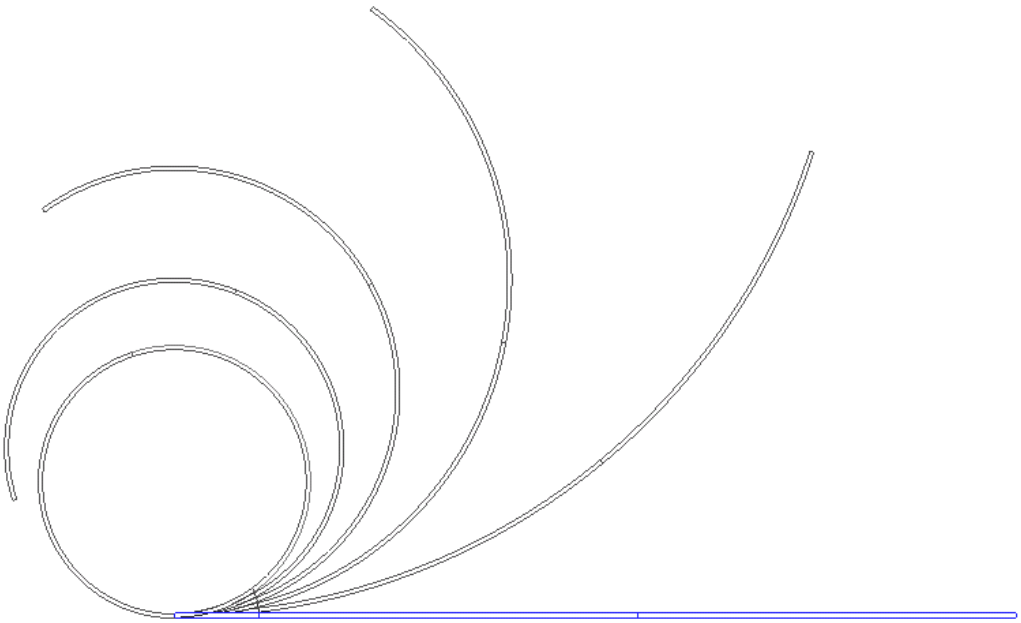


Figure 5.3: Final Deformation and intermediate stages

5.2.2 Pressure on a thin circular plate

Consider a circular plate with radius $r = 1.0$ [mm] and thickness $t = 0.1$ [mm] clamped at its outer edge as shown in Fig. 5.4, cf. (Yosibash et al., 2007). A pressure of $P = 0.01$ MPa is acting on the upper surface, always perpendicular to the boundary's current deformation. We use the compressible Neo-Hookean model (2.68) with the parameters $K = 2000$ [MPa] and $c_{10} = 0.5$ [MPa]. This example problem demonstrates both a change in the loaded area at which pressure is applied, as well as the direction of the pressure, in the terminology of Sec. 3.3.2 we have a body attached follower load. There is no analytical solution available, so a comparison with the h-version code Abaqus/implicit[2] is used in the investigation. We

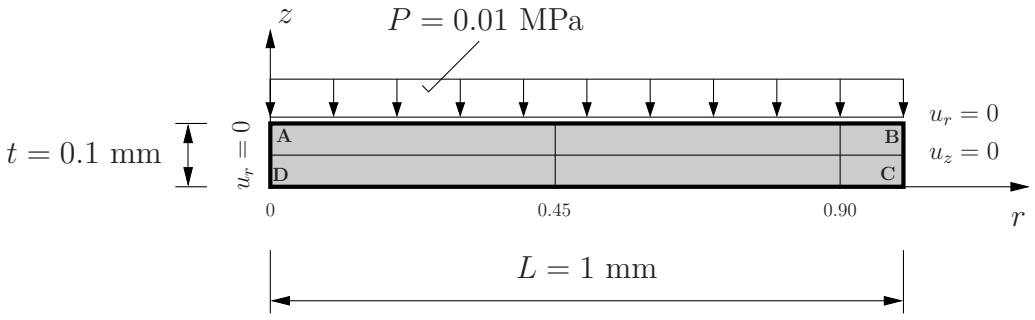


Figure 5.4: Geometry and dimensions for the circular plate

compute the solution by 10 increments with Abaqus obtaining a deflection of 0.182571 [mm] at point A and a deflection of 0.178883 [mm] at point D, using 8-noded linear C3D4 elements (100×10 elements), see Fig. 5.5. The same problem is computed by two different p-FE meshes with AdhoC, a 6-element and 8-element mesh - see Fig. 5.5. The deflections at the upper and lower center of the plate obtained by h-FE and p-FE methods are summarized in Tab. 5.1 and Fig. 5.6.

Table 5.1: Deflections at points A and D for the h-FE and two p-FE meshes

Abaqus			6 el mesh			8 el mesh		
DOF (# el. ; # iter.)	A	D	DOF (p)	A	D	DOF (p)	A	D
315 (2×20 ; 57)	-0.177739	-0.171180	15 (1)	-0.029559	-0.029552	21 (1)	-0.029818	-0.029809
1689 (5×50; 32)	-0.181917	-0.178249	43 (2)	-0.175045	-0.170344	59 (2)	-0.182626	-0.177907
4143 (8×80; 82)	-0.182409	-0.178729	71 (3)	-0.170592	-0.167631	97 (3)	-0.178841	-0.175423
6379 (10×100; 71)	-0.182571	-0.178839	111 (4)	-0.180264	-0.176697	151 (4)	-0.181775	-0.178166
			163 (5)	-0.181809	-0.178141	221 (5)	-0.182367	-0.178686
			227 (6)	-0.182250	-0.178575	307 (6)	-0.182550	-0.178866
			303 (7)	-0.182461	-0.178778	409 (7)	-0.182647	-0.178961

This example shows that the p-FEM implementation of finite deformation follower loads in AdhoC converges to the same result as the Abaqus implementation. Concerning the efficiency, the p-version solution reaches the same accuracy as the h version with significantly less degrees of freedom.

[2]Abaqus is a trademark of ABAQUS, Inc., Rising Sun Mills, Providence, RI, USA.

The help of Zohar Yosibash and Moty Szanto for carrying out the Abaqus computations is acknowledged.

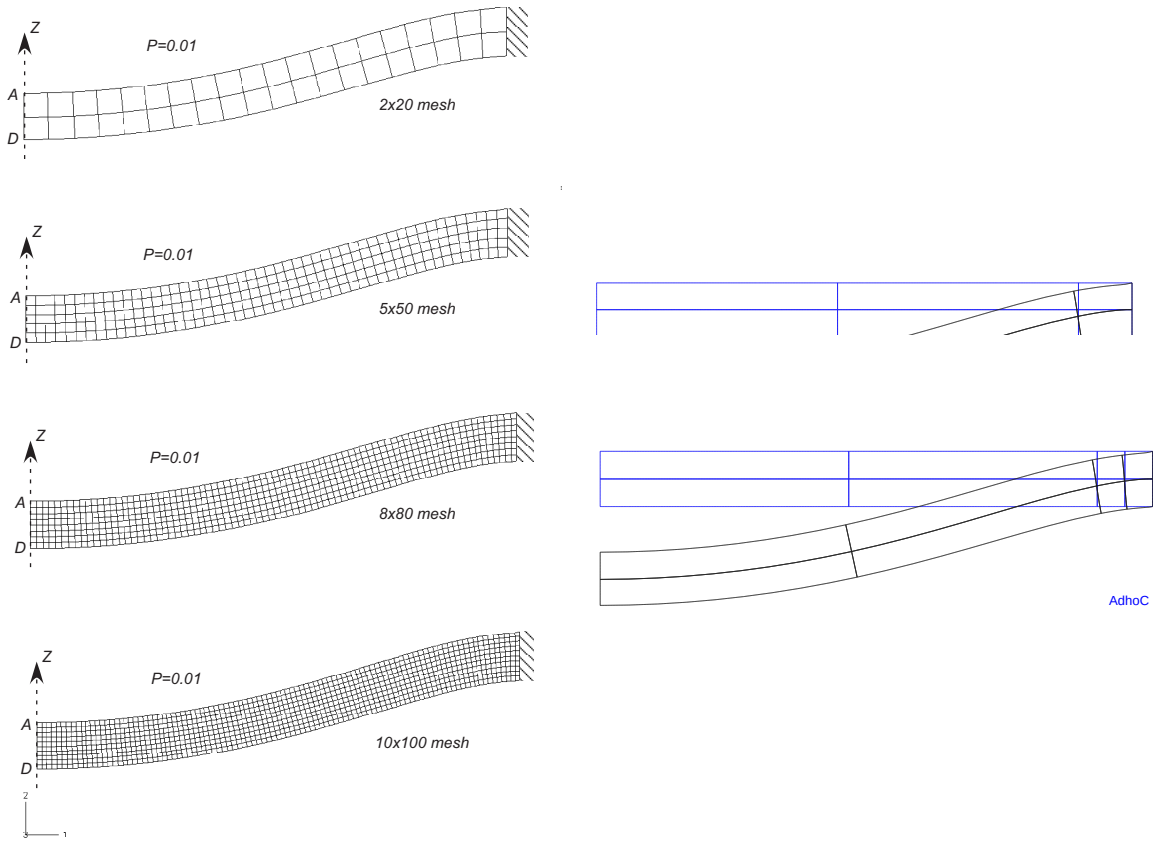


Figure 5.5: Abaqus meshes (left), and AdhoC 6 and 8 elements meshes (right)

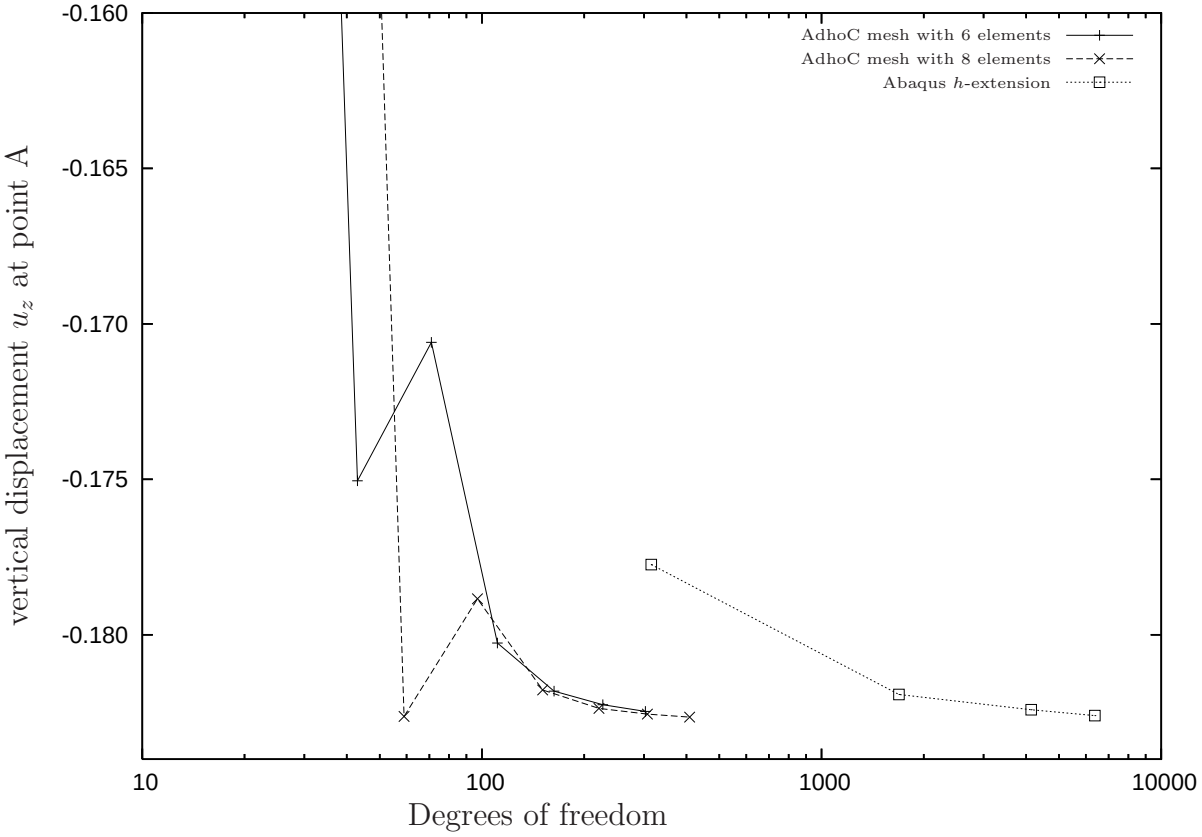


Figure 5.6: Convergence of the p- and h-version calculations

5.3 On volumetric locking-free behavior of p-version hyperelastic finite elements under finite deformations

5.3.1 The locking problem and remedies

The p-version of the finite element method based on the displacement formulation is known to be locking free beyond a moderate polynomial order for nearly-incompressible problems in *linear* elasticity (see (Babuška and Suri, 1992a,b; Suri, 1996) and references therein). Recently, p-FEMs have been shown to be very efficient for finite-deformation problems (Düster et al., 2003), and following (Heisserer et al., 2007; Yosibash et al., 2007) we demonstrate that the locking free property carries over to finite-deformation analyses of nearly incompressible Neo-Hookean and generalized hyperelastic materials.

The locking problem has accompanied and fueled the development of finite elements since the sixties, see for example, the synopsis of Zienkiewicz and Taylor (Zienkiewicz and Taylor, 2000a), Hughes (Hughes, 2000) and Belytschko et al. (Belytschko et al., 2000): the performance of displacement-based low-order elements is known to deteriorate in bending dominated problems and for the nearly incompressible case. The term 'locking' is used since the mid seventies for situations, where the displacement approximated by a finite element discretization is far too small when compared to the exact solution of the mathematical problem. Common to those problems is the presence of constraints (Suri, 1996) where the numerical solution deteriorates as a characteristic parameter approaches a critical limit, e.g. for volumetric locking in linear elasticity as the Poisson ratio $\nu \to 0.5$ or in finite deformation compressible elasticity as the compression modulus $K \to \infty$. For plate-like structures in bending-dominated situations the critical parameter is the element aspect ratio of the element thickness over a typical element length $t/h \to 0$. The critical parameter gives the name for the type of locking and shear, membrane and volumetric locking is distinguished.

First there were heuristic approaches to overcome those defects, in the mean time there exist mathematical investigations and definitions. It has to be pointed out that locking depends on the error measure considered: elements might show no locking in energy norm but deteriorate significantly in the pointwise error of stresses. A heuristic method to quantify locking is the constraint count proposed by Hughes (Hughes, 2000, p. 209), where the number of equations is compared to the number of constraints. A sound mathematical definition and assessment of locking and robustness is given by Babuška and Suri (Babuška and Suri, 1992a,b; Suri, 1996).

Remedies against locking were first developed in the setting of geometrical linear (small strains) linear elasticity and are still investigated for nonlinear problems. Three basic approaches can be pointed out: increasing the order of the polynomial approximation (p-FEM), reduced integration with stabilization and the field of mixed methods. We show here the locking-free property of displacement based p elements for *finite* deformations. To provide the context, some milestones for the other approaches are also mentioned.

Reduced integration and stabilization. An simple yet effective approach is to use reduced or selective integration, for early references see Doherty et al. (Doherty et al., 1969), Zienkiewicz et al. (Zienkiewicz et al., 1971) and Hughes et al. (Hughes et al., 1977). The idea is to eliminate the 'parasitic' stresses or strains responsible for locking by a reduced or selective integration of the corresponding terms in the stiffness matrix. The notion 'parasitic' describes strains and stresses that do not exist in the exact solution (Bischoff, 1999, Sec. 6.4).

For a mathematical explanation see Arnold (Arnold, 1981). The major drawback of reduced integration is that the resulting stiffness matrices are rank-deficient and un-physical zero-energy modes introduce oscillatory errors, the so-called 'hourglass' effect.

Stabilization procedures were developed by Flanagan, Belytschko and co-workers (Flanagan, 1981), (Belytschko et al., 1986; Belytschko and Tsay, 1983) based on the theoretical insight provided by (Malkus and Hughes, 1978) and (Koslo and Frazier, 1978), who showed the equivalence of selective integration and mixed methods under certain conditions. The stabilization is achieved by adding additional stiffness.

Mixed methods. Different from the classical approach, where the displacements field is the only primary variable, the Hu-Washizu functional introduces strain and stress as further independent fields. From a mathematical point of view, the constrained minimum-problem is transformed to a saddle-point problem (Suri, 1996). The direct application of the three-field Hu-Washizu principle is complicated (Belytschko et al., 2000) and there were attempts to reduce the number of primary variables. In the hybrid multi-field methods a set of variables is statically condensed on element level. The 'assumed strain' approach is preferred over the 'assumed stresses', as most constitutive models are strain-driven. Two paths are followed, either to *enhance* the strain field so parasitic strains are compensated or to *eliminate* parasitic strain terms. The enhanced assumed strain (EAS) method proposed by Simo and Rifai (Simo and Rifai, 1990) includes the method of incompatible modes by Wilson et al. (Wilson et al., 1973) as a special case. Under certain loading conditions the EAS method shows hour-glassing. Several strategies have been proposed in Bischoff et al. (Bischoff et al., 1999), Reese, Simo, Wriggers and co-workers (Reese et al., 1999; Reese and Wriggers, 2000; Simo et al., 1993; Wriggers and Korelc, 1996; Wriggers and Reese, 1996) to overcome this drawback. The EAS method was generalized to finite deformations by Simo and Amero (Simo and Armero, 1992).

The other path eliminates parasitic strains and can be summarized under the term 'B-bar' approach, cf. Simo and Hughes (Simo and Hughes, 1986). The assumed natural strain (ANS) method is used by Hughes and Tezduyar (Hughes and Tezduyar, 1981) to avoid shear locking and was extended to nonlinear problems by Dvorkin and Bathe (Dvorkin and Bathe, 1984). The discrete strain gap (DSG) method proposed by Bletzinger and co-workers (Bletzinger et al., 2000; Koschnick, 2004) generalizes the ANS concept.

High-order methods. Since the very beginning when locking was recognized, high-order methods were proposed to overcome it, see e.g. Irons in 1966 (Irons, 1966). Zienkiewicz and Taylor (Zienkiewicz and Taylor, 2000b, p. 320) report that high-order elements are applied with excellent results for incompressible problems (Arnold, 1981; Vogelius, 1983), but would pose other difficulties and are thus seldom used in practice. In the mean time most difficulties are resolved, see the monographs of Szabó and Babuška (Szabó and Babuška, 1991) and Schwab (Schwab, 1998). There exist mature commercial high-order finite element codes like StressCheck (Engineering Software Research & Development, 2006) and a number of academic codes like AdhoC (Düster et al., 2004) so the theoretical benefits of high-order elements for locking problems shown by Babuška and Suri (Babuška and Suri, 1992a,b; Suri, 1996) can be exploited for practical applications. For geometrical linear thin Reissner-Mindlin plates it was observed that a moderate polynomial degree of $p = 4$ is sufficient to avoid shear locking, see for example Holzer et al. (Holzer et al., 1990), Rank et al. (Rank et al., 1998b), Szabó and Babuška (Szabó and Babuška, 1991). Nübel (Nübel, 2005) demonstrates the robustness of geometrical linear p-FEM used in nearly incompressible problems also for the deformation

theory of plasticity.

This section shows that the locking-free properties of p-FEM carry over to geometrical nonlinear problems of near incompressibility, where hyperelastic constitutive models are used. We investigate the Neo-Hookean model problem of a sphere under internal pressure where a semi-analytical solution is provided in (Yosibash et al., 2007). It serves as a benchmark problem against which p-FEM solutions are compared to demonstrate that an increase in polynomial order vanquishes locking, and to explain amplification of the errors in the stresses for near incompressibility.

5.3.2 Thick-walled sphere under internal pressure

The semi-analytical solution obtained in (Yosibash et al., 2007) to an accuracy in the displacements of 10^{-8} allows to compute the average relative error of the numerical finite element solution in displacement and stress along a radial cutline. This kind of 'norm' is used to study locking. A first investigation discusses that p-refinement can overcome locking even on very coarse grids. In a second study the performance with respect to locking of h-refinement with fixed (moderate) polynomial degree is explored.

5.3.2.1 First investigation: p-extension

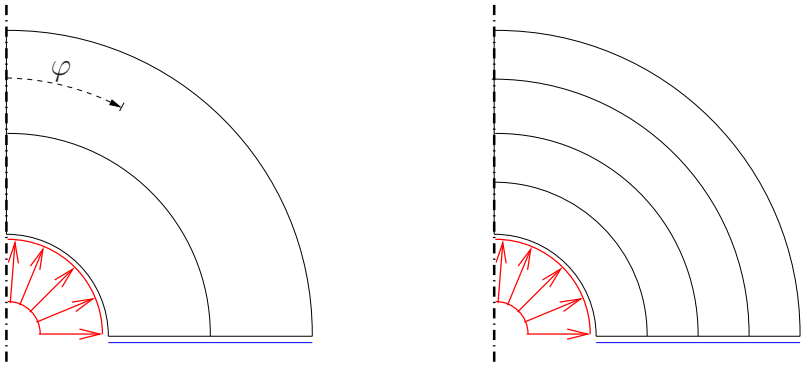
The thick-walled sphere under internal pressure is discretized by 2 and 4 p-axisymmetric elements (see meshes in Fig. 5.7) using blending functions (Sec. 4.3) for an exact representation of the circular curves. The calculations are carried out with the academic p-finite element code AdhoC. The main purpose of this example is to demonstrate the convergence characteristics of p-FEMs for progressively more incompressible materials as defined in (2.68) with $c_{10} = 0.5$ MPa. This is realized by raising the compression modulus $K = 10^k$ MPa, $k = 1, \ldots, 5$. The number of degrees of freedom is increased by uniformly incrementing the polynomial order from 1 to 9 yielding in the 2-element case 6, 16, 26, 40, 58, 80, 106, 136 and 170 degrees of freedom. Throughout all runs 15×15 integration points for each element were used to rule out any significant effects of the quadrature rule for comparison.

Fig. 5.8 shows the average relative error in displacement u_r in percent versus the degrees of freedom calculated from sampling the data at 101 points along a radial cutline

$$\text{Avg. error} \doteq \frac{1}{101} \sum_{j=1}^{101} \left| \frac{u_r^{\text{EX}}(R_j) - u_r^{\text{FE}}(R_j)}{u_r^{\text{EX}}(R_j)} \right| \times 100 \tag{5.8}$$

on a log-log scale. As reference the semi-analytical solution was evaluated at the sampling points. The average error in radial stress σ_{rr} in percent calculated in the same manner is shown in Fig. 5.9. Each of the dots corresponds to an increment in the polynomial order of one. From the plot of the error in displacement it is clearly visible that total locking for the case of 2 elements and $K = 100\,000$ MPa is present until $p = 4$, afterwards the solution converges rapidly and with $p = 5$ the average error in displacement is already down to less than 2 %. If 4 elements are used $p = 4$ corresponds to an error of about 6 % while the error decreases to 0.1 % for $p = 5$.

From Fig. 5.9 we can see, that the relative error in stress is amplified as the material becomes more and more incompressible. This can be readily explained by noting that in



(a) 2 element p-FE mesh (b) 4 element p-FE mesh

Figure 5.7: Axisymmetric meshes with curved elements and symmetry boundary condition. Meshes are classified by the number of divisions along the arc and in radial direction. The examples shown here are denoted 1x2 and 1x4.

Eq. (5.9), cf. (Yosibash et al., 2007), governing the radial stresses any error in the placement $r = f(R)$ or its derivative violating the incompressibility constraint is magnified by K, in our examples K up to 100 000 MPa for the nearly incompressible case,

$$\sigma_{rr} = K\left(-1 + \underbrace{\frac{f^2 f'}{R^2}}_{\det \mathbf{F}}\right) - \frac{4c_{10}\left(f^2 - f'^2 R^2\right)}{3\left(\frac{f^2 f'}{R^2}\right)^{\frac{5}{3}} R^2}. \tag{5.9}$$

The relative error in stress along a radial cutline in Fig. 5.10 shows that this is not only true in an average sense, but also pointwise.

Remark 17 *The expression (5.9) describing the radial stresses contains a term* $(-1 + \det \mathbf{F})$ *that is multiplied by the compression modulus* K*. Note the similarity to the penalty function method for constraint enforcement described in Sec. 4.9.1.3. The penalty parameter* κ *can be identified with the compression modulus* K *and the constraint equation* $\boldsymbol{C}_c$ *(4.249) corresponds here to* $(-1 + \det \mathbf{F})$*. Approaching the incompressible case, i.e. when* K *is large, the volume must remain constant which is equivalent to the constraint* $\det \mathbf{F} = 1$*. Hence the term in the brackets must vanish. If small errors in the solution* f *or its derivative violate this condition, the error is amplified by large* $K >> 1$ *for nearly incompressible materials.*

It is interesting to note, that with a moderate number of 170 degrees of freedom ($p = 9$) and only 2 elements, even for the nearly incompressible case of $K = 100\,000$ MPa, an average relative error in u_r smaller than 0.000 32 % is obtained, while the large K results in an average error in the stress σ_{rr} of about 6.5 %. However, in the stresses there is rapid convergence for polynomial order $p > 4$ as well (although still at a high level). As one would expect from Eq. (5.9) if the bulk modulus is increased by one magnitude, also the error in stresses grows approximately one magnitude, this can be verified in Fig. 5.10 and Table 5.2 (compare the values for $p = 9$).

Fig. 5.11 shows how the pointwise deviation from incompressibility ($\det \mathbf{F} = 1$) is decreasing as the compression modulus grows. Using the *in*compressible analytical solution given

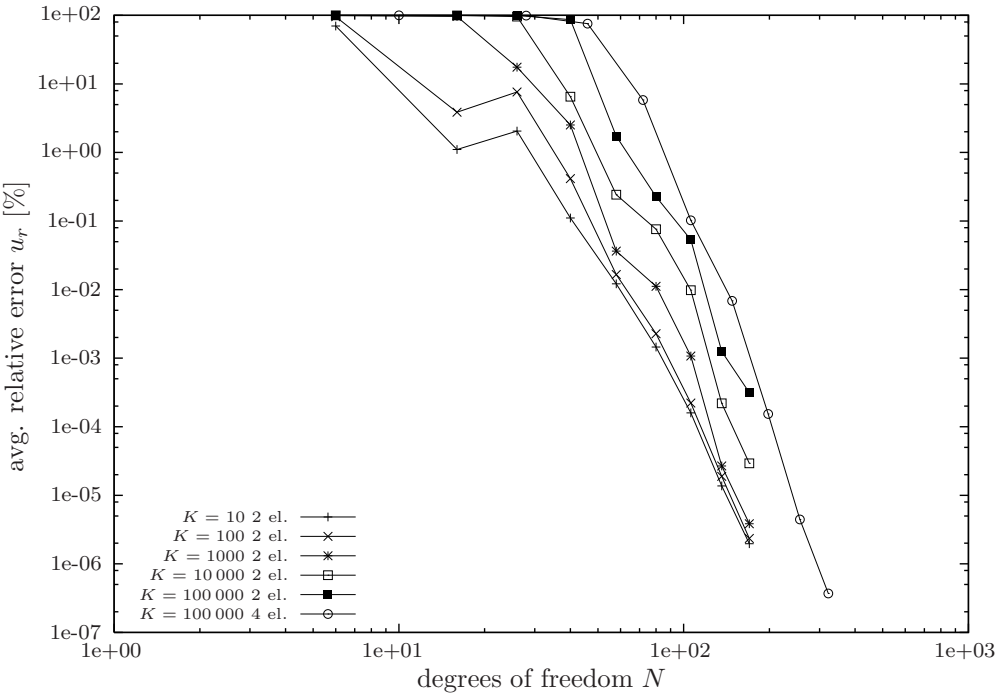


Figure 5.8: Average relative error in u_r [%]

Table 5.2: Sphere under internal pressure: DOF (polynomial order p) and average relative errors in percent in u_r and σ_{rr} for p-FE solutions

DOF (p) 2 elements	Avg. Error (u_r) % $K = 10$	Avg. Error (u_r) % $K = 100$	Avg. Error (u_r) % $K = 1000$	Avg. Error (u_r) % $K = 10\,000$	Avg. Error (u_r) % $K = 100\,000$
6 (1)	6.983280846E+01	9.5269932E+01	9.907157826E+01	9.981561118E+01	9.997844902E+01
16 (2)	1.101921483E+00	3.8662678E+00	9.646173999E+01	9.942663224E+01	9.986553204E+01
26 (3)	2.055641371E+00	7.6180657E+00	1.756328147E+01	9.595459426E+01	9.974518752E+01
40 (4)	1.108291789E-01	4.1478166E-01	2.521988004E+00	6.506828996E+00	8.661859778E+01
58 (5)	1.212597316E-02	1.6560222E-02	3.635192519E-02	2.417838390E-01	1.727937377E+00
80 (6)	1.452696466E-03	2.2684369E-03	1.115035145E-02	7.609565235E-02	2.270126864E-01
106 (7)	1.595114879E-04	2.2172175E-04	1.075499406E-03	9.814558987E-03	5.323631678E-02
136 (8)	1.370535820E-05	1.8897797E-05	2.677895669E-05	2.198140702E-04	1.251180778E-03
170 (9)	1.964790025E-06	2.3470184E-06	3.853879283E-06	2.920482985E-05	3.145745740E-04
DOF (p) 2 elements	Avg. Error (σ_{rr}) % $K = 10$	Avg. Error (σ_{rr}) % $K = 100$	Avg. Error (σ_{rr}) % $K = 1000$	Avg. Error (σ_{rr}) % $K = 10\,000$	Avg. Error (σ_{rr}) % $K = 100\,000$
6 (1)	3.006689337E+02	3.6220284E+02	3.276664797E+02	2.462204001E+02	2.229869114E+02
16 (2)	2.465804728E+02	1.8789976E+03	7.591388343E+02	5.637689561E+02	5.328481905E+02
26 (3)	1.337053119E+02	3.0415066E+02	3.804915473E+03	3.068254348E+03	2.564841507E+03
40 (4)	1.547535621E+01	1.5451258E+02	1.487362855E+03	1.105656722E+04	2.290003269E+04
58 (5)	1.511532528E+00	1.2900826E+01	1.353470545E+02	1.367770978E+03	1.029684008E+04
80 (6)	2.842779005E-01	1.8132423E+00	1.795774934E+01	1.748979049E+02	1.803357358E+03
106 (7)	3.098556237E-02	2.1263489E-01	2.031664676E+00	2.526832023E+01	4.039467240E+02
136 (8)	2.658960948E-03	1.7439388E-02	1.657037181E-01	1.433039045E+00	1.284992373E+01
170 (9)	6.173770274E-04	4.4520136E-03	4.174696015E-02	3.993825467E-01	6.420052203E+00

in the appendix of (Yosibash et al., 2007), we can assess how 'incompressible' the solution is for a given compression modulus K. Notice that incompressible solutions are provided for a series of problems for which the internal displacement and pressure are computed that satisfy the equilibrium equations. Fig. 5.12 shows the relative difference in internal pressure in percent between the (compressible) numerical p-finite element solution and the incompress-

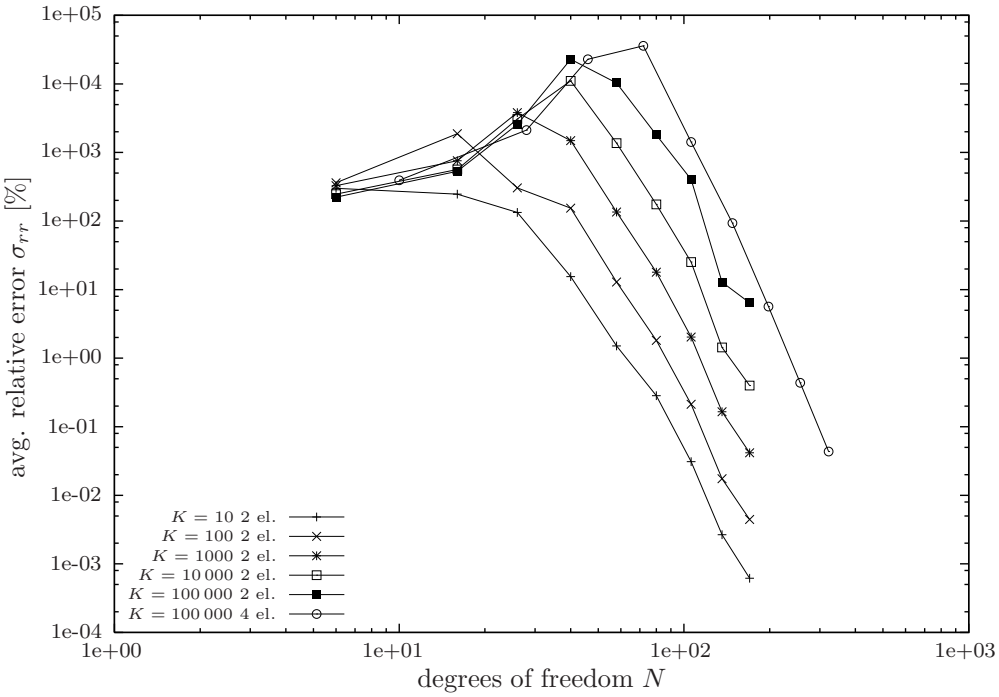


Figure 5.9: Average relative error in the radial stresses σ_{rr} [%]

ible reference for a given internal displacement $u_r(R_i)$. For the computations with K up to 10 000 MPa a polynomial order $p = 9$ was sufficient. If the compression modulus reaches $K = 100\,000$ MPa the oscillations on a very small scale noticeable in Fig. 5.12(b) vanish if p is raised to 13. In this latter case the plot shows that a relative difference between the compressible semi-analytical solution and the incompressible solution (denoted by the diamond shape) matches the numerical solution.

Following Suri (Suri, 1996), we can conclude that p-finite elements are locking-free in the presented finite deformation example as the error curves for displacements and stresses in Fig. 5.8 and Fig. 5.9 remain parallel and converge for $p > 4$ also for very high compression moduli K.

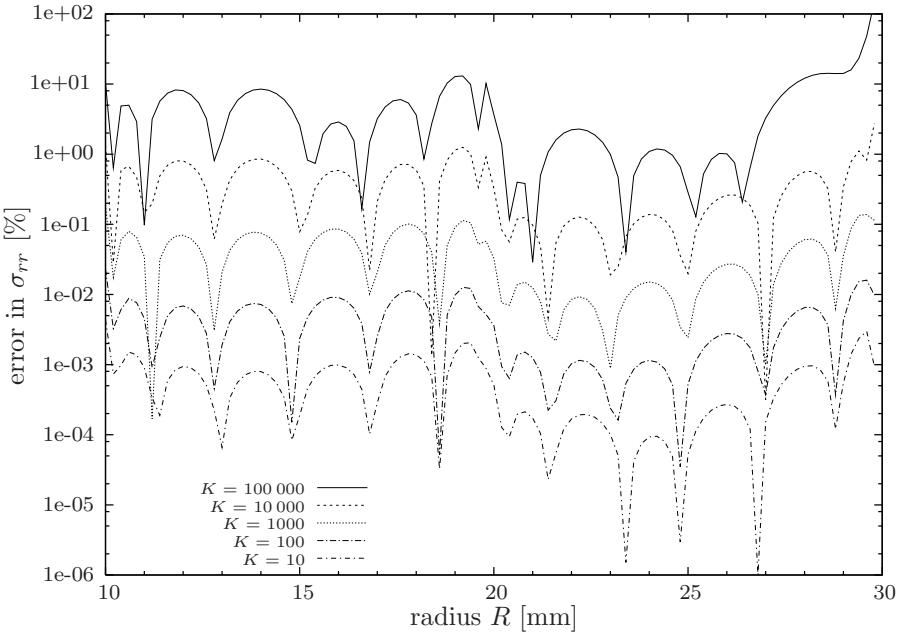


Figure 5.10: Mesh with 2 elements, $p = 9$: relative error in σ_{rr} [%] for increasing K

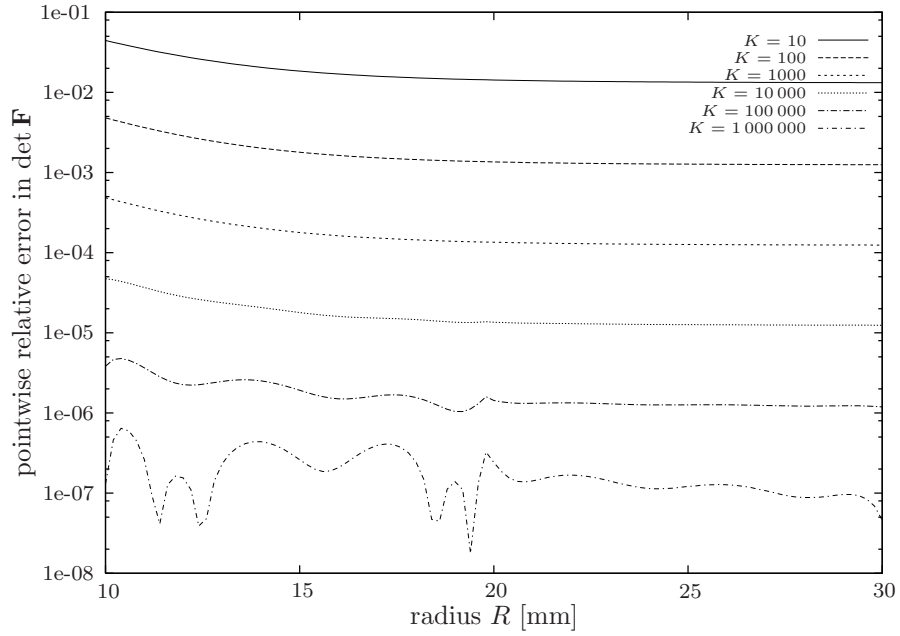
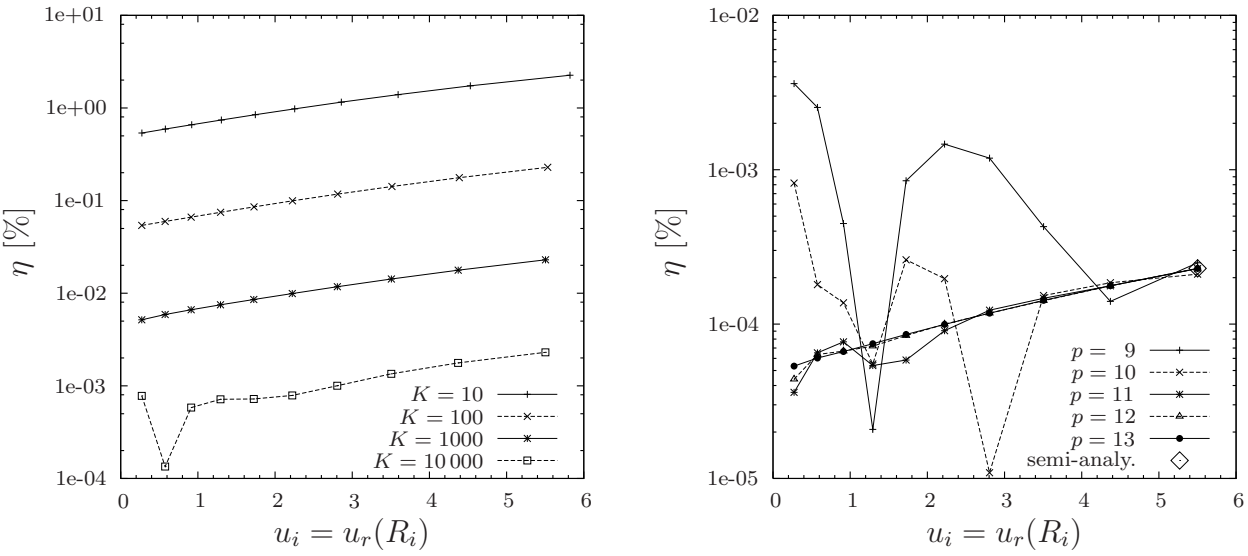


Figure 5.11: Pointwise error in the incompressibility constraint $|\det \mathbf{F} - 1|$ for $p = 9$



(a) $p = 9$, $K = 10 \rightarrow 10\,000$ MPa (b) $p = 9 \rightarrow 13$, $K = 100\,000$ MPa

Figure 5.12: The relative difference $\eta = \left| \frac{P^{\mathrm{FE}}(u_i) - P^{\mathrm{IC}}(u_i)}{P^{\mathrm{IC}}(u_i)} \right| \times 100$ in internal pressure P between the compressible FE solution (2 element mesh) and the incompressible analytic solution as a function of the displacement $u_i = u_r(R_i)$. η decreases as K is increased. Note that in (b) the semi-analytic compressible solution for $K = 100\,000$ MPa (computed within an accuracy of 10^{-8} in u_i) has an apparently same small relative difference compared to the incompressible result as the FE solution for $p = 13$.

5.3.2.2 Second investigation: h-extension for fixed p

After the study of p-refinement, we investigate the locking phenomenon of the hyperelastic sphere under internal pressure concerning h refinement with fixed moderate polynomial degree. In the following, we only report results for the bulk modulus $K = 10\,000$ MPa. The features discussed here are representative also for the other values of K that were used in Sec. 5.3.2.1.

The basic plot for the discussion of variants is Fig. 5.13. Here the average error in displacement u_r as defined in (5.8) along a radial cutline is plotted in double logarithmic scale versus the degrees of freedom. The lines with the crosses correspond to h-refinement for different fixed polynomial degrees p (one element in arc direction and refinement in radial direction, the sequence of meshes is denoted as 1xN, $N = 1, \ldots, 10$, i.e. 1x1, ... , 1x10). The lines marked with triangle, circle and square show p-refinement, $p = 1, ...9$, for a fixed mesh. The higher convergence rate of the p-extension is evident. For h-refinement and $p = 3$ locking is overcome and slow convergence is visible, however the error on the 1x10-mesh is still larger than 10%.

If also divisions in arc direction are used, compare Fig. 5.14, a better convergence is recovered in the beginning however the curves level off at a small slope and for the 7x9 mesh the error is only little below 0.1%. To clarify this effect look at Fig. 5.15, where an analogous plot is given for $p = 5$. Here, there is also an almost horizontal branch visible for the 1xN-meshes but an increasing slope is recovered if the arc is subdivided.

From this it is apparent, that two effects contribute to this leveling off. First the distortion of the elements dictates a certain polynomial degree to render a certain accuracy. This can already be studied in the base plot Fig. 5.13 if the polynomial degree is raised on a certain mesh, e.g. the 1x4-mesh. Second this leveling off indicates an error, that is induced by the exact description of the circular arcs by trigonometric functions. For example, with $p = 5$, Fig. 5.15, the fading out that is present on the 1xN and 2xN-meshes is overcome with increasing arc-refinement. This is in line with the growing ability to capture rigid body rotations with increasing polynomial order of the shape functions when the boundaries of the domain are described with non-polynomial functions, cf. Sec. 4.3.2, like in the present example.

If an h-refinement for the high polynomial degree $p = 7$, Fig. 5.16, is carried out a different type of error comes to light. The geometric error connected to the exact description of the circular arcs is almost gone, when the arc-direction is divided twice but the curves still level off and no accuracy higher than about 10^{-8}% can be reached. This is caused by the so called model error, Sec. 4.4. As the semi-analytic reference solution described in (Yosibash et al., 2007) was determined to an accuracy in displacements of approximately 10^{-8} there is a threshold for any error measure based on this reference solution that can not be gone below. Similar effects are discussed in (Nübel, 2005, Sec. 6.2.2.2).

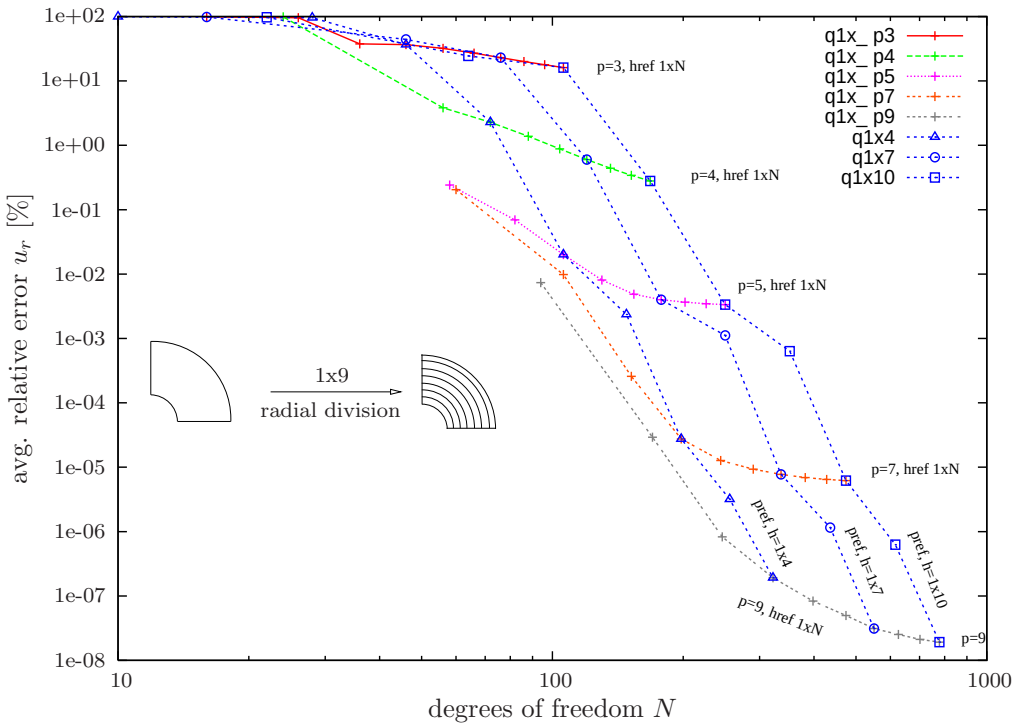


Figure 5.13: $K = 10\,000$ MPa: Comparison of h-extension with fixed p and p-extension on a fixed mesh

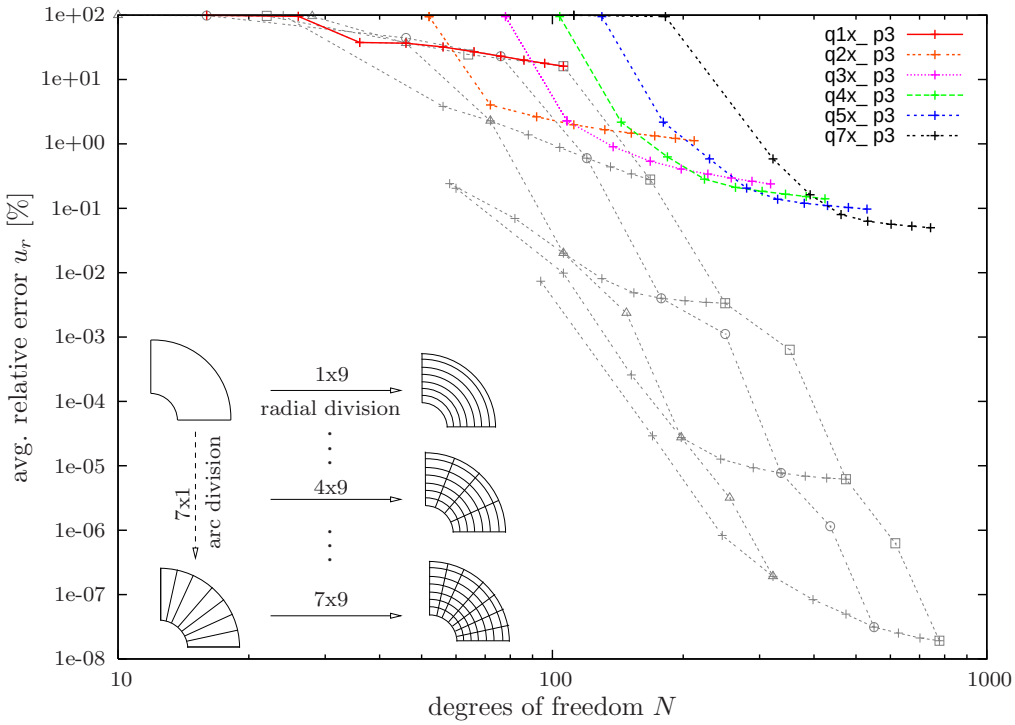


Figure 5.14: $K = 10\,000$ MPa, $p = 3$: h-refinement – influence of arc-divisions

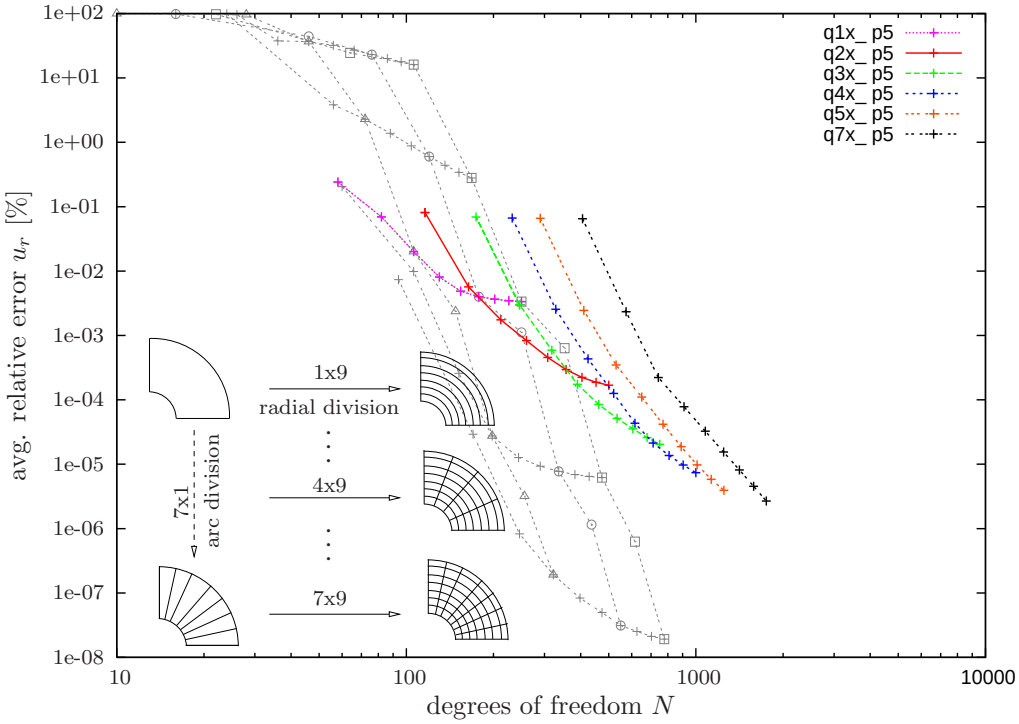


Figure 5.15: $K = 10\,000$ MPa, $p = 5$: h-refinement – geometric error

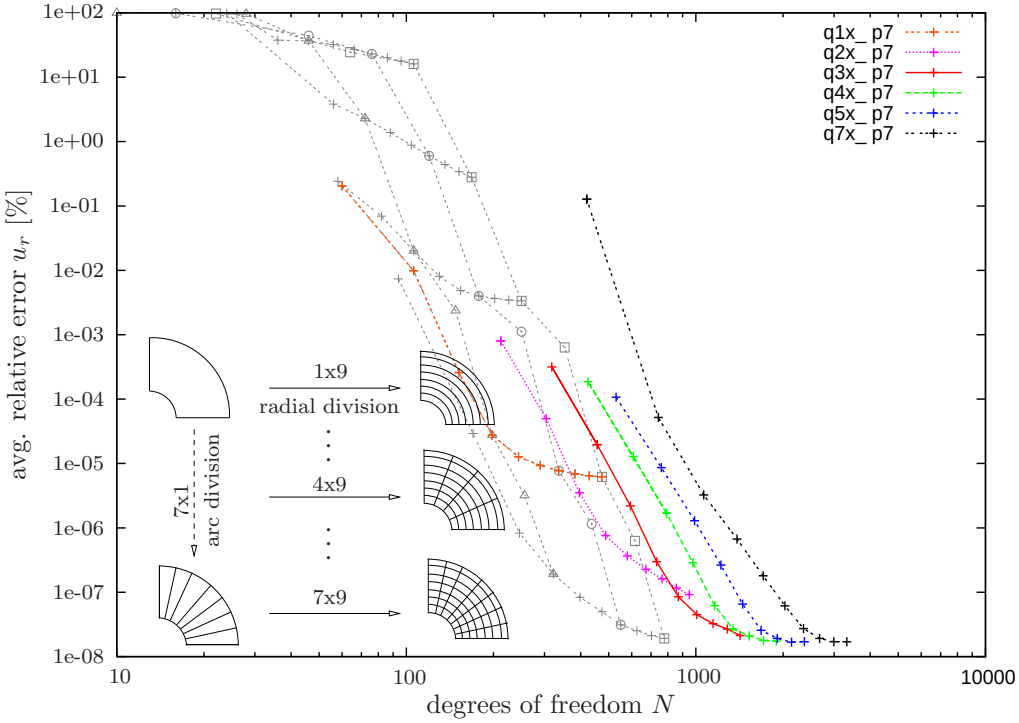


Figure 5.16: $K = 10\,000$ MPa, $p = 7$: h-refinement – model error

5.4 Simulation of metal powder die compaction and cold isostatic pressing

5.4.1 Die compaction examples

5.4.1.1 Cylinder

A cylinder made of copper powder with initial relative density (see Eq. (2.32)) of $\rho_{\mathrm{rel0}} = \rho_{\mathrm{R}}/\rho_0 = 0.42$ is compacted under displacement control. The reference density ρ_0 is the density of solid copper while ρ_{R} is the density of the powder body in the initial state. Exploiting the radial symmetry we use axisymmetric elements, see Fig. 5.17 for the system and the boundary conditions. This example is analogous to the die compaction experiments that were conducted to determine the parameters of the constitutive model in (Bier et al., 2007). Due to the homogenous deformation, i.e. the deformation gradient is the same for all points of the body, $\mathbf{F}(\vec{X}, t) = \mathbf{F}(t)$ cf. (Haupt, 2000, p. 26), one element with linear Ansatz functions is sufficient. The current relative density ρ_{rel} is calculated according to (2.32).

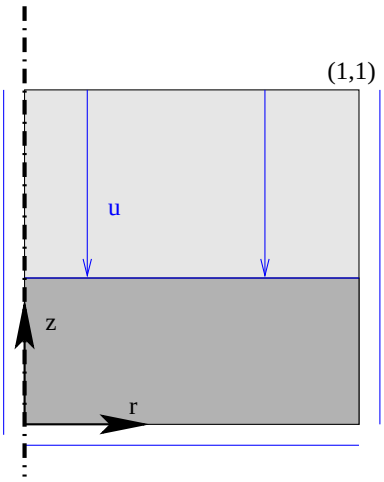


Figure 5.17: System and boundary conditions for the die-compaction example with displacement control

This example serves mainly to verify the proper implementation of the stress algorithm including unloading and reloading. At the end of a partial run the current state is written to the mesh file storing the element displacement vectors and the internal variables at all integration points. In the restart procedure these quantities are read and used e.g. to compute the tangential stiffness matrix for the next loading process. This restart feature is necessary for more complex processes as described in Sec. 5.4.4.1. Fig. 5.18 displays the evolution of the axial and radial Cauchy stress vs. the relative density. It can be seen that the reloading follows exactly the elastic unloading path. As the sample is clamped in radial direction, an unloading in axial direction decreases the axial stress σ_z to zero while there remain radial stresses due to the boundary conditions.

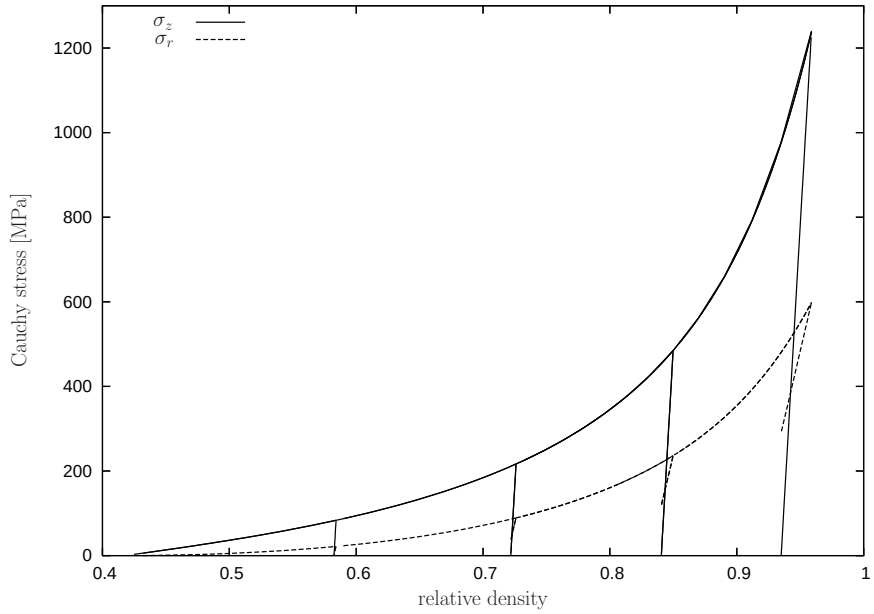


Figure 5.18: Axial stress σ_z and radial stress σ_r vs. relative density ρ_{rel}

5.4.1.2 L-shaped domain

The second example is a rotationally symmetric structure with L-shaped cross section. Axisymmetric elements are used to discretize the system shown in Fig. 5.19. Displacement controlled pressing at the top and bottom introduces an inhomogeneous deformation (final $\overline{u}_{\text{bottom}} = 2.4 = -0.8\overline{u}_{\text{top}}$), cf. Fig. 5.20. The reentrant corner allows to study the influence of a singularity.

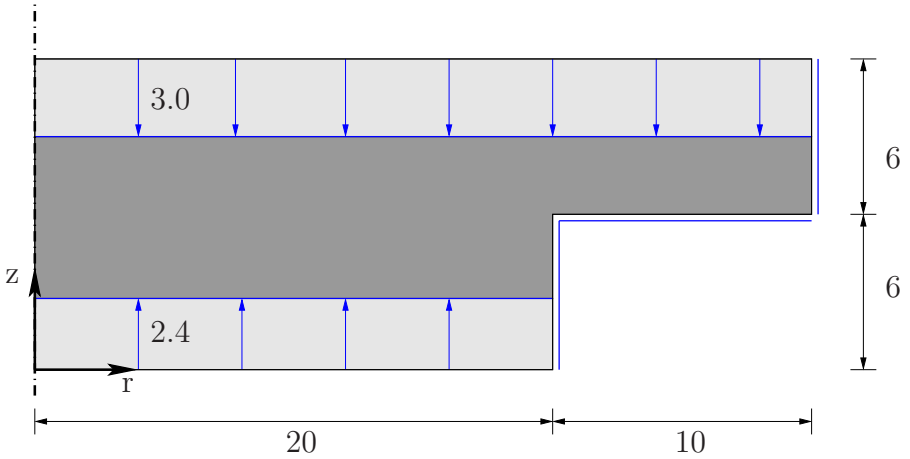


Figure 5.19: System and boundary conditions for the L-shape example with displacement control

The deformed body is depicted in Fig. 5.20 and the distribution of the relative density is shown in Fig. 5.21.

It is well known for *linear* problems with corner singularities how to construct a suitable mesh for the p-version of the finite element method (Szabó and Babuška, 1991). The same

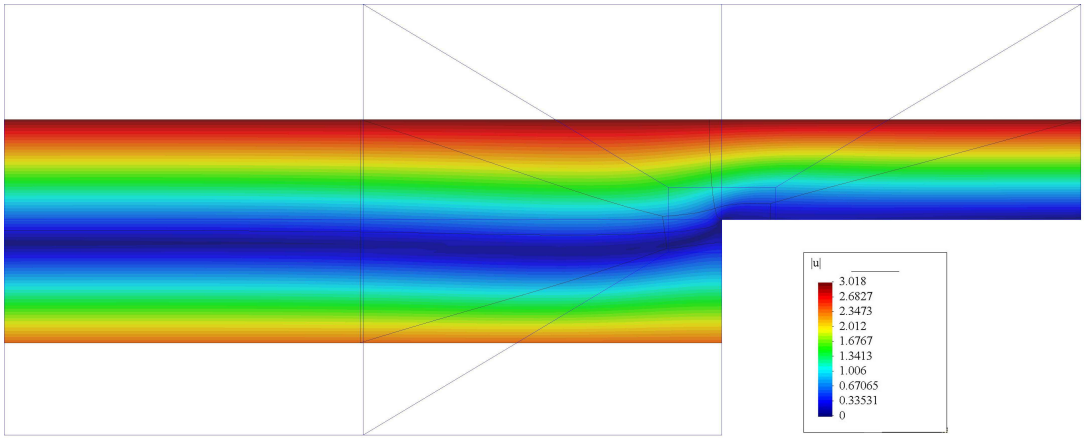


Figure 5.20: Inhomogeneous deformation, shown is $|\vec{u}|$. The background mesh depicts the undeformed configuration.

mesh layout is applied for this geometric and material nonlinear problem. We compare the p-version approach where the polynomial degree of the hierarchic Ansatz functions is raised on a fixed mesh graded once towards the reentrant corner with a geometric progression factor of 0.15 shown in Fig 5.22(a) (referred as 'graded mesh') to meshes with an uniform subdivision of the base elements and fixed low polynomial degree as shown in Fig 5.22(b). This uniform h-refinement subdivides each of the 5 elements of the base mesh in up to 41×41 sub-elements ('h-meshes'), Fig. 5.22(c) shows an example where each base element is divided in 15×15 elements.

Elasto-plastic ($\eta = 0$) and inelastic computations ($\eta > 0$) were carried out for both mesh layouts to study the convergence in terms of stresses, see Tab 2.4. As a constitutive model with internal variables is employed, the stresses are initially only available at the Gauss points. To be able to extract the stress at the point $r = 5, z = 9$ for all used meshes without further processing, an odd number of Gauss-Legendre points is chosen resulting in an integration order in each spatial direction of $p+2$ for odd polynomial orders p and $p+3$ for even p. Details are given when the different extension strategies are discussed.

The number of degrees of freedom is increased on the graded mesh by raising the polynomial degree of the Ansatz functions, while on the h-meshes uniform subdivision of elements was done and a moderate polynomial degree $p = 2$ and $p = 3$ was used.

For elastoplasticity ($\eta = 0$) we found that the h-refinement approach comes to the limit when the mesh is very fine. For $p = 2$ and an integration order of $p+3 = 5$ the stress algorithm is not able to find a solution to the local nonlinear equation system for some integration points very close to the singularity. The global step size is consequently reduced over and over to a level that rendered calculations infeasible. This point was reached when during the h-refinement process[3] each base element was divided in 35×35 subelements, corresponding to 6125 elements and about 49.000 degrees of freedom with $p = 2$.

Therefore, the viscoplastic formulation was chosen with a small viscosity, $\eta = 1$. The effecting regularization is investigated in (Hartmann and Bier, 2008) and it is demonstrated

[3]For the elasto plastic h-refinement with $\eta = 0$ each of the five base elements is subdivided in n×n elements, with $n = \{1, 3, 5, 7, 9, 11, 13, 15, 25, 27, 29, 31, 33, 35\}$

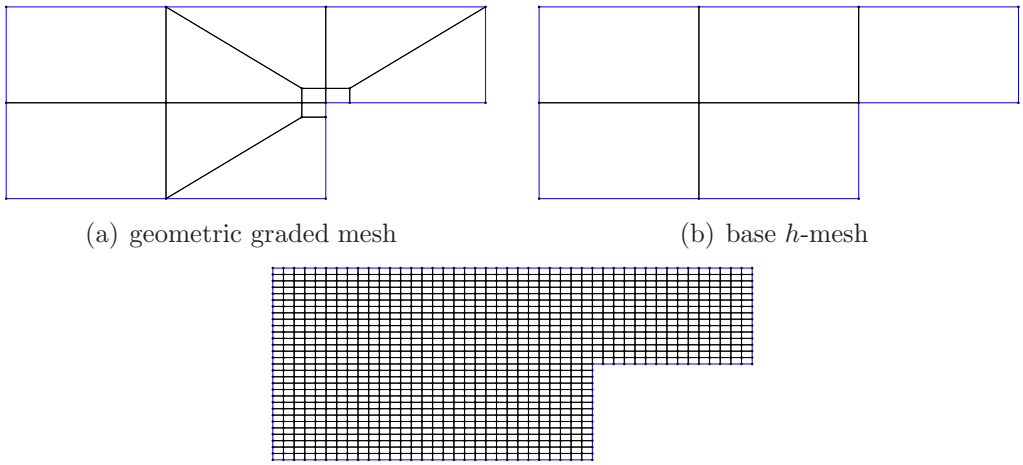
Figure 5.21: Spatial distribution of the relative density mapped on the reference configuration. The mesh is refined once towards the singularity with the geometric progression factor 0.15. The polynomial degree of the shape functions is $p = 9$ on all elements.

there that a small viscosity did not alter the resulting stresses significantly. To rule out any effects of rate-dependence and to compare only the different spatial discretization strategies the following calculations were performed with a *fixed* small step size of 1/150 of the total displacement and the order of Gauss integration set to 5 for the h-refinement[4] with $p = 2$ and $p = 3$. For p-refinement on the geometrically graded mesh, the Gauss order is set to $p + 3$ for even p, e.g. if $p = 4$ we use an integration order of 7, and $p + 4$ for odd polynomial degrees of the Ansatz functions.

We investigate the convergence of the axial Cauchy stress σ_{zz} at the material point $r = 5, z = 9$. The stresses are plotted vs. the logarithm of the degrees of freedom in Fig. 5.23 comparing the different discretization strategies. The same data is shown in Fig. 5.24, where the abscissa shows the CPU-time in logarithmic scale. If instead of the computation time the total number of unknowns, i.e. displacement degrees of freedom *plus* internal variables (seven at each Gauss point) is plotted, Fig. 5.25, it is apparent that an investment in the polynomial order pays off. Because for the h-refinement with $p = 2$ and $p = 3$ in both cases an integration order of 5 is used, those curves are almost parallel as the number of internal variables that dominate the total number of unknowns is the same, the minimal shift is due to the different number of displacement degrees of freedom. The step-like descent of the p-extension is caused by the construction of the integration orders to be $p + 4$ for odd polynomial orders and $p + 3$ for even p. Consequently, both $p = 5$ and $p = 6$ are integrated with order 9, for example.

From these plots it is apparent that for this nonlinear example the p-extension on the geometrically refined mesh needs significantly less degrees of freedom than the uniform h-

[4]For the viscoplastic h-refinement with $\eta = 1$ each base element was subdivided in n×n elements, for $p = 2$ and $p = 3$ the same subdivisions were used with $n = \{1, 3, 5, 7, 9, 11, 13, 15, 25, 29, 35, 41\}$



(a) geometric graded mesh (b) base h-mesh

(c) h-refined mesh, e.g. 15×15

Figure 5.22: The two classes of meshes employed. A geometrically graded mesh towards the singularity is used for p-extension while for h-extension with a fixed polynomial degree each element of the base h-mesh is uniformly subdivided in up to 41×41 elements.

refinement to achieve the same accuracy. It may be argued, whether it is fair to compare a p-version on a geometrically graded mesh with a h-version computation on a uniformly refined mesh. It is true, that a priori information where to refine is exploited for the geometric mesh. Therefore, we also compare in terms of computation time. The computational cost to set up the stiffness matrix of an element with a high polynomial degree is higher but due to the better rate of convergence of the p-extension this investment pays off also for finite strain applications with strongly nonlinear material like in this example.

The global equilibrium was checked by computing the out of balance force. To this end the reaction forces in axial direction along all boundary edges were added. This procedure is applied and described in detail in Sec. 4.9.1.5 for the same geometric setting however with a linear constitutive model. The absolute residuum is below 10^{-5} N and the relative value (residuum over reaction force along top edge) for the reported runs is well below 10^{-8}.

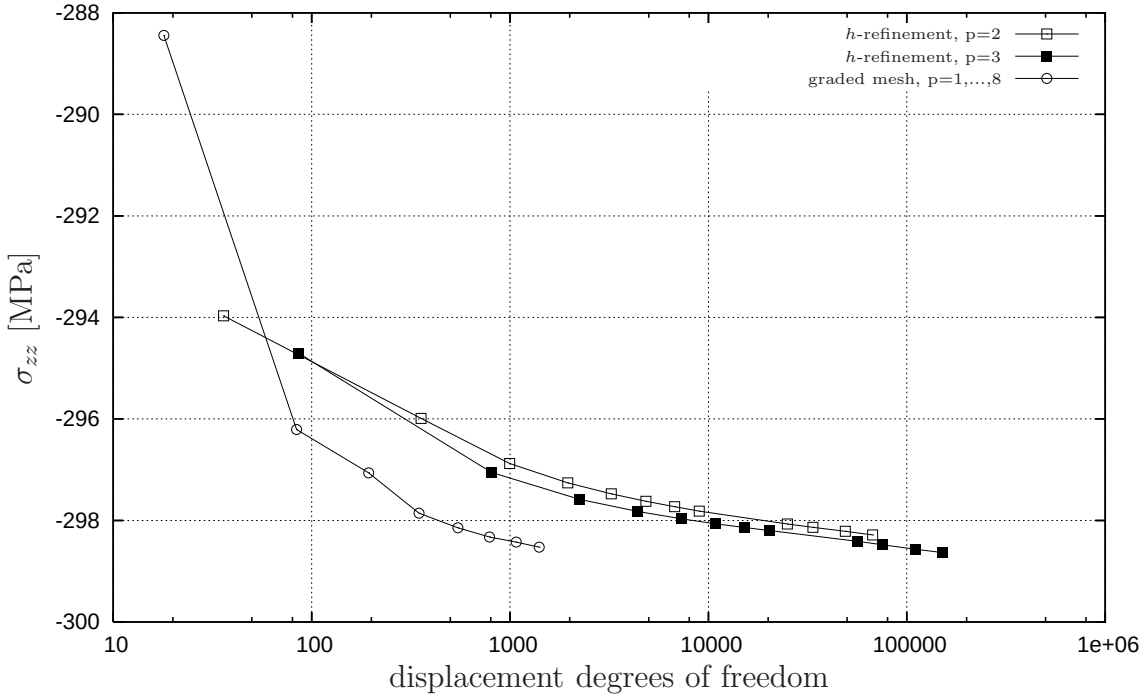


Figure 5.23: Convergence of Cauchy-stress σ_{zz} at point r=5, z=9 for the h- and p-refinement strategies in terms of the displacement degrees of freedom.

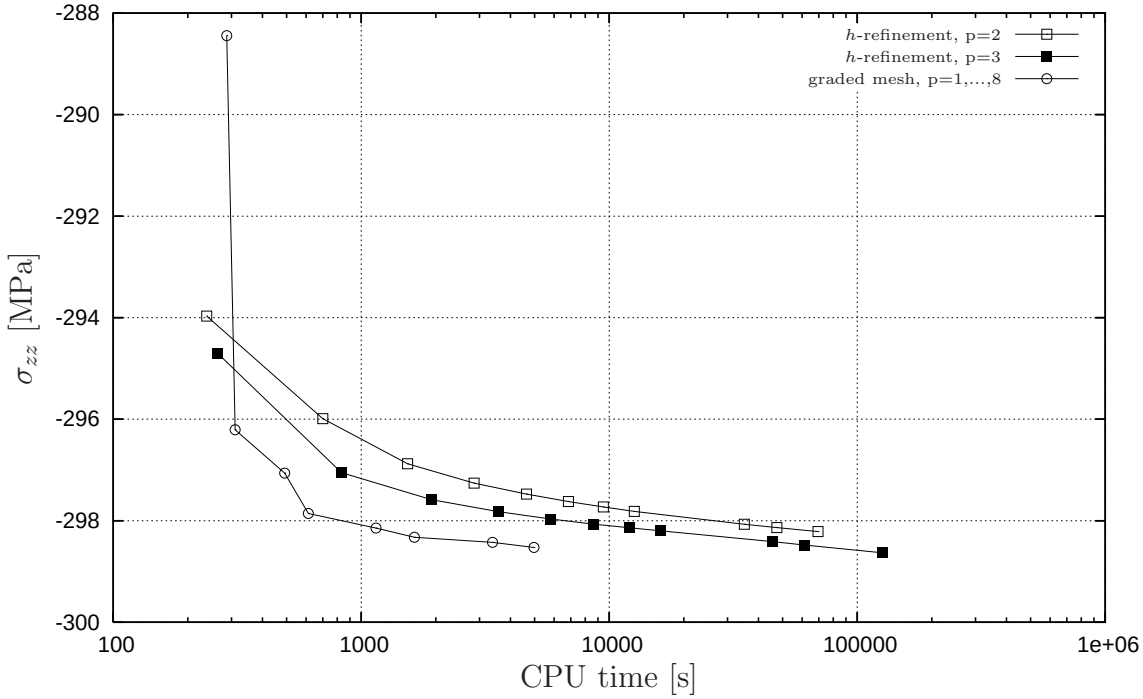


Figure 5.24: Convergence of Cauchy-stress σ_{zz} at point r=5, z=9 for the h- and p-refinement strategies vs. the CPU-time of the computation.

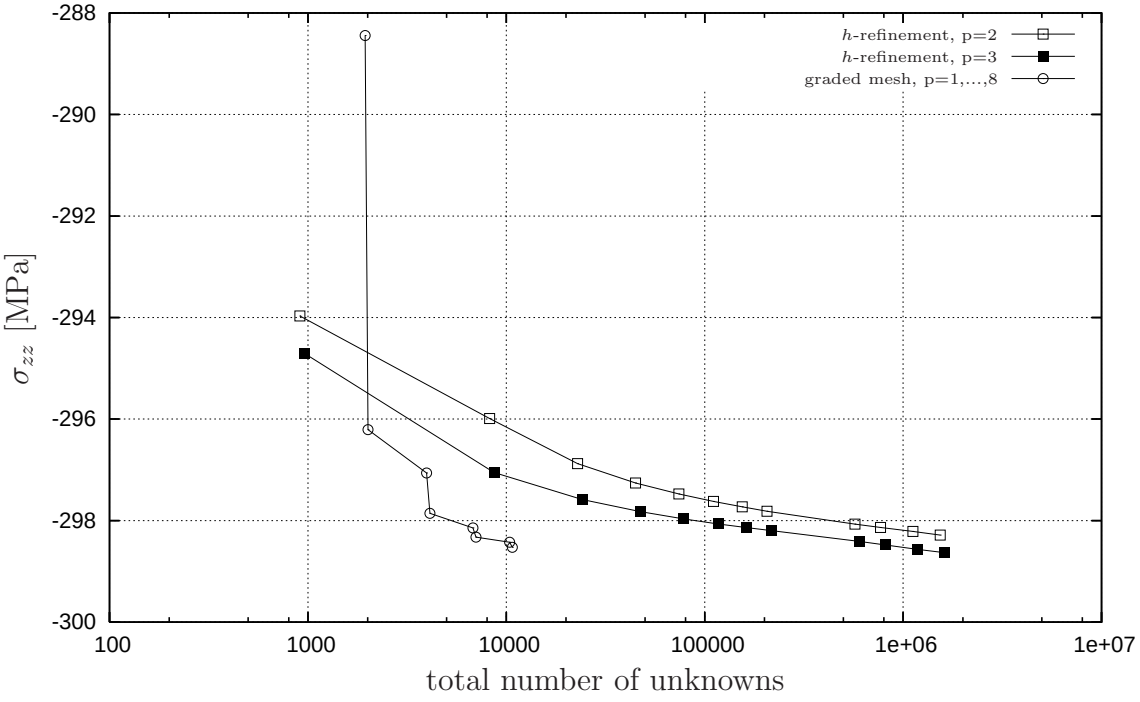


Figure 5.25: Convergence of Cauchy-stress σ_{zz} at point r=5, z=9 for the h- and p-refinement strategies vs. the total number of unknowns.

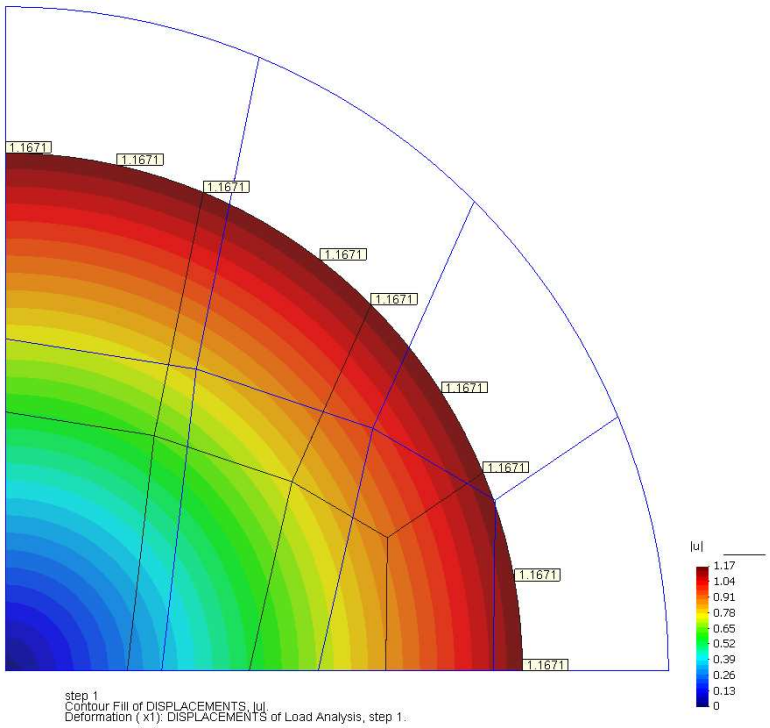
5.4.2 Cold isostatic pressing

To show the performance of an implicit p-version approach for a 'smooth' finite deformation problem the cold isostatic pressing of a ball made of copper powder is modeled. Exploiting the rotational symmetry only a quarter in the longitudinal plane of the sphere is meshed with axisymmetric elements. The pressure acts on the current surface, hence follower loading needs to be applied. The global implicit formulation using the MLNA, see Sec. 4.7.2, to solve the nonlinear equation system has the benefit that after an initial densification of the powder the stepsize can be increased resulting in short computation times compared to an explicit approach. Tab. 5.3 reports the computation time, number of displacement degrees of freedom and total number of unknowns for raising the polynomial degree p of the Ansatz functions on the fixed 7 element mesh. As for the elasto-plastic case ($\eta = 0$) 8 unknowns have to be determined at each quadrature point from a local nonlinear equation system, the number of displacement degrees of freedom must be augmented by all the internal variables of the complete structure to obtain the total number of unknowns for a meaningful criterion of the problem size as given in the last column of Tab. 5.3. It must be emphasized, that the stress algorithm is given in a three-dimensional formulation. An adaption to the axisymmetric case, where only 5 internal variables are required, has not been carried out. To obtain accurate results for the coarse mesh shown in Fig. 5.26 the circular arc is described analytically with the blending function method as introduced in Sec. 4.3. We show the deformation after springback when the pressure of 500 MPa is released.

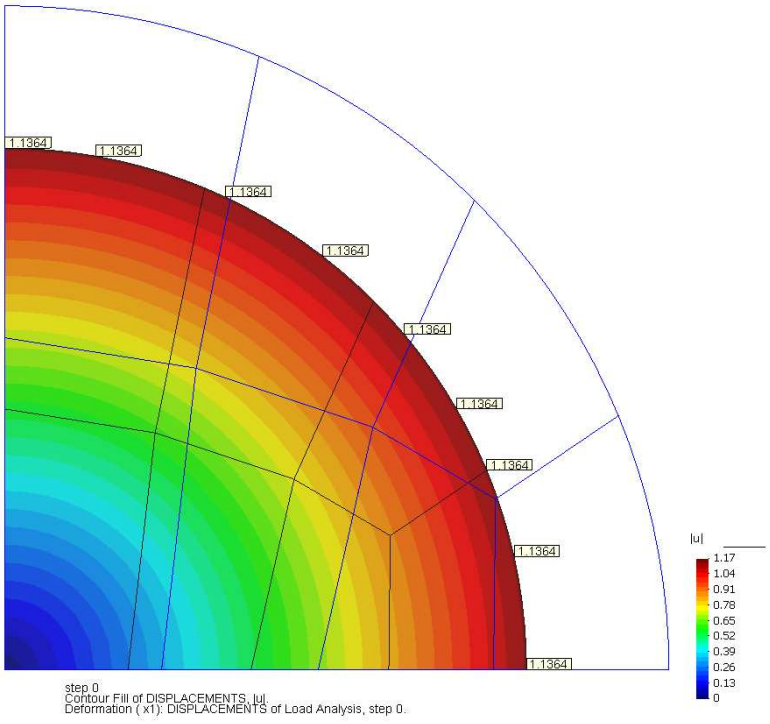
Table 5.3: Cold isostatic pressing of a sphere with 7 elements. The order of Gauss-Legendre integration is chosen $p + 1$ in each spatial direction

p	CPU time [s]	dof	no. of GPs	total no. of unknowns
3	2.39	50	112	946
4	3.18	128	175	1528
5	5.37	188	252	2204
6	8.87	262	343	3006
7	14.67	350	448	3934
8	24.13	452	567	4988
9	38.68	568	700	6168
10	59.38	698	847	7474

As the displacement $|\vec{u}|$ along the arc should be constant, the influence of integration order and polynomial degree on the displacement can be investigated in the following. Compare Sec. 4.6.4 for a general discussion of considerations concerning numerical quadrature. In Fig. 5.27 and Fig. 5.28 it can be seen, that the influence of the order of Gaussian integration is very small once a sufficient order greater or equal $p + 1$ is chosen. Relevant is the polynomial degree as can be seen in Fig. 5.29 and Fig. 5.30 for higher Ansatz orders and Gauss order of $p+1$. The four arcs that are visible in most plots stem from the fact that the circular boundary is composed of four elements. The displacement field on each element is approximated by Ansatz functions of a certain degree. As only C^0 continuity over the element boundaries is prescribed, we see the peaks where two elements meet. For the chosen mesh with 7 elements, Fig. 5.31 shows on a logarithmic scale the deviation $\big|\,|\vec{u}| - |\vec{u}_{\text{ref}}|\,\big|$ for $p = 3, 4, 5, 6$ along the circular arc after compression. The reference value $|\vec{u}_{\text{ref}}| = 1.16706829500301$ is the mean value of 800 sample points calculated from an overkill solution with 93 elements and $p = 10$. From Fig. 5.31 we see an increase in the polynomial order of the shape functions corresponds to an increase in the accuracy of the representation of the circular arc of at least one order.

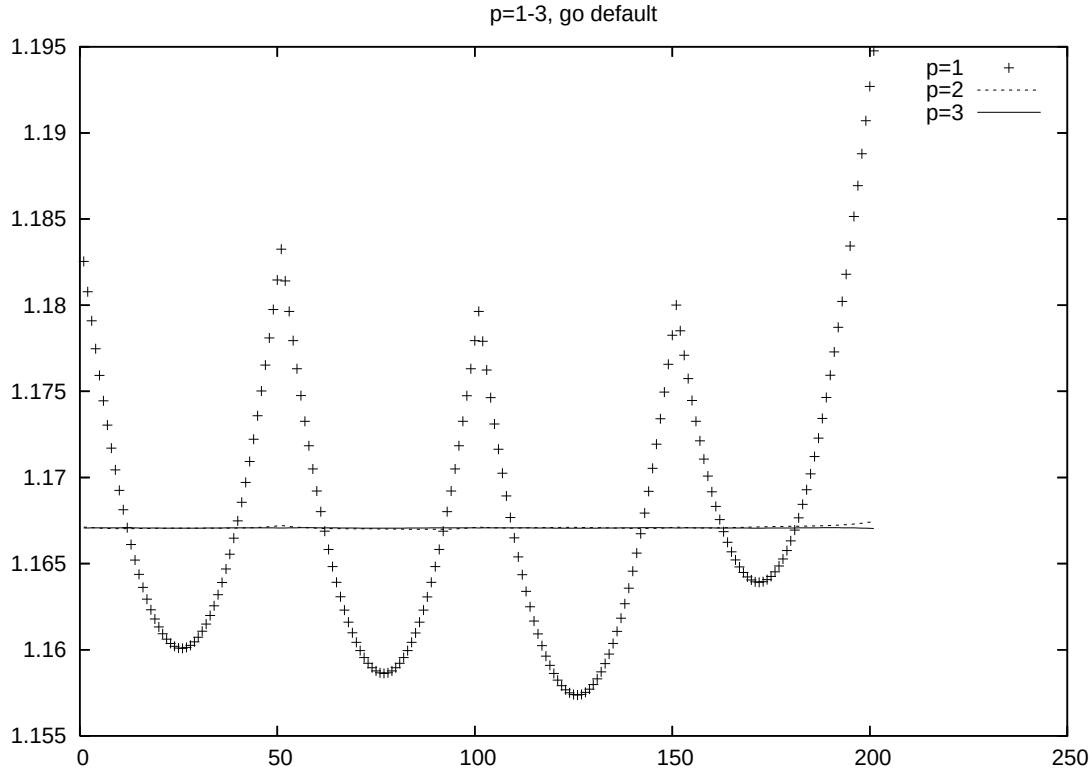


(a) Displacement $|\vec{u}| = 1.1671$ along the arc for an applied pressure of 500 MPa. The relative density is 0.88753 throughout the domain.

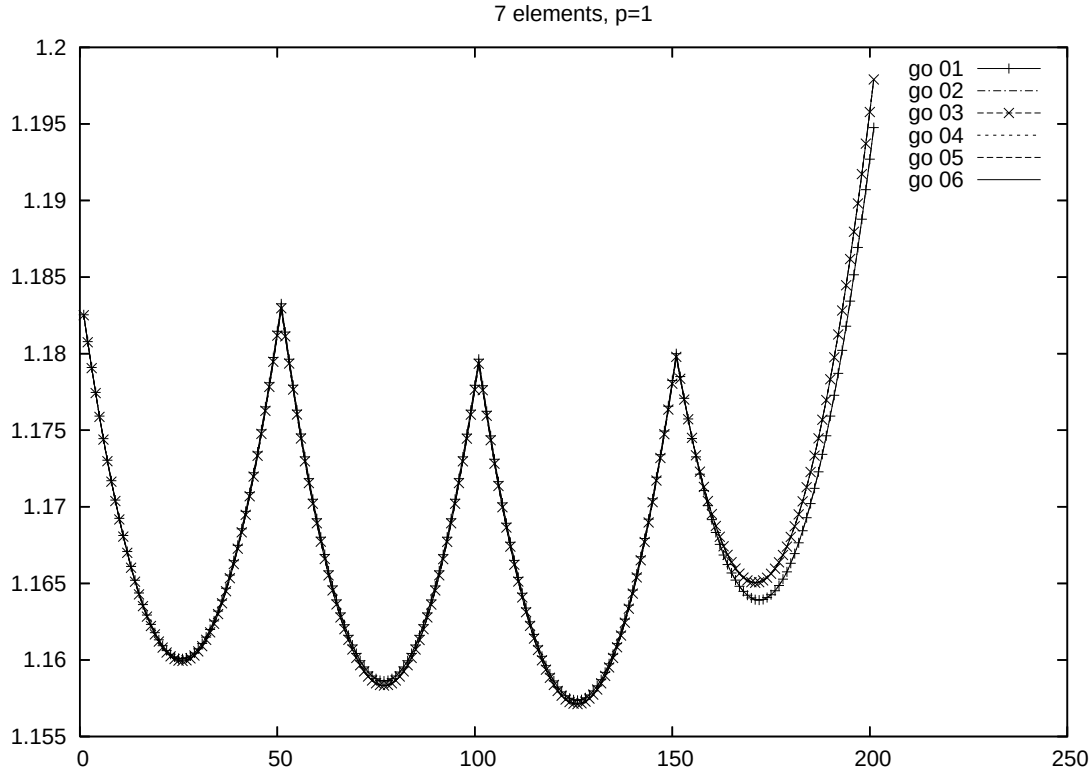


(b) Remaining displacement $|\vec{u}| = 1.1364$ along the arc after unloading (springback). The relative density after springback is 0.86631.

Figure 5.26: Cold isostatic pressing of a sphere, here 7 elements, $p = 3$. Note the very precise approximation of the circular arc on the coarse mesh due to the blending function method.

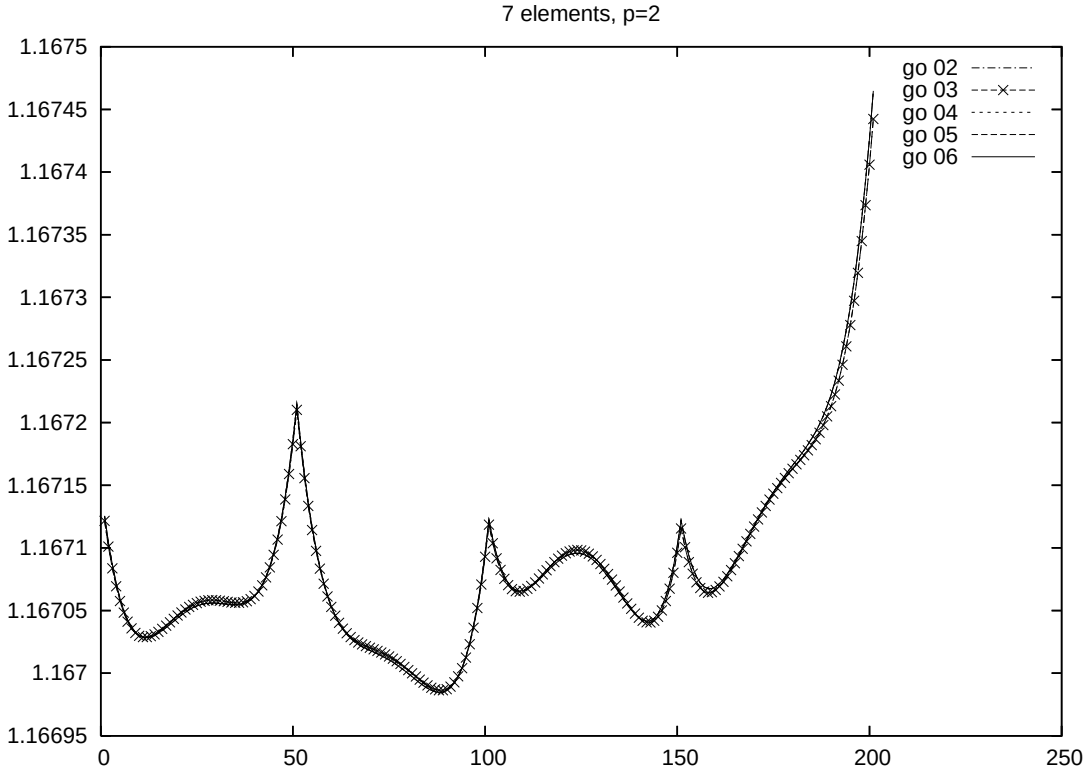


(a) Displacement $|\vec{u}|$ at 200 points along the curved boundary, uniform $p = 1, 2, 3$, Gauss order $p+1$

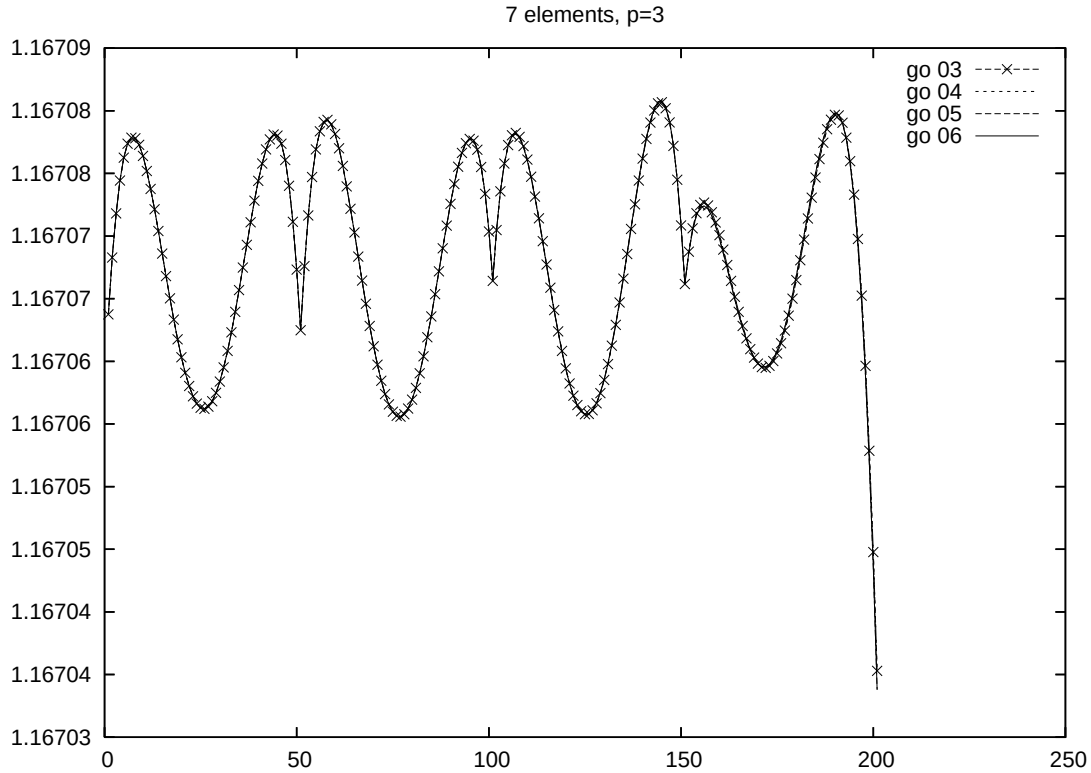


(b) Study of the influence of the Gauss order: uniform $p = 1$, increasing Gauss order

Figure 5.27: Cold isostatic pressing of a sphere, here 7 elements, $p = 1, 2, 3$. The displacement $|\vec{u}|$ at 201 points along the circular arc is shown. The plots reveal that the influence of the Gauss-order is small compared to the influence of the Ansatz-order p.

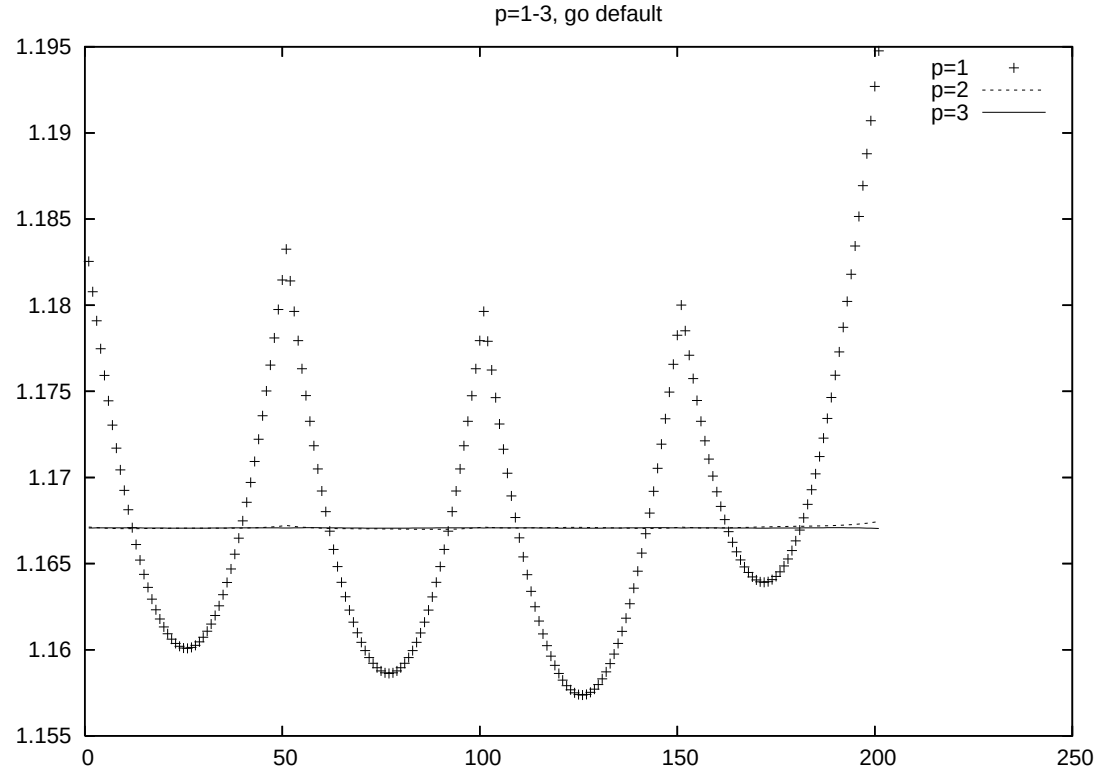


(a) Study of the influence of the Gauss order: uniform $p = 2$, increasing Gauss order

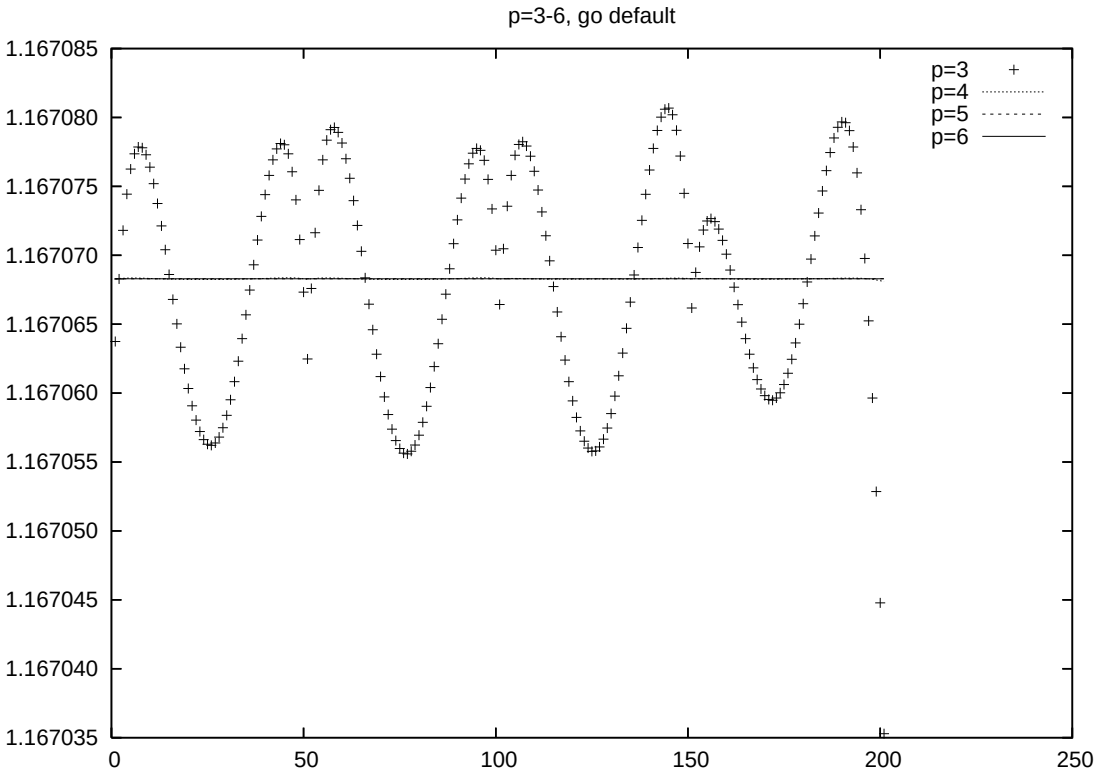


(b) Study of the influence of the Gauss order: uniform $p = 3$, increasing Gauss order

Figure 5.28: Cold isostatic pressing of a sphere, here 7 elements, $p = 2, 3$. The displacement $|\vec{u}|$ at 201 points along the circular arc is shown. The plots reveal that the influence of the Gauss-order is small compared to the influence of the Ansatz-order p.

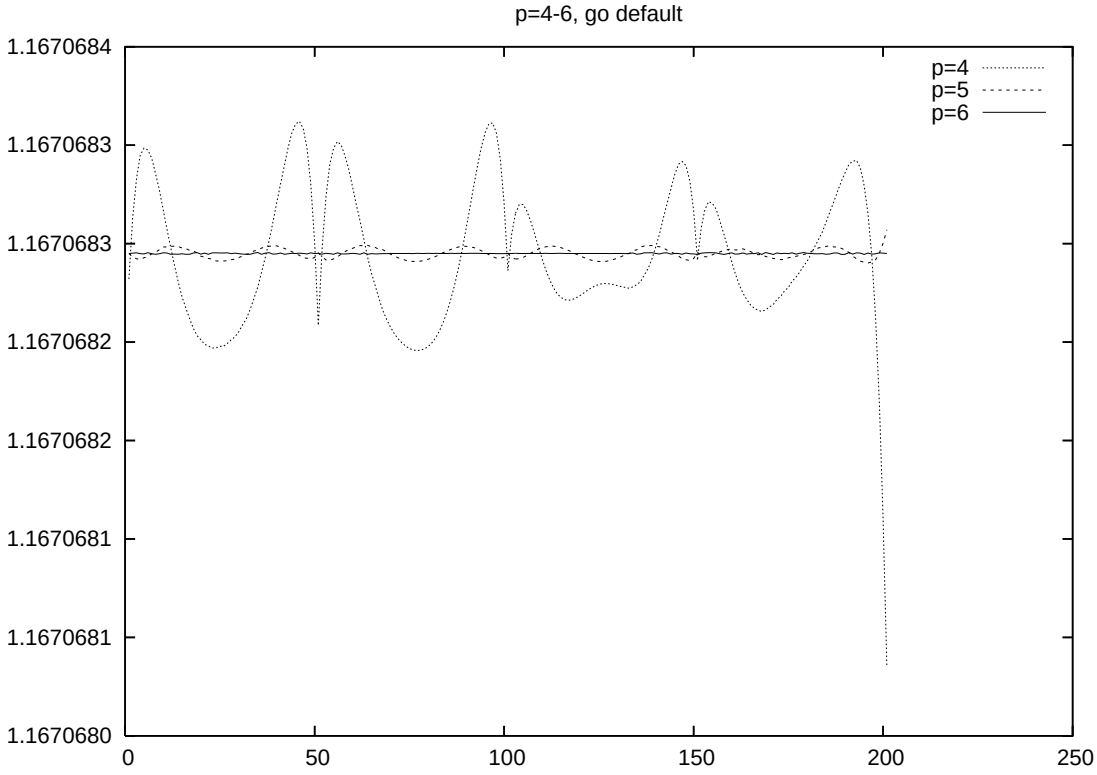


(a) uniform $p = 1, 2, 3$

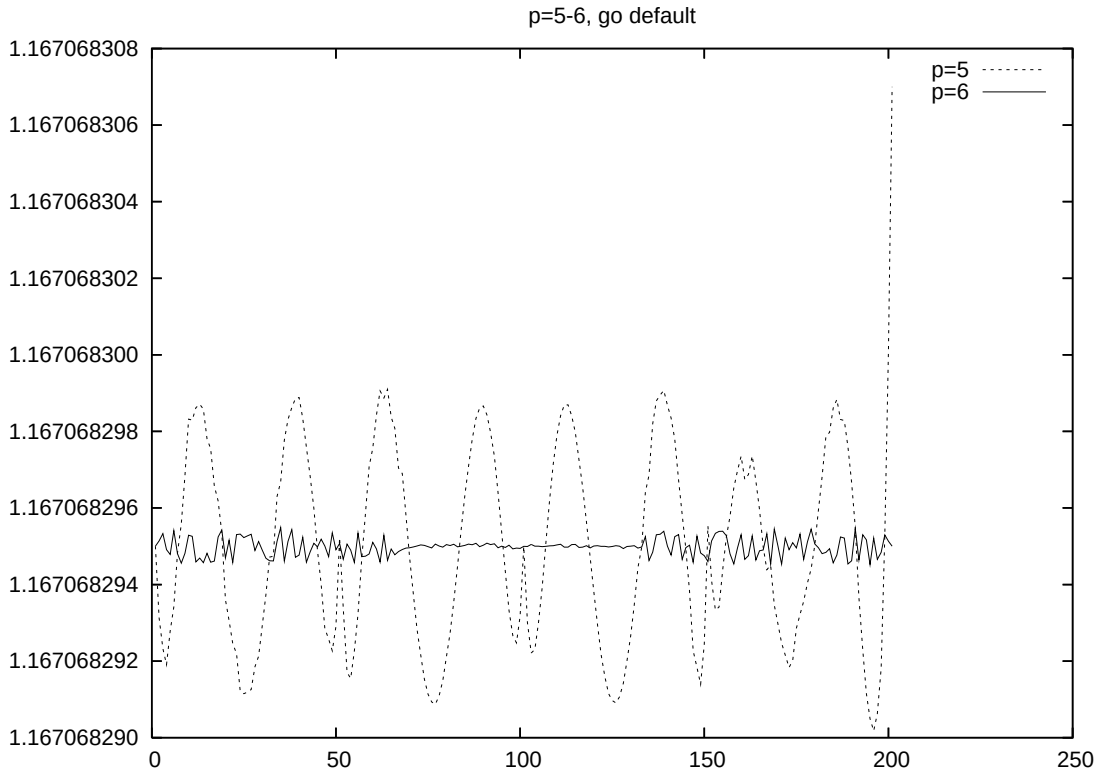


(b) uniform $p = 3, 4, 5, 6$

Figure 5.29: Cold isostatic pressing of a sphere, here 7 elements, $p = 1 - 6$. The displacement $|\vec{u}|$ at 201 points along the circular arc is shown. Gauss order $p + 1$



(a) uniform $p = 4, 5, 6$



(b) uniform $p = 5, 6$

Figure 5.30: Cold isostatic pressing of a sphere, here 7 elements, $p = 4 - 6$. The displacement $|\vec{u}|$ at 201 points along the circular arc is shown. Gauss order $p + 1$

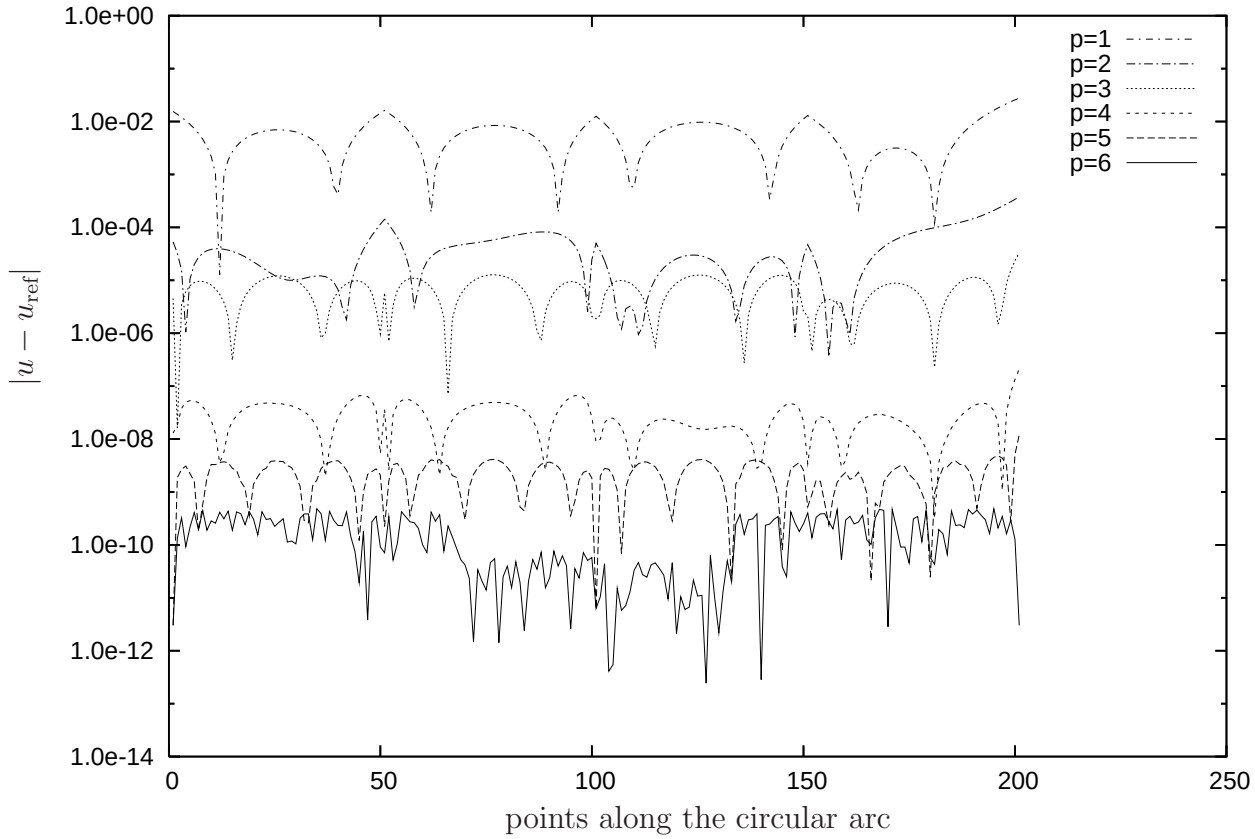


Figure 5.31: Influence of the polynomial degree on the deviation $|\vec{u} - \vec{u}_{\mathrm{ref}}|$ in the displacement along the circular arc after compaction with 500 MPa. Gauss order $p + 1$

5.4.3 Rubber isostatic pressing

Rubber isostatic pressing (RIP) is a process equivalent to cold isostatic pressing (CIP) that initially was developed for the densification of permanent magnet powders (Sagawa et al., 2000) that are sintered afterwards. The benefit of an isotropic densification for creating magnets is that an initial orientation of the powder given by a pulsed magnetic field is not disturbed during pressing. Pulsed fields can be much stronger than a stationary field. RIP is now also applied for other materials like titanium alloys, ceramics and ordinary metal powders and a wide variety of shapes can be produced, cf. Fig. 5.32. For a comparison of CIP and RIP consult (Yang et al., 2004) and the references therein.

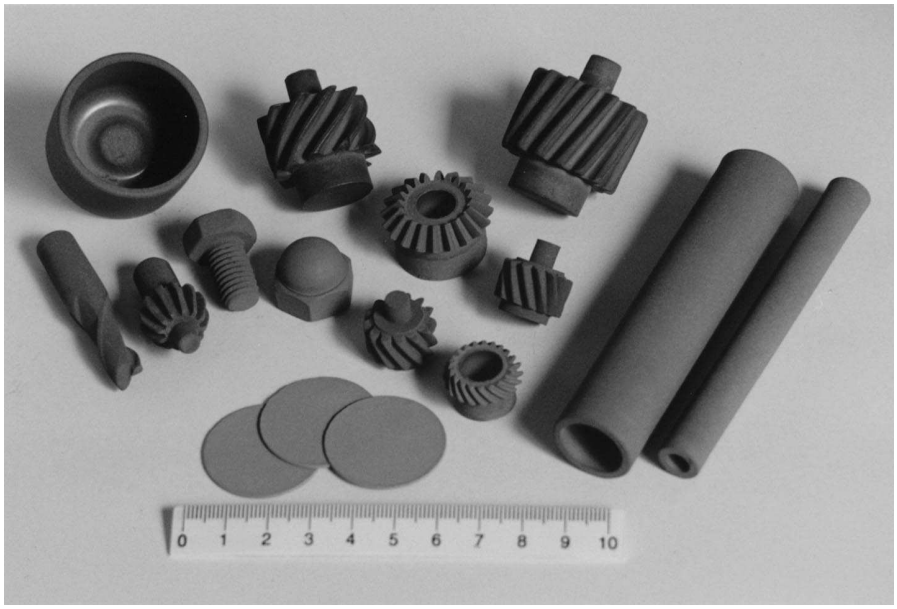


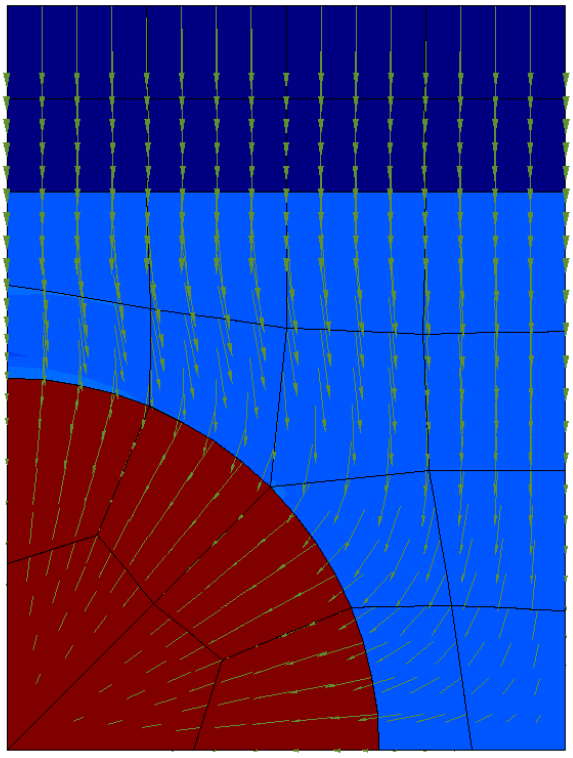
Figure 5.32: Green compacts of titanium powder produced by RIP, (Sagawa et al., 2000)

The powder is filled in a cavity of a massive rubber mold that in turn is pressed by a punch in a die-compaction machine. For industrial applications the most important know how concerns the automated filling of the powder with a high initial density. During the compaction the powder is formed in an almost isotropic deformation generated in the rubber mold itself. The principle is similar to CIP, however, as it does not use fluid pressure it is simpler and safer. The isotropy or anisotropy of the deformation of the powder is influenced by the size of the cavity and the thickness of the rubber mold in lateral direction. A greater wall thickness results in a larger lateral deformation of the powder with respect to the shrinkage along the press-axis (Sagawa et al., 2000). This can be studied in Fig. 5.35(a) where u_r is plotted.

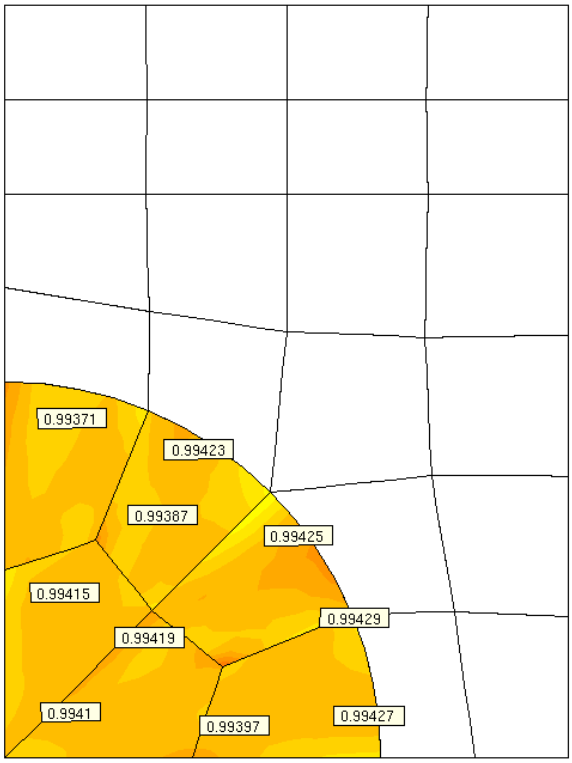
The examples in this section just apply the idea of RIP and give a first qualitative insight. For quantitative predictions a contact formulation allowing separation of the compacted powder and the surrounding rubber during spring back would be needed.

5.4.3.1 Spherical cavity

In this numerical example a steel die under displacement control is pressed on a rubber mold with a spherical cavity filled with powder. The axisymmetric domain is characterized by three materials, the steel die on the top and the hyperelastic material embedding the spherical cavity of powder. The hyperelastic mold is characterized by the polyconvex strain energy function proposed by Hartmann and Neff (Hartmann and Neff, 2003), see Eqs. (2.64) – (2.66), with the parameters $\alpha = 0.00367$ MPa, $c_{10} = 0.1788$ MPa, $c_{01} = 0.1958$ MPa and a compression modulus of $K = 5000$ MPa. The bottom and right boundaries are fixed in normal direction. The rubber transforms the vertical deformation of the die to an almost isostatic pressure on the embedded powder as can be seen by the displacement vectors in Fig. 5.33(a). Fig. 5.33(b) shows that the compacted powder has almost homogenous relative density.



(a) Deformation trajectories. The powder domain is displayed in red, the steel die is the dark blue regions on top and the rubber mold is indicated in light blue.



(b) The resulting relative density in the compacted powder is almost homogenous.

Figure 5.33: Rubber isostatic pressing of a sphere

5.4.3.2 Tube for vacuum circuit breakers

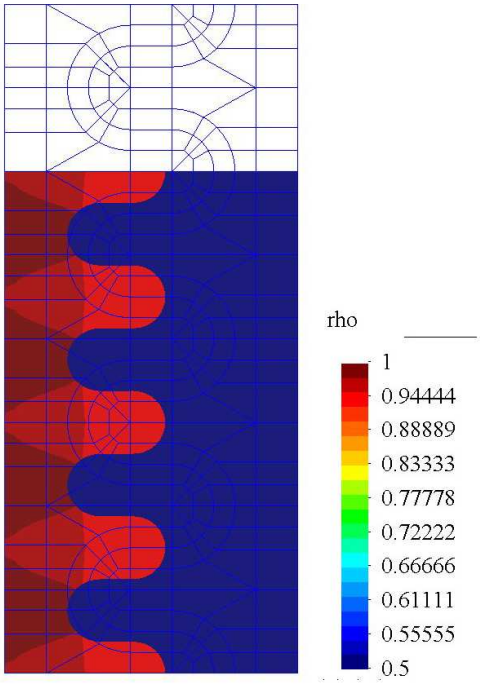
This example is inspired by a rubber mold and corresponding ceramic tube of a vacuum circuit breaker[5] CIP'ed from Al_2O_3, Fig. 5.34(a). Here we investigate a similar setting applying RIP and the constitutive model for copper powder.

[5]Ceramitec trade fair 2006 in Munich

In the initial configuration the domain is 70 mm in radial and 160 mm in axial direction. The dimensions of the powder region in radial direction are maximum 50 mm and minimum 20 mm. The circular arcs have a radius of 10 mm. Along the top boundary a normal displacement in negative axial direction of 40 mm is prescribed.

A hyperelastic mold characterized by a polyconvex strain energy function, cf. Sec. 5.4.3.1 for the parameters, defines the cavity for the powder. The circular arcs are exactly described using the blending function concept, cf. Sec. 4.3. In the axisymmetric analysis all boundaries are fixed in normal direction, only along the top boundary a displacement controlled deformation in axial direction is prescribed. Perfect bonding is assumed at the interface between powder and mold. For the analysis a polynomial degree $p = 8$ is set in all elements. The number of displacement degrees of freedom is 18 238, the CPU time was 5489.64 seconds, i.e. about 1.5 hours on an Opteron processor with 2.4 GHz. The undeformed mesh and the deformed configuration with corresponding relative density can be seen in Fig. 5.34(b).

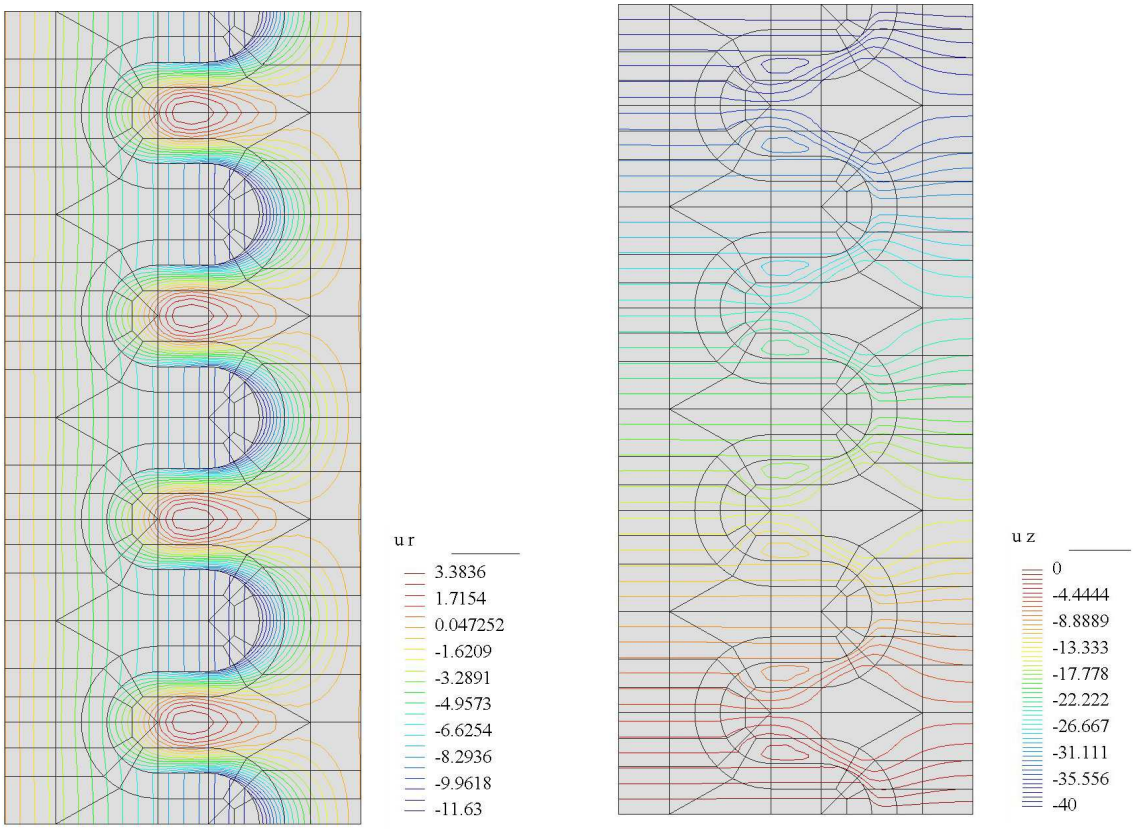
(a) Specimen made of Al_2O_3 together with its rubber mold



(b) Initial and deformed configuration. The resulting relative density in the compacted powder is almost homogenous. The blue domain designates the elastomer where no density is plotted.

Figure 5.34: Rubber isostatic pressing of an insulator

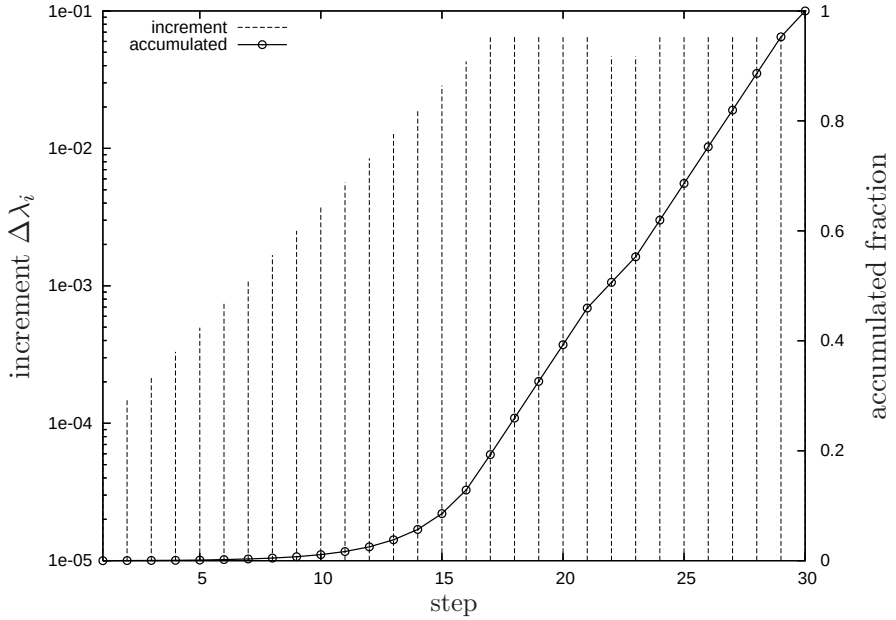
Assessing the isotropy of the deformation we see in Fig. 5.35 almost perfect horizontal iso-lines of the vertical deformation u_z in the powder domain. The effect mentioned on p. 146 that a wider layer of rubber mold in lateral direction induces a larger lateral compaction of the powder can be studied from the iso-lines of u_r in those parts where the rubber cone advances into the powder region. Apart from these regions the vertical u_r lines are almost parallel indicating an almost isotropic compaction.



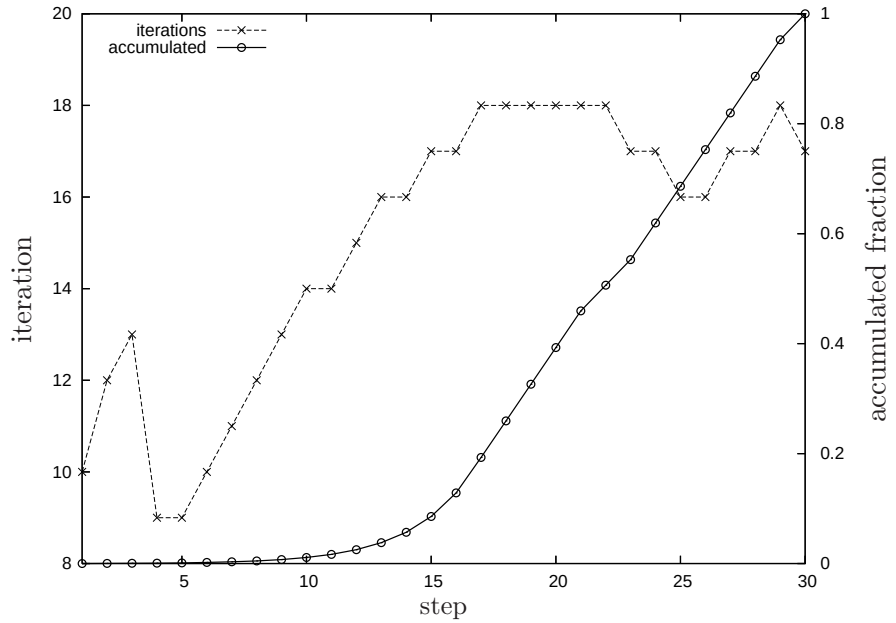
(a) Radial deformation u_r (b) Axial deformation u_z

Figure 5.35: Rubber isostatic pressing of an insulator

Fig. 5.36 shows the displacement increment (normalized to the total applied displacement) and accumulated displacement vs. the steps. It is clearly visible that in the beginning small steps are necessary and later as the material consolidates the stepsize can be increased. The individual steps needed under 20 global iterations for convergence.



(a) Increment and accumulated displacement



(b) Global iterations and accumulated displacement

Figure 5.36: Rubber isostatic pressing of an insulator. Adaptive step control of the increment $\Delta\lambda_i \overline{u}_{\text{top}}$ and the so far accumulated fraction $\sum_{i=1}^{\text{step}} \Delta\lambda_i$.

5.4.4 Experimental validation of a complex process

5.4.4.1 CIP of a powder cylinder with a rigid spherical inset

This example models a 'complex' experiment where uniaxial die compaction is followed by cold isostatic pressing (CIP). The experimental setup and simulation with an explicit h-version code is described in detail in (Szanto et al., 2007). Here the calculations are done applying the p-version of the FEM. A steel sphere is embedded in a cylinder of copper powder that is compressed. This inset causes an inhomogeneous deformation in the body. The final body is shown in Fig. 5.37. Fixing the position of the sphere embedded in the powder inside the CIP chamber is very difficult. Therefore, as a first step uniaxial pre-compaction of the specimen with the sphere inside is done. This brittle body is carefully transferred to the CIP unit where the isostatic pressure is applied.

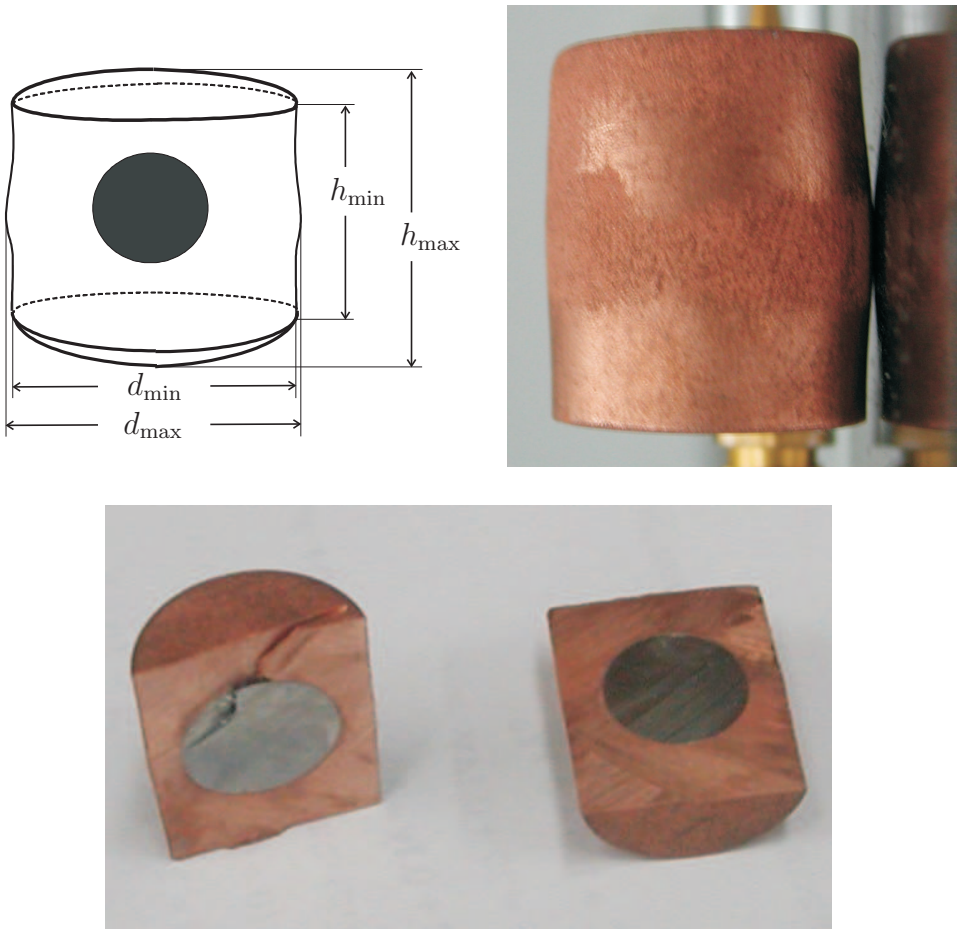


Figure 5.37: Cut specimen after the experiment. The minimum and maximum diameter and height are available for comparison with the simulation.

5.4.4.2 Modeling the process

The procedure of the experiment suggests to split the modeling of the process into four stages (Fig. 5.38).

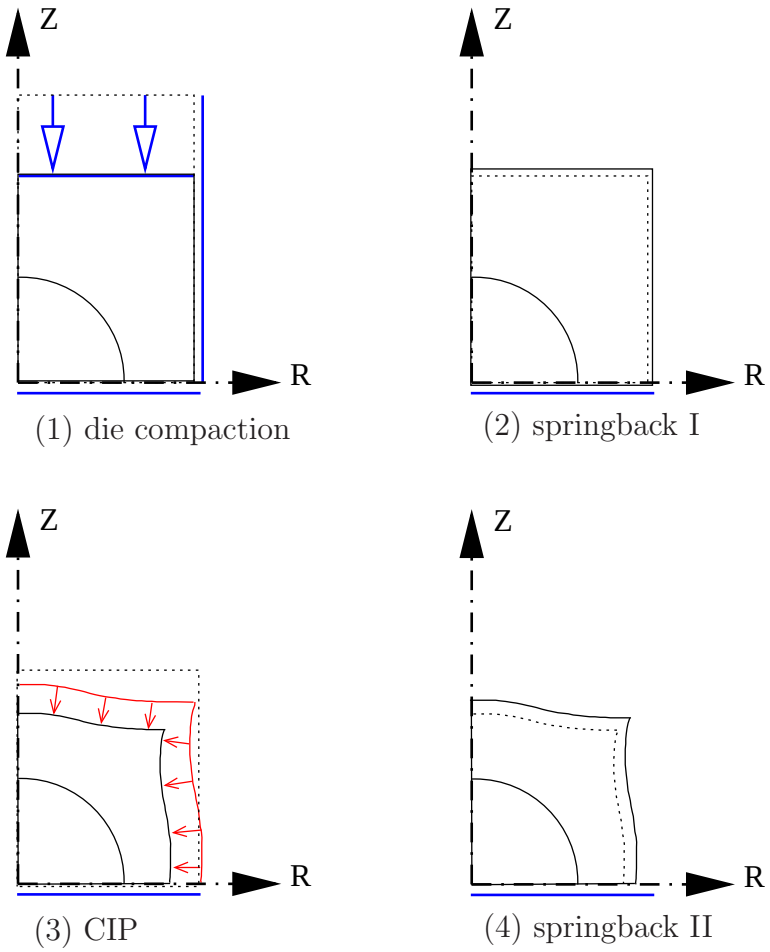


Figure 5.38: Schematic representation of the four stages of the process. Displacement perpendicular to the blue lines is clamped or prescribed as in (1). Dotted lines represent the shape at the start of the stage, solid lines at the end.

1. Die compaction with displacement control. The experimental data is compiled in the paper of (Szanto et al., 2007). The initial height reported there corresponds to a relative density of 0.376 for the copper domain. As the material parameters for the constitutive model were determined for powder with initial relative density of 0.42, the initial height corresponding to this relative density was calculated by purely geometrical considerations from the powder mass, the known volume of the steel sphere, density of copper and the dimensions of the cylindrical die. For the numerical simulation we use the resulting initial height of 29.55 mm.

   After die compaction the body has an average height of 23.482 mm. A corresponding deformation of 6.068 mm is prescribed to model this stage. The rotational symmetry allows to use axisymmetric elements and as the origin is approximately placed in the center of the sphere only a quarter of the longitudinal plane has to be modeled setting symmetry boundary conditions as shown in Fig. 5.39. Hence the initial height of the computational domain is 14.775 mm and the radius of the body is 10.015 mm. A deformation $\overline{u}_z = -3.034$ mm is prescribed at the upper edge. The left and right edges are fixed in radial direction ($\overline{u}_r = 0$), while the bottom edge is clamped in axial direction

($\overline{u}_z = 0$). Calculations showed that the rigid steel sphere is numerically equivalent to a clamped circular arc because it is much stiffer than the copper powder. Consequently, the calculations reported here were carried out with the arc clamped, i.e. along the arc $\overline{u}_r = 0, \overline{u}_z = 0$.

2. Springback I: The final state of the die compaction is the starting configuration for the springback stage where only symmetry boundary conditions are set and the reaction forces induced by the prescribed displacements of stage 1 are reduced to zero.

3. CIP: The body from stage 2 is subjected to isostatic pressure of 300 MPa acting on the current deformed configuration (follower loading) of the outer boundary.

4. Springback II: Also after CIP there is springback where the load is taken away. What remains is the overall "plastic" (remaining) deformation.

For the simulation a coarse mesh, as shown in Fig. 5.40, is used. Again the curved boundary is described by the blending function method. Counting from the arc in the lower left corner we distinguish three layers of elements where different polynomial degrees may be set as shown in Fig. 5.40.

The numerical simulation follows the four stages explained above. The state at the end of each run (displacement vectors of all elements, internal variables at all Gauss points) is the starting configuration for the following stage with the new boundary conditions. The rate-independent version of the constitutive model ($\eta = 0$) was used for comparison with the calculations of (Szanto et al., 2007). Different polynomial degree distributions were investigated. Comparing the numerically obtained dimensions to the experimental values we find that the difference is below 3.5 % and the moderate computation time as indicated in Tab. 5.5 allows to apply the simulation to gain insight into the process. Taking into account that no friction was modeled this is a very good agreement. It is interesting to note that in (Szanto et al., 2007) the same constitutive model is used within an explicit h-version code. The dimensions after the process match the experimental results very well, however, the calculation times were in the magnitude of days.

In our implicit approach it proved vital to implement an automatic control of the stepsize. Since the load-deflection slope at the beginning of the loading process is very flat, small loading steps have to be used. On the other hand, as the material gets denser the stepsize can be increased. The deformed body and the distribution of the relative density at the end of the process is shown in Fig. 5.41. The relative density along a cutline from the center of the sphere to the top right corner (in reference configuration) in Fig. 5.42 allows to rate the quality of the approximation. As the relative density depends on the determinant of the deformation gradient, i.e. on derivatives of the displacement, there may be discontinuities over element boundaries as the displacement field is only C^0 continuous there. These discontinuities are an indicator for the accuracy – analogous to arguments used in error estimators based on recovery methods, see for example (Rank and Zienkiewicz, 1987). For the uniform polynomial degree pattern $p = 4 - 4 - 4$, i.e. $p = 4$ in each of the three layers of elements shown in Fig. 5.40, a discontinuity is visible that is smoothed if the polynomial degree for the second and third layer of elements is raised using $p = 4 - 6 - 8$ indicating an improvement. If the polynomial order is raised to $p = 4 - 8 - 10$ an almost smooth curve is obtained.

Table 5.4: Experimental data for the embedded sphere example (sphere diameter 11.89 mm and mass 6.8749 g)

Experiment	total mass (g)	initial dimensions		after die compaction		after CIP			
	only powder	D	H	D (mm)	H (mm)	H_{min}	H_{max}	D_{min}	D_{max}
1	38.48	20.03	32.66	20.1	23.44	21.18	21.28	17.35	17.58
	31.6051								
2	38.4692	20.03	33.21	20.1	23.47	21.13	21.31	17.38	17.6
	31.5943								
3	38.442	20.03	32.53	20.1	23.5	21.05	21.32	17.33	17.64
	31.5671								
4	38.44	20.03	32.58	20.1	23.5	21.07	21.31	17.33	17.62
	31.5651								
5	38.447	20.03	32.5	20.1	23.5	21.03	21.3	17.35	17.63
	31.5721								
mean value	31.58074	20.030	32.696	20.100	23.482	21.092	21.304	17.348	17.614
std. dev.	0.01791	0.000	0.294	0.000	0.027	0.062	0.015	0.020	0.024

Table 5.5: Specimen dimensions (mm) after the entire process. Comparison of experiment (Szanto et al., 2007) and FE results. The CPU time includes all four stages of the process. The number in brackets is the relative difference between the experiment and the FE result in %.

Dimensions after release of pressure				H_{min}	H_{max}	D_{min}	D_{max}
Experiment (mean value)				21.092	21.304	17.348	17.614
standard deviation				0.062	0.015	0.020	0.024
numerical results							
run	dof	int.vars.	$\sum$ CPU [s]				
p=2	29	882	37.88	20.4437 (3.07%)	20.9088 (1.86%)	17.9495 (3.47%)	18.0834 (2.66%)
p=3	143	1568	78.32	20.4379 (3.10%)	20.8741 (2.02%)	17.9430 (3.43%)	18.0965 (2.74%)
p=4	228	2450	150.15	20.4369 (3.11%)	20.8811 (1.99%)	17.9398 (3.41%)	18.1071 (2.80%)
p=5	341	3528	278.69	20.4386 (3.10%)	20.8852 (1.97%)	17.9375 (3.40%)	18.1132 (2.83%)
p=2-4-6	330	3010	295.31	20.4394 (3.09%)	20.8929 (1.93%)	17.9456 (3.44%)	18.0862 (2.68%)
p=2-6-8	576	5026	837.61	20.4399 (3.09%)	20.8912 (1.94%)	17.9453 (3.44%)	18.0864 (2.68%)
p=3-6-8	584	5222	878.79	20.4426 (3.08%)	20.8876 (1.95%)	17.9418 (3.42%)	18.0968 (2.74%)
p=4-6-8	600	5474	944.73	20.4391 (3.10%)	20.8867 (1.96%)	17.9392 (3.41%)	18.1057 (2.79%)
p=4-8-10	926	8050	2514.59	20.4398 (3.09%)	20.8877 (1.95%)	17.9388 (3.41%)	18.1054 (2.79%)

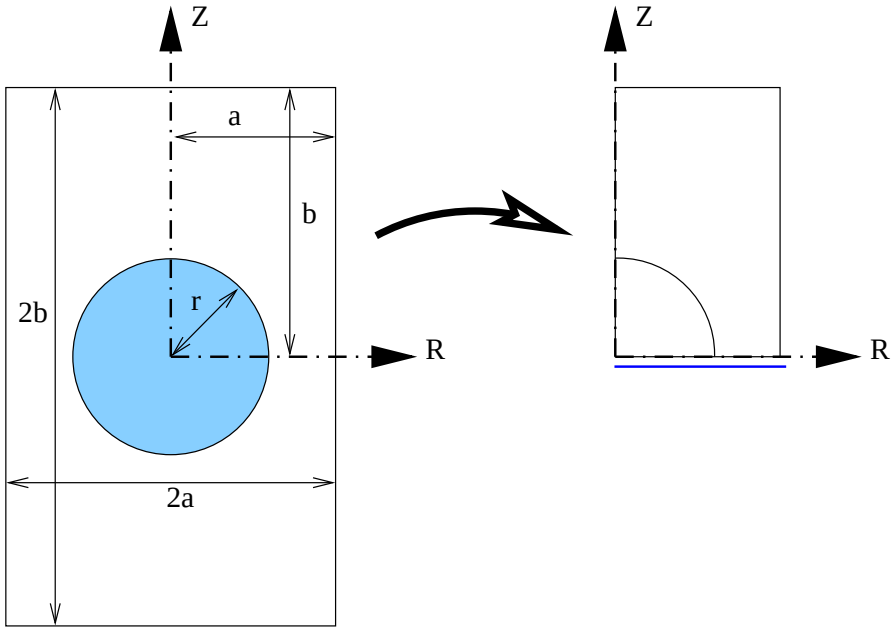


Figure 5.39: Cross section of the initial geometry in cylindrical coordinates and the reduced axisymmetric system with symmetry boundary conditions. The geometric dimensions are $a = 10.015$ mm ($2a = 20.03$ mm), $b = 14.775$ mm ($2b = 29.55$ mm) and the steel sphere has a diameter of 11.89 mm (radius $r = 5.945$ mm). The z-axis coincides with the axis of rotation.

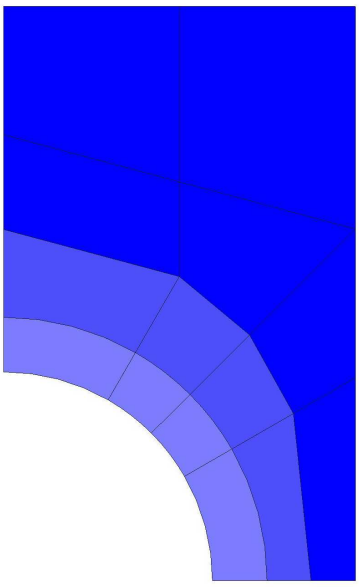
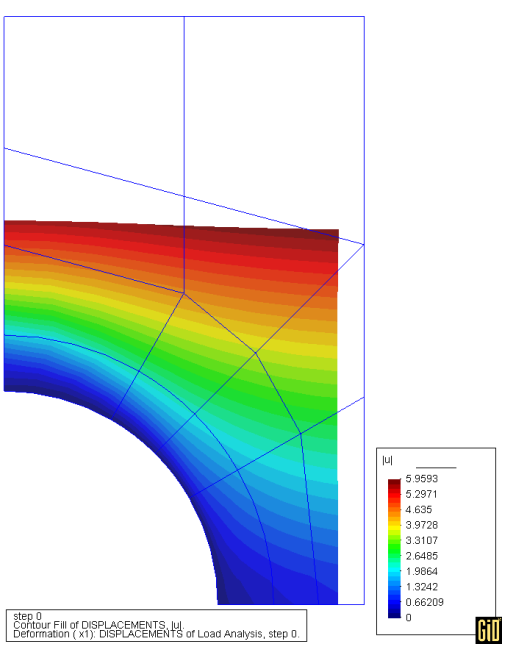
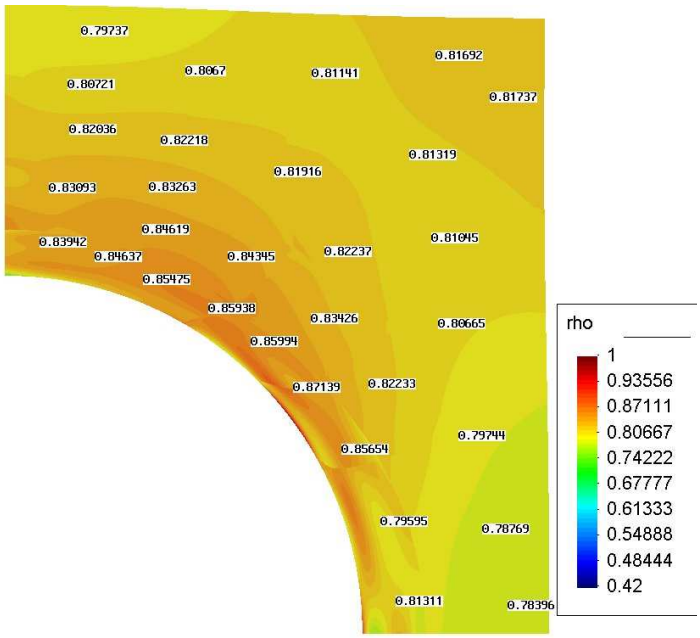


Figure 5.40: Coarse p-version mesh. The colors group the three areas where different Ansatz orders may be set. For example p=4-8-10 codes for $p = 4$ in the region adjacent to the arc, $p = 8$ in the next layer and $p = 10$ in the remaining elements.



(a) Undeformed mesh and deformation $|\vec{u}|$



(b) Spatial distribution of the relative density in the powder ($p = 5$ in all elements)

Figure 5.41: Displacement and distribution of relative density after the entire process

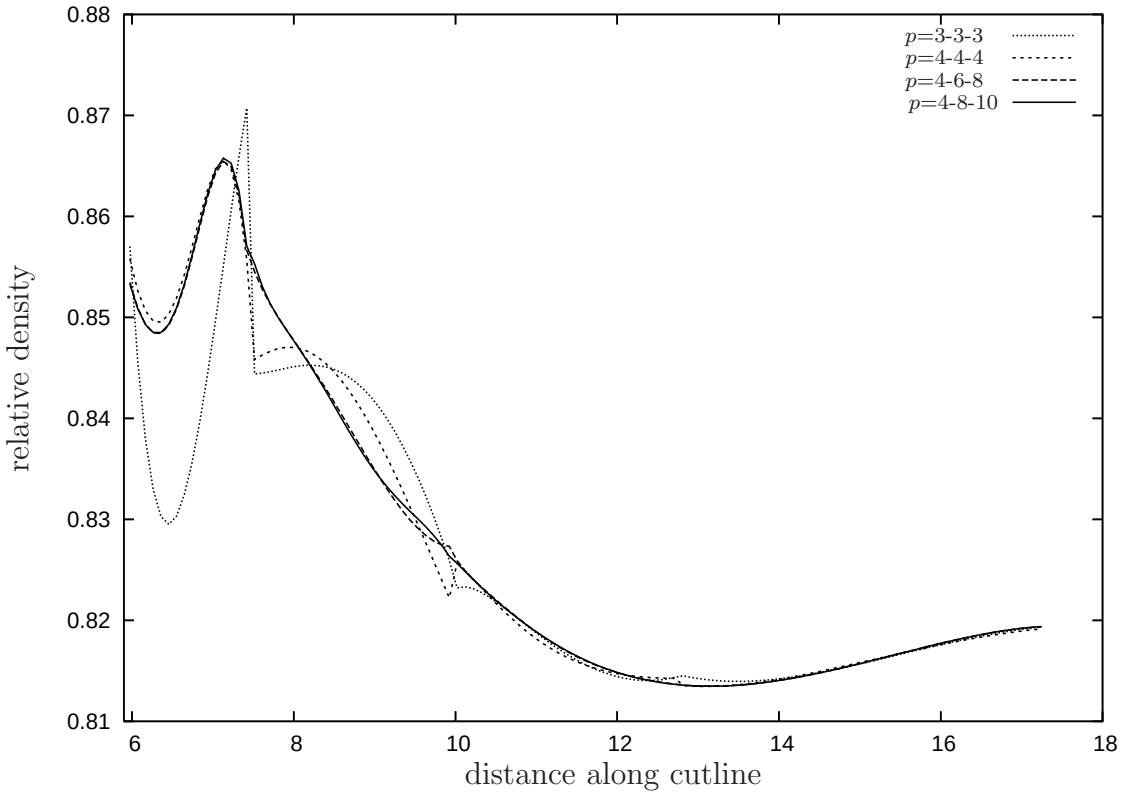


Figure 5.42: Convergence of the relative density along a cutline from the center of the arc to the top right corner (in reference configuration) for different polynomial patterns

Chapter 6

Conclusion

A constitutive model for metal powder compaction using a unique and convex single surface yield function is successfully incorporated in the implicit hierarchic high-order finite element code AdhoC . This setup is used to model large strain material nonlinear problems occurring in powder compaction problems. The Multi-Level Newton Algorithm (MLNA) is applied with a local stress algorithm (to update the internal variables) that provides the consistent tangent operator for the global Newton iteration. For material models with internal variables a significant computational effort must be spent to solve the local nonlinear equations describing the evolution of the internal variables. Therefore, besides the number of displacement degrees of freedom also the number of internal variables, for example 7 at each Gauss point, must be taken into account when judging the complexity of a simulation.

For the cold isostatic pressing process a deformation dependent load formulation ('follower loading') is derived and implemented in the p-FEM context. The implementation is verified against (semi)analytical hyperelastic solutions. These solutions also provide the possibility to investigate the robustness of the p-version with respect to finite strain volumetric locking for the first time. In this study additionally the effect of super-parametric mapping is visible that is examined also separately for large rigid body rotations.

Besides these points, reaction forces for hierarchic high order elements where the Ansatz functions do not form a partition of unity are introduced and a heuristic adaptive step-control mechanism is applied for the simulation of metal powder compaction. Starting with uniaxial die compaction experiments analogous to those used to calibrate the constitutive model, the restart procedure is verified. Next, cold isostatic pressing of metal powder is investigated numerically and a related process, rubber isostatic pressing is examined to explain how a die compaction of a thick rubber mold with a cavity for the powder results in an almost isostatic pressure like CIP. Finally, a 'complex' validation experiment where uniaxial die compaction is followed by cold isostatic pressing demonstrates that implicit high order elements are promising candidates for efficient and validated solutions for nonlinear problems.

For possible further investigations one could think of two directions. The first is to apply even more advanced finite strain constitutive models in the p-FEM context. The second line is to improve the performance of the numerical method.

- A natural line for future research is to move from cold isostatic pressing to hot isostatic pressing (HIP). Then, the constitutive model has to take the temperature into account, too. Also one could think of an additional electric field controlling the temperature in

the sample.

- Hierarchy is a powerful concept, not only for data organization, but also for adaptive control.
  - Hierarchical integration schemes can be used to control the error of numerical integration of the element matrices and vectors.
  - For the time integration hierarchical schemes like the Runge-Kutta method can be used to derive error indicators adaptively controlling the step size.
  - Analogous principles may be exploited for the spatial discretization were an adaptive *hp* method (modifying the mesh and polynomial degrees of the elements) or *pq* method (adapting the polynomial degrees in different spatial directions for a given mesh) is driven by error indicators computed from models of different hierarchy-levels.
  - The natural support of the hierarchic *p*-version for validation and verification can be combined with stochastic considerations to assess the level of confidence in the results based on uncertain input data.

In summery, hierarchic high order finite element methods together with hierarchic concepts provide powerful tools to address many nonlinear problems providing inherent verification properties.

Appendix A

List of publications originating from the GIF collaboration

1. Bier, W., and Hartmann, S. (2005):
   A finite strain viscoplasticity model for cold compaction processes of metal powder.
   In *Proceedings in Applied Mathematics and Mechanics*, volume 5, pages 263–264.

2. Heisserer, U., Düster, A., and Rank, E. (2005):
   Follower loads for axisymmetric high order finite elements.
   In *Proceedings in Applied Mathematics and Mechanics*, volume 5, pages 405–406.

3. Bier, W., and Hartmann, S. (2006):
   A finite strain constitutive model for metal powder compaction using a unique and convex single surface yield function.
   European Journal of Mechanics Series A/Solids, 25:1009 – 1030.

4. Yosibash, Z., Hartmann, S., Heisserer, U., Düster, A., Rank, E., and Szanto, M. (2007):
   Axisymmetric pressure boundary loading for finite deformation analysis using p-FEM.
   Computer Methods in Applied Mechanics and Engineering, 196:1261–1277.

5. Bier, W., Dariel, M., Frage, N., Hartmann, S., and Michailov, O. (2007):
   Die compaction of copper powder designed for material parameter identification.
   International Journal of Mechnical Sciences, 49:766–777.

6. Heisserer, U., Hartmann, S., Düster, A., and Yosibash, Z. (2007):
   On volumetric locking-free behavior of p-version finite elements under finite deformations.
   Communications in Numerical Methods in Engineering (currently online available)
   DOI: 10.1002/cnm.1008.

7. Szanto, M., Bier, W., Frage, N., Hartmann, S., and Yosibash, Z. (2007):
   Experimental based finite element simulation of cold isostatic pressing of metal powders.
   International Journal of Mechanical Science (currently online availavle)
   DOI: 10.1016/j.ijmecsci.2007.10.004.

8. Hartmann, S., and Bier, W. (2008):
High-order time integration applied to metal powder plasticity.
International Journal of Plasticity, 24:17–54.

9. Heisserer, U., Hartmann, S., Düster, A., Bier, W., Yosibash, Z., and Rank, E. (2008):
p-FEM for finite deformation powder compaction.
Computer Methods in Applied Mechanics and Engineering, 197:727–740.

Appendix B

List of symbols

Most of the symbols used in this work are listed with a short description below. For each symbol a reference is given to the page where it is introduced or used in a manner clarifying its meaning. For an overview of the concepts of the symbolism compare the notes on p. v .

B.1 Scalars

α	material parameter	18
δ_{ij}	Kronecker delta: $\delta_{ij} = 1$ if $i = j$, else 0	44
κ	penalty parameter	101
λ	material parameter	17
μ	material parameter	17
ρ	density in current configuration	12
ρ_{R}	density in reference configuration	12
ρ_{rel}	relative density in current configuration	12
$\rho_{\mathrm{R}\,\mathrm{rel}}$	relative density in reference configuration	12
ρ_0	initial relative density at time $t = 0$	12
A	surface of a body	26
A_{u_d}	part of the surface where Dirichlet (displacement) b.c. are prescribed	26
A_{t_d}	part of the surface where Neumann (force) b.c. are prescribed	26
c_{ij}	material parameter in hyperelastic strain energy functions w for the incompressible part	17
$\mathrm{d}v, \mathrm{d}V$	differential volume element in reference and current configuration	9
e	specific internal energy	13
F	yield function	21
I_1	first invariant of a tensor	20

J_2	second invariant of the deviator of a tensor	20
$\mathrm{I}_{\overline{\mathbf{C}}}, \mathrm{II}_{\overline{\mathbf{C}}}$	invariants of the unimodular right Cauchy-Green tensor $\mathbf{C}$	18
J	determinant of the deformation gradient $\mathbf{F}$	9
K	bulk modulus, material parameter in hyperelastic strain energy functions for the compressible behavior	18
n_{dof}	total number of degrees of freedom	40
n_{e}	number of elements Ω^{e}	40
n_{gr}	number of elements in column matrix $\boldsymbol{\Theta}^h$ of $\operatorname{Grad}\vec{u}$	62
n_{i}	number of integration points of the whole structure	82
n_{i}^e	number of integration points in element Ω^{e}	82
n_ξ	number of integration points in element Ω^{e} in local ξ direction	82
n_η	number of integration points in element Ω^{e} in local η direction	82
n_ζ	number of integration points in element Ω^{e} in local ζ direction	82
n_{em}	number of element modes = 'elemental' degrees of freedom in p-FEM	77
$n_{\mathrm{modes}}(x)$	number of modes in x-direction	40
$n_{\mathrm{modes}}(y)$	number of modes in y-direction	40
$n_{\mathrm{modes}}(z)$	number of modes in z-direction	40
n_{sd}	number of space, i.e. physical dimensions	25
n_{st}	number of strains/stresses in column matrix representation	61
n_{u}	number of unknown degrees of freedom	66
n_{p}	number of prescribed degrees of freedom	66
n_{b}	number of global bubble (internal) degrees of freedom	91
n_{ei}	number of local interface degrees of freedom	91
n_{eb}	number of local bubble degrees of freedom	91
p	polynomial degree of Ansatz functions	49
p	pressure	32
r	heat supply	13
t	time	
v, V	volume in current and reference configuration	12
V	$\in \mathbb{R}^{n_{\mathrm{sd}}}$ original domain	40
∂V	$\in \mathbb{R}^{n_{\mathrm{sd}}-1}$boundary of the original domain	40

B.2 Shape functions and Ansatz spaces

$\mathcal{S}_{\mathrm{ts}}^{p_\xi,p_\eta}(\Omega_{\mathrm{st}}^{\mathrm{q}})$	trunk space	46
$\mathcal{S}_{\mathrm{ps}}^{p_\xi,p_\eta}(\Omega_{\mathrm{st}}^{\mathrm{q}})$	tensor product space	46
$N_i(\xi)$	$\in \mathbb{R}$, global Ansatz function defined on Ω	40

$N_i^e(\xi)$	$\in \mathbb{R}$, local Ansatz function on element Ω^{e}	40
$N_i^p(\xi)$	$\in \mathbb{R}$, Lagrangian shape functions	44
$N_i(\xi)$	$\in \mathbb{R}$, hierarchical shape functions (integrated Legendre poly.s)	45
$\phi_j(\xi)$	$\in \mathbb{R}$, integrated Legendre polynomials	45
$L_j(\xi)$	$\in \mathbb{R}$, Legendre polynomials	45

B.3 Matrices and column matrices

$\boldsymbol{\Theta}$	$\in \mathbb{R}^{n_{\mathrm{gr}}}$, discretized displacement gradient vector $\boldsymbol{\Theta}^h$	66
$\tilde{\boldsymbol{\Phi}}^h$	$\in \mathbb{R}^{n_{\mathrm{st}}}$, vector form of the consecutive relationship	64
$\boldsymbol{\xi}_{jkl}$	Gauss point (jkl)	82
$\mathbf{A}$	matrix in the general DAE-system	16
$\boldsymbol{B}_\ell$	$\in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{dof}}}$, linear strain-displacement matrix	67
$\boldsymbol{B}_{n\ell}(\beta)$	$\in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{dof}}}$, nonlinear strain-displacement matrix	67
$\mathbf{B}^{*e}$	$\in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{em}}}$, local element strain-displacement matrix for $\mathbf{E}^e$	77
$\mathbf{B}^e$	$\in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{em}}}$, local element strain-displacement matrix for $\delta\mathbf{E}^e$ and $\Delta\mathbf{E}^e$	77
$\tilde{\boldsymbol{C}}_L^h$	$\in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{st}}}$, matrix form of the consistent tangent operator $\tilde{\mathcal{C}}$ operating on the reference configuration	64
$\boldsymbol{E}^h$	$\in \mathbb{R}^{n_{\mathrm{st}}}$, column matrix form of the Green-Lagrange strain tensor $\mathbf{E}$	61
$\mathbf{E}^e$	$\in \mathbb{R}^{n_{\mathrm{st}}}$, element Green-Lagrange strain column matrix	77
$\boldsymbol{F}$	$\in \mathbb{R}^{n_{\mathrm{u}}+n_{\mathrm{Q}}}$, DAE system consisting of the discretized variational principle and the evolution equations	83
$\mathbf{f}_{\mathrm{int}}^e$	$\in \mathbb{R}^{n_{\mathrm{modes}}}$, internal force vector belonging to element Ω^{e}	100
$\bar{\boldsymbol{f}}_{\mathrm{int}}$	$\in \mathbb{R}^{n_{\mathrm{p}}}$, part of the global internal force vector belonging to prescribed displacement dof.	100
$\boldsymbol{f}_{\mathrm{int}}$	$\in \mathbb{R}^{n_{\mathrm{u}}}$, part of the global internal force vector belonging to unknown displacement dof.	100
$\boldsymbol{g}_i$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, base vectors of convective coordinates	70
$\boldsymbol{g}$	$\in \mathbb{R}^{n_{\mathrm{dof}}}$, discretized variational principle with contributions on Gauss point level	82
$\boldsymbol{G}$	$\in \mathbb{R}^{n_{\mathrm{gr}} \times n_{\mathrm{dof}}}$, gradient displacement matrix	66
$\boldsymbol{G}_i$	$\in \mathbb{R}^{n_{\mathrm{gr}} \times n_{\mathrm{sd}}}$, one block of the gradient displacement matrix	66
$\boldsymbol{H}$	$\in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{gr}}}$, incidence matrix for the calculation of $\boldsymbol{E}^h$	62
$\mathbf{I}$	unit matrix	16
$\boldsymbol{I}_n$	$\in \mathbb{R}^n$, unit matrix	99
$\boldsymbol{k}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, body force vector	70
$\boldsymbol{K}$	$\in \mathbb{R}^{n_{\mathrm{dof}} \times n_{\mathrm{dof}}}$, global tangential stiffness matrix	78
$\mathbf{k}_C^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{em}}}$, element material stiffness matrix	78
$\mathbf{k}_G^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{em}}}$, element geometric stiffness matrix	78

$\mathbf{k}_F^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{em}}}$, element follower load stiffness matrix	78
$\boldsymbol{M}_\Theta$	$\in \mathbb{R}^{n_{\mathrm{st}} \times n_{\mathrm{gr}}}$, matrix holding entries of $\boldsymbol{\Theta}^h$ for the calculation of $\boldsymbol{E}^h$	62
$\boldsymbol{N}$	$\in \mathbb{R}^{n_{\mathrm{sd}} \times n_{\mathrm{dof}}}$, Matrix of shape functions	40
$\mathbf{N}^e$	$\in \mathbb{R}^{n_{\mathrm{sd}} \times n_{\mathrm{em}}}$ matrix of shape functions of element Ω^{e}	77
$\mathbf{q}$	column matrix holding all internal variables of the complete structure	15
$\mathbf{q}^e$	column matrix of the assembled internal variables of all Gauss points in element Ω^{e}	82
$\mathbf{q}^{e(jkl)}$	column matrix of the internal variables at Gauss point (ijk) in element Ω^{e}	82
$\boldsymbol{q}$	column matrix of the assembled internal variables of all Gauss points in a structure	82
$\boldsymbol{R}$	$\in \mathbb{R}^{n_{\mathrm{u}}+n_{\mathrm{Q}}}$, time discretized DAE-system	84
$\tilde{\mathbf{r}}$	differential equation for the evolution of internal variables $\mathbf{q}$	15
$\boldsymbol{t}_{\mathrm{R}}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, traction vector related to reference configuration	70
$\tilde{\boldsymbol{T}}^h$	$\in \mathbb{R}^{n_{\mathrm{st}}}$, column matrix representation of the 2PK stress tensor $\tilde{\mathbf{T}}$	64
$\widehat{\tilde{\boldsymbol{T}}}$	$\in \mathbb{R}^{n_{\mathrm{gr}} \times n_{\mathrm{gr}}}$, stores components of $\tilde{\mathbf{T}}$ in special order	72
$\boldsymbol{u}^h$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, discretized displacement field of the whole domain	40
$\mathbf{u}^e$	$\in \mathbb{R}^{n_{\mathrm{em}}}$ vector of element modal displacements	77
$\boldsymbol{u}$	$\in \mathbb{R}^{n_{\mathrm{dof}}}$, vector of unknown coefficients	40
$\boldsymbol{u}_a$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, column matrix of all modal displacements	66
$\boldsymbol{u}$	$\in \mathbb{R}^{n_{\mathrm{u}}}$, vector of unknown degrees of freedom	66
$\overline{\boldsymbol{u}}$	$\in \mathbb{R}^{n_{\mathrm{p}}}$, vector of prescribed degrees of freedom	66
$\mathbf{X}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, local position vector $\in Oel$ in reference configuration	77
$\boldsymbol{Z}_a^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{dof}}}$, incidence matrix for displacements dofs (global to local)	77
$\boldsymbol{Z}^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{u}}}$, incidence matrix of the free displacement dofs	77
$\overline{\boldsymbol{Z}}^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{p}}}$, incidence matrix of the prescribed displacement dofs	77
$\boldsymbol{Z}_{\boldsymbol{q}}^{e(jkl)}$	$\in \mathbb{R}^{n_{\mathrm{q}} \times n_{\mathrm{Q}}}$, incidence matrix for the internal variables of a Gauss point	82
$\boldsymbol{Z}_i^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{u}}}$, incidence matrix for the global interface dofs to local interface dofs	91
$\boldsymbol{Z}_b^e$	$\in \mathbb{R}^{n_{\mathrm{em}} \times n_{\mathrm{b}}}$, incidence matrix for the global bubbles dofs to local bubble modes	91
$\mathbf{Z}^{\mathrm{ei}}$	$\in \mathbb{R}^{n_{\mathrm{ei}} \times n_{\mathrm{em}}}$, incidence matrix for all local modes to local interface dofs	91
$\mathbf{Z}^{\mathrm{eb}}$	$\in \mathbb{R}^{n_{\mathrm{eb}} \times n_{\mathrm{em}}}$, incidence matrix for all local modes to local bubble dofs	91

B.4 Geometry: points, vectors, elements and mapping

$\kappa(t, \vec{\xi})$	motion of point $\mathbf{P}$	8
$\vec{\xi}$	vector holding the coordinates of point $\mathbf{P}$	7
$\boldsymbol{\phi}(\vec{\xi})$	mapping from standard element $\Omega_{\mathrm{st}}^{\mathrm{q}}$ to reference configuration	9

$\boldsymbol{\varphi}(\vec{\xi})$	mapping from standard element $\Omega^{\mathrm{q}}_{\mathrm{st}}$ to current configuration	9
$\vec{\chi}_{\mathrm{R}}(\vec{X}, t)$	motion with respect to reference configuration	8
Ω	$\in \mathbb{R}^{n_{\mathrm{sd}}}$ discretized domain approximating V	40
$\partial\Omega$	$\in \mathbb{R}^{n_{\mathrm{sd}}-1}$ boundary of the discretized domain Ω	40
Ω^{e}	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, discretized element — a part of Ω	40
$\Omega^{\mathrm{q}}_{\mathrm{st}}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, standard element	46
Ω_{ξ}	$\in \mathbb{R}^{n_{\mathrm{sd}}-1}$, re-parametrization of the loaded surface	34
$\mathrm{d}\vec{a}, \mathrm{d}\vec{A}$	oriented differential surface element in reference and current configuration	9
$\vec{e}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, error vector, difference between exact mathematical $\vec{u}_{\mathrm{ex}}$ and FE solution $\vec{u}^h$	59
$\vec{e}_d$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$ unit vector in direction $d, d = 1, ..., n_{\mathrm{sd}}$	25
$\vec{g}_k$	tangent vectors to the isolines of coordinates in the current configuration	9
$\vec{G}_L$	gradient vectors to the coordinate surfaces in the reference configuration	9
$\vec{k}$	volume load per unit volume	25
$\vec{n}$	surface normal in current configuration	13
$\vec{n}_{\mathrm{R}}$	surface normal in reference configuration	26
P	material point of body $\mathcal{B}$	7
$\vec{q}$	heat flux	13
$\vec{t}_{\mathrm{R}}$	surface traction vector in reference configuration	25
$\vec{t}$	surface traction vector in spatial configuration	25
$\vec{u}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$ continuous displacement field	9
$\vec{u}_{\mathrm{ex}}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, exact solution of the displacement	52
$\vec{u}_{\mathrm{rbm}}$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, rigid body mode	54
$\vec{u}^h$	$\in \mathbb{R}^{n_{\mathrm{sd}}}$, FE solution $\vec{u}^h$ of the displacement	59
$\delta\vec{u}$	variation of the displacement field $\vec{u}$	25
$\mathrm{d}\vec{x}, \mathrm{d}\vec{X}$	differential line element in reference and current configuration	9
X	label of of point P in reference configuration	7
x	label of of point P in current configuration	7
$\vec{X}$	position vector of point X in reference configuration	8
$\vec{x}$	position vector of point x in current configuration	8

B.5 Tensors

If not otherwise noted, all tensors listed here are of second order.

$(\bullet_*)$	material (pulled back) tensor	11

$(\bullet^*)$	spatial (pushed forward) tensor	11
$\hat{\mathbf{\Gamma}}$	strain tensor operating on the intermediate configuration	19
$\hat{\mathbf{\Gamma}}_{\mathrm{e}}$	elastic strain tensor operating on the intermediate configuration	19
$\hat{\mathbf{\Gamma}}_{\mathrm{v}}$	inelastic strain tensor operating on the intermediate configuration	19
$\tilde{\mathbf{\Phi}}$	constitutive relation yielding the Second Piola-Kirchhoff stress tensor	15
$\mathbf{A}$	Almansi strain tensor	10
$\mathbf{B}$	left Cauchy-Green strain tensor	10
$\overline{\mathbf{B}}$	unimodular left C.-G. tensor, volume preserving part of $\mathbf{B}$	18
$\mathbf{C}$	right Cauchy-Green strain tensor	10
$\overline{\mathbf{C}}$	unimodular right C.-G. tensor, volume preserving part of $\mathbf{C}$	18
$\hat{\mathbf{C}}_{\mathrm{e}}$	elastic right Cauchy-Green strain tensor	20
$\mathbf{C}_{\mathrm{v}}$	inelastic right Cauchy-Green strain tensor	20
$\tilde{\mathcal{C}}$	fourth order elasticity tensor operating on the reference configuration	17
$\mathbf{D}$	rate of deformation tensor, strain rate (symmetric)	10
$\mathbf{D}_{\mathrm{v}}$	inelastic strain rate	20
$\mathbf{E}$	Cauchy-Green strain tensor	10
$\delta\mathbf{E}$	variation of the Cauchy-Green strain tensor	27
$\mathbf{F}$	deformation gradient	9
$\hat{\mathbf{F}}$	volume changing (volumetric) part of the deformation gradient	9
$\overline{\mathbf{F}}$	volume preserving (isochoric) part of the deformation gradient	9
$\hat{\mathbf{F}}_{\mathrm{e}}$	elastic part of the deformation gradient	10
$\mathbf{F}_{\mathrm{v}}$	inelastic part of the deformation gradient	10
$\mathbf{G}$	deformation gradient	27
$\mathbf{I}$	second order unit tensor	17
$\mathbf{L}$	spatial velocity gradient	10
$\mathbf{L}_{\mathrm{v}}$	inelastic velocity gradient	20
$\mathbf{R}$	rotation tensor	9
$\mathbf{S}$	weighted Cauchy or Kirchhoff stress	14
$\mathbf{T}$	Cauchy stress tensor	13
$\tilde{\mathbf{T}}$	Second Piola-Kirchhoff stress tensor	14
$\hat{\mathbf{T}}$	stress in the intermediate configuration	20
$\mathbf{T}_{\mathrm{R}}$	First Piola-Kirchhoff stress tensor	13
$\hat{\mathbf{P}}$	Mandel stress tensor	20
$\mathbf{U}$	material stretch tensor	9
$\mathbf{V}$	spatial stretch tensor	9
$\mathbf{W}$	spin tensor (antimetric)	10

B.6 Work, potential energy & strain energy functions

π	virtual work / weak form	26
π_{La}	virtual work with constraints incorporated by Lagrangian multipliers	97
π_{Pen}	virtual work with constraints incorporated by penalty method	101
π_{int}	internal virtual work	27
π_{ext}	external virtual work	27
Π	potential energy	28
Π_{int}	internal potential energy	28
Π_{ext}	external potential energy	28
Ψ	strain energy density function in terms of $\mathbf{C}$	16
$\hat{\Psi}$	strain energy density function in terms of $\mathbf{F}$	16
$\tilde{\Psi}$	strain energy density function in terms of $\mathbf{E}$	16
U	strain energy density function, volume changing part	18
$w, \hat{w}$	strain energy density function, volume preserving part	18

B.7 Operators

$\mathbf{A} \cdot \mathbf{B} = a_i^{\ j} b^i_{\ j}$	inner product of two second order tensors	17
$\mathsf{D}_{\boldsymbol{u}}\mathcal{F}(\boldsymbol{x}_0)[\boldsymbol{u}]$	Gâteaux or directional derivative	29
$\delta\mathbf{A}(\boldsymbol{x}, \delta\boldsymbol{u})$	first variation of $\mathbf{A}(\boldsymbol{x})$ in direction $\delta\boldsymbol{u}$	30
$\mathbf{A}^T$	transpose of tensor $\mathbf{A}$	11
$\mathbf{A}^{-1}$	inverse of tensor $\mathbf{A}$	11
$\mathbf{A}^{-T}$	transpose of the inverse of tensor $\mathbf{A}$, i.e. $(\mathbf{A}^{-1})^T$	11
$\mathbf{A}^D$	deviator of tensor $\mathbf{A}$, i.e. $\mathbf{A} - 1/3(\mathrm{tr}\,\mathbf{A})\mathbf{I}$	20
$\langle a \rangle$	Macauley-brackets, i.e. $\langle a \rangle = 0$ for $a \leq 0$ and $\langle a \rangle = a$ for $a > 0$	20
det	determinant of a tensor	9
div	divergence with respect to current configuration	14
Div	divergence with respect to reference configuration	14
grad	gradient with respect to current configuration	10
Grad	gradient with respect to reference configuration	9
$\mathsf{L}[\mathcal{F}(\boldsymbol{x})]_{\boldsymbol{x}=\boldsymbol{x}_0}$	linear part of a function $\mathcal{F}$ at $\boldsymbol{x}_0$	30
$\mathrm{sym}(\mathbf{A})$	symmetric part of the tensor $\mathbf{A}$	27
$\mathrm{tr}\,\mathbf{A} = a^k_{\ k}$	trace of a tensor	17

Bibliography

Abbo, A. and Sloan, S. (1995). A smooth hyperbolic approximation to the Mohr-Coulomb yield criterion. *Computers & Structures*, 54:427–441.

Actis, R., Szabó, B., and Schwab, C. (1999). Hierarchic models for laminated plates and shells. *Computer Methods in Applied Mechanics and Engineering*, 172:79–107.

Ainsworth, M. (1996). A preconditioner based on domain decomposition for hp-FE approximation on quasi-uniform meshes. *SIAM Journal on Numerical Analysis*, 33(4):1358–1376.

Altenbach, J. and Altenbach, H. (1994). *Einführung in die Kontinuumsmechanik*. Teubner-Verlag.

Argyris, J., Straub, K., and Symeonidis, S. (1982). Static and dynamic stability of nonlinear elastic systems under nonconservative forces – natural approach. *Computer Methods in Applied Mechanics and Engineering*, 32:59–83.

Argyris, J. and Symeonidis, S. (1981). Nonlinear finite element analysis of elastic systems under nonconservative loading – natural formulation. Part I. quasistatic problems. *Computer Methods in Applied Mechanics and Engineering*, 26:75–123.

Arnold, D. (1981). Discretization by finite elements of a model parameter dependent problem. *Numerische Mathematik*, 37:405–421.

Arnold, M. and Frischmuth, K. (1998). Solving problems with unilateral constraints by DAE methods. *Mathematics and Computers in Simulation*, 47:47–67.

Arruda, E. and Boyce, M. (1993). A three-dimensional constitutive model for the large stretch behavior of rubber elastic materials. *Journal of the Mechanics and Physics of Solids*, 41:389–412.

ASM (1998). *ASM Handbook, Vol. 07: Powder Metal Technologies*. ASM International, 2 edition.

Aubertin, M. and Li, L. (2004). A porosity-dependent inelastic criterion for engineering materials. *International Journal of Plasticity*, 20:2179–2208.

Babuska, I. and Rank, E. (1987). An expert system-like feedback approach in the hp-version of the finite element method. *Finite Elements in Analysis and Design*, 3:127 – 147.

Babuška, I., Nobile, F., Oden, T., and Temponeand, R. (2004). Reliability, uncertainty estimates, validation and verification. ICES Report 04-05, Institute for Computational Engineering and Sciences (ICES), The University of Texas at Austin.

Babuška, I. and Oden, T. (2003). V&V in computational engineering and science. Part I: Basic Concepts. ICES Report 03-52, Institute for Computational Engineering and Sciences (ICES), The University of Texas at Austin.

Babuška, I. and Rheinboldt, W. (1979). Analysis of optimal finite element meshes in R^1. *Mathematics of Computation*, 33:435–463.

Babuška, I. and Strouboulis, T. (2001). *The finite element method and its reliability*. Oxford University Press.

Babuška, I. and Suri, M. (1992a). Locking effects in the finite element approximation of elasticity problems. *Numerische Mathematik*, 62:439–463.

Babuška, I. and Suri, M. (1992b). On locking and robustness in the finite element method. *SIAM Journal on Numerical Analysis*, 29:1261–1293.

Babuška, I. and Szabó, B. (1982). On the rates of convergence of the finite element method. *International Journal for Numerical Methods in Engineering*, 18:323–341.

Banerjee, U. and Suri, M. (1992a). Analysis of numerical integration in p-version finite element eigenvalue approximation. *Numerical Methods for Partial Differential Equations*, 8:381–394.

Banerjee, U. and Suri, M. (1992b). The effect of numerical quadrature in the p-version of the finite element method. *Mathematics of Computation*, 59(199):1–20.

Bathe, K. (1996). *Finite element procedures*. Prentice Hall.

Bathe, K. (2002). *Finite-Elemente-Methoden*. Springer-Verlag.

Bazilevs, Y., Beirão da Veiga, L., Cottrell, J., Hughes, T. J. R., and Sangalli, G. (2006). Isogeometric analysis: approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, 16(7):1031–1090.

Belytschko, T., Liu, W., and Moran, B. (2000). *Nonlinear finite elements for continua and structures*. John Wiley & Sons.

Belytschko, T., Liu, W. K., and Engelmann, E. (1986). The gamma-elements and related concepts. In Hughes, T. J. R. and Hinton, E., editors, *Finite Element Methods for Plate and Shell Structures*, volume 1: Element Technology, pages 316–347. PineridgePress International, Swansea, UK.

Belytschko, T. and Tsay, C. (1983). A stabilization procedure for the quadrilateral plate bending element with one-point quadrature. *International Journal for Numerical Methods in Engineering*, 19:405–420.

Betsch, P. (1996). *Statische und dynamische Berechnungen von Schalen endlicher elastischer Deformationen mit gemischten finiten Elementen*. PhD thesis, Institut für Baumechanik und Numerische Mechanik, Universität Hannover.

Bier, W., Dariel, M., Frage, N., Hartmann, S., and Michailov, O. (2007). Die compaction of copper powder designed for material parameter identification. *International Journal of Mechnical Sciences*, 49:766–777.

Bier, W. and Hartmann, S. (2006). A finite strain constitutive model for metal powder compaction using a unique and convex single surface yield function. *European Journal of Mechanics Series A/Solids*, 25:1009 – 1030.

Bigoni, D. and Piccolroaz, A. (2004). Yield criteria for quasibrittle and frictional materials. *International Journal of Solids and Structures*, 41:2855–2878.

Bischoff, M. (1999). *Theorie und Numerik einer dreidimensionalen Schalenformulierung*. PhD thesis, Institut für Baustatik, Universität Stuttgart.

Bischoff, M., Wall, W., and Ramm, E. (1999). Stabilized enhanced assumed strain elements for large strain analysis without artificial kinematic modes. In *Proceedings of ECCM '99, European Conference on Computational Mechanics*, München, Germany.

Bletzinger, K.-U., Bischoff, M., and Ramm, E. (2000). A unifed approach for shear-locking-free triangular and rectangular shell fnite elements. *Computers & Structures*, 75:321–334.

Bonet, J. and Wood, R. (1997). *Nonlinear continuum mechanics for finite element analysis*. Cambridge University Press, New York.

Brenan, K., Campbell, S., and Petzold, L. (1996). *Numerical Solutions of Initial-Value Problems in Differential-Algebraic Equations*. Classics in Applied Mathematics. SIAM, Philadelphia, PA.

Bröker, H. (2001). *Integration von geometrischer Modellierung und Berechnung nach der p-Version der FEM*. PhD thesis, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München.

Bufler, H. (1984). Pressure loaded structures under large deformations. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik*, 64:287–295.

Bufler, H. (1985). Zur Potentialeigenschaft der von einer Flüssigkeit herrührenden Druckbelastung. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik*, 65:130–132.

Chen, Q. and Babuška, I. (1995). Approximate optimal points for polynomial interpolation of real functions in an interval and in a triangle. *Computer Methods in Applied Mechanics and Engineering*, 128:405–417.

Christensen, P. (2000). *Computational nonsmooth mechanics: Contact, friction and plasticity*. Doctoral thesis, Department of Mechnical Engineering, Linköping University, Linköping Studies in Science and Technology, No. 657.

Chung, J. and Hulbert, G. (1993). A time integration algorithm for structural dynamics with improved numerical dissipation: the generalized-α method. *Journal of Applied Mechanics, Transactions ASME*, 60:371–375.

Clough, R. W. (2004). Early history of the finite element method from the view point of a pioneer. *International Journal for Numerical Methods in Engineering*, 60:283–287.

Cook, R. D., Malkus, D. S., and Plesha, M. E. (2001). *Concepts and Applications of Finite Element Analysis*. Wiley, 4rd edition.

Cottrell, J., Reali, A., Bazilevs, Y., and Hughes, T. J. R. (2006). Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering*, 195:5257–5296.

Crisfield, M. (1991). *Non-linear finite element analysis of solids and structures, Volume 1*. John Wiley & Sons.

Demkowicz, L. (2004). Finite element methods for maxwell equations. In Stein, E., de Borst, R., and Hughes, T., editors, *Encyclopedia of Computational Mechanics*, pages 723–737. John Wiley & Sons.

Demkowicz, L. (2006). *Computing with hp-adaptive finite elements: one and two dimensional elliptic and Maxwell problems*, volume 1. Chapman & Hall / CRC Applied Mathematics & Nonlinear Science.

Dhondt, G. (2004). *The Finite Element Method for Three-dimensional Thermomechanical Applications*. Wiley, Chichester, England.

Doherty, W. P., Wilson, E. L., and Taylor, R. L. (1969). Stress analysis of axisymmetric solids using higher order quadrilateral finite elements. Technical report, Struct. Eng. Lab. Report SESM 69-3.

Düster, A. (2001). *High order finite elements for three-dimensional, thin-walled nonlinear continua*. PhD thesis, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München.

Düster, A. (2005). High order fem. Lecture notes, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München, Germany.

Düster, A., Bröker, H., Heidkamp, H., Heisserer, U., Kollmannsberger, S., Krause, R., Muthler, A., Niggl, A., Nübel, V., Rücker, M., and Scholz, D. (2004). *AdhoC^4 – User's Guide*. Lehrstuhl für Bauinformatik, Technische Universität München.

Düster, A., Bröker, H., and Rank, E. (2001). The p-version of the finite element method for three-dimensional curved thin walled structures. *International Journal for Numerical Methods in Engineering*, 52:673–703.

Düster, A., Hartmann, S., and Rank, E. (2003). p-fem applied to finite isotropic hyperelastic bodies. *Computer Methods in Applied Mechanics and Engineering*, 192:5147–5166.

Düster, A., Niggl, A., Nübel, V., and Rank, E. (2002). A numerical investigation of high-order finite elements for problems of elasto-plasticity. *Journal of Scientific Computing*, 17:429–437.

Düster, A., Niggl, A., and Rank, E. (2007). Applying the hp-d version of the FEM to locally enhance dimensionally reduced models. *Computer Methods in Applied Mechanics and Engineering*, 196:3524–3533.

Düster, A. and Rank, E. (2001). The p-version of the finite element method compared to an adaptive h-version for the deformation theory of plasticity. *Computer Methods in Applied Mechanics and Engineering*, 190:1925–1935.

Düster, A. and Rank, E. (2002). A p-version finite element approach for two- and three-dimensional problems of the J_2 flow theory with non-linear isotropic hardening. *International Journal for Numerical Methods in Engineering*, 53:49–63.

Dvorkin, E. and Bathe, K. (1984). A continuum mechanics based four-node shell element for general nonlinear analysis. *Engineering Computations*, 1:77–88.

Ehlers, W. (1995). A single-surface yield function for geomaterials. *Archive of Applied Mechanics*, 65:246–259.

Ellsiepen, P. and Hartmann, S. (2001). Remarks on the interpretation of current non-linear finite element analyses as differential-algebraic equations. *International Journal for Numerical Methods in Engineering*, 51:679–707.

Engineering Software Research & Development, I. (2006). *StressCheck Release 7.0 – Users Guide*. Engineering Software Research & Development, St. Louis.

Felippa, C. A. (1977). Error analysis of penalty function techniques of constraint definition in linear algebraic systems. *International Journal for Numerical Methods in Engineering*, 11:709–728.

Flanagan, D.P. und Belytschko, T. (1981). A uniform strain hexahedron and quadrilateral with orthogonal hourglass control. *International Journal for Numerical Methods in Engineering*, 17(5):679–706.

Flory, P. (1961). Thermodynamic relations for high elastic materials. *Transaction of the Faraday Society*, 57:829–838.

Fritzen, P. (1997). *Numerische Behandlung nichtlinearer Probleme der Elastizitäts- und Plastizitätstheorie*. Doctoral thesis, Department of Mathematics, University of Darmstadt.

German, R. M. (1998). *Powder metallurgy of iron and steel*. Wiley, New York.

Gordon, W. and Hall, C. (1973a). Construction of curvilinear co-ordinate systems and applications to mesh generation. *International Journal for Numerical Methods in Engineering*, 7:461–477.

Gordon, W. and Hall, C. (1973b). Transfinite element methods: Blending function interpolation over arbitrary curved element domains. *Numerische Mathematik*, 21:109–129.

Graeff-Weinberg, K. (1995). *Ein Finite-Element-Konzept zur lokalen Netzverdichtung und seine Anwendung auf Koppel- und Kontaktprobleme*. PhD thesis, Otto-von-Guericke-Universität Magdeburg.

Grossmann, C. and Roos, H.-G. (2005). *Numerik partieller Differentialgleichungen*. Vieweg.

Gurtin, M. (2003). *An introduction to continuum mechanics*. Academic Press.

Hartmann, S. (2001). Numerical studies on the identification of the material parameters of Rivlin's hyperelasticity using tension-torsion tests. *Acta Mechanica*, 148:129–155.

Hartmann, S. (2002). Computation in finite strain viscoelasticity: finite elements based on the interpretation as differential-algebraic equations. *Computer Methods in Applied Mechanics and Engineering*, 191(13-14):1439–1470.

Hartmann, S. (2003a). *Finite-Elemente Berechnung inelastischer Kontinua - Interpretation als Algebro-Differentialgleichungssysteme*. Postdoctoral thesis, Institut für Mechanik, Universität Kassel.

Hartmann, S. (2003b). On displacement control within the DIRK/MLNA approach in nonlinear finite element analysis. In Bathe, K.-J., editor, *Computational Fluid and Solid Mechanics 2003*, volume 1, pages 316–319, Oxford. Elsevier.

Hartmann, S. (2004). Newton- vs. Multilevel-Newton method in the FEM. *Proceedings in Applied Mathematics and Mechanics*, 4:318–319.

Hartmann, S. (2005). A remark on the application of the Newton-Raphson method in nonlinear finite element analysis. *Computational Mechanics*, 36:100–116.

Hartmann, S. and Bier, W. (2008). High-order time integration applied to metal powder plasticity. *International Journal of Plasticity*, 24:17–54.

Hartmann, S., Lührs, G., and Haupt, P. (1997). An efficient stress algorithm with applications in viscoplasticity and plasticity. *International Journal for Numerical Methods in Engineering*, 40:991–1013.

Hartmann, S. and Neff, P. (2003). Polyconvexity of generalized polynomial-type hyperelastic strain energy functions for near-incompressibility. *International Journal of Solids and Structures*, 40:2767–2791.

Hartmann, S., Quint, K. J., and Hamkar, A.-W. (2008). Displacement control in time-adaptive non-linear finite-element analysis. *ZAMM-Zeitschrift für Angewandte Mathematik und Mechanik*, 88(5):342–364.

Hartmann, S. and Wensch, J. (2007). Finite element analysis of viscoelastic structures using Rosenbrock-type methods. *Computational Mechanics*, 40(2):383–398.

Haupt, P. (2000). *Continuum mechanics and theory of materials*. Springer.

Haupt, P. and Tsakmakis, C. (1989). On the application of dual variables in continuum mechanics. *Journal of Continuum Mechanics and Thermodynamics*, 1:165–196.

Haupt, P. and Tsakmakis, C. (1996). Stress tensors associated with deformation tensors via duality. *Archive of Mechanics*, 48:347–384.

Heisserer, U. (2001). *Solution of the semidiscrete equations of structural dynamics by the generalized-alpha method and its implementation in a p-FEM code.* Diploma thesis, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München.

Heisserer, U. (2007). *AdhoC^4 – Technical Guide.* Lehrstuhl für Bauinformatik, Technische Universität München.

Heisserer, U., Düster, A., and Rank, E. (2005). Follower loads for axisymmetric high order finite elements. In *Proceedings in Applied Mathematics and Mechanics*, volume 5, pages 405–406.

Heisserer, U., Hartmann, S., Düster, A., and Yosibash, Z. (2007). On volumetric locking-free behavior of p-version finite elements under finite deformations. *Communications in Numerical Methods in Engineering (in press) DOI: 10.1002/cnm.1008.*

Hellinger, E. (1914). Die allgemeinen Ansätze der Mechnanik der Kontinua. In Klein, F. and Müller, C., editors, *Enzyklopädie der Mathematischen Wissenschaften*, volume IV Pt. 4 of *8*, pages 601–694. Teubner Verlag, Leipzig.

Hibbitt, H. (1979). Some follower forces and load stiffness. *International Journal for Numerical Methods in Engineering*, 14:937–941.

Hinnant, H. (1994). A fast method of numerical quadrature for p-version finite element matrices. *International Journal for Numerical Methods in Engineering*, 37:3723–3750.

Holzap fel, G. (2000). *Nonlinear solid mechanics.* John Wiley & Sons.

Holzer, S., Rank, E., and Werner, H. (1990). An implementation of the hp-version of the finite element method for Reissner-Mindlin plate problems. *International Journal for Numerical Methods in Engineering*, 30:459–471.

Holzer, S. and Yosibash, Z. (1996). The p-version of the finite element method in incremental elasto-plastic analysis. *International Journal for Numerical Methods in Engineering*, 39:1859–1878.

Hughes, T. J. R. (2000). *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis.* Dover Publications.

Hughes, T. J. R., Cottrell, J. A., and Bazilevs, Y. (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194:4135–4195.

Hughes, T. J. R. and Pister, K. (1978). Consistent linearization in mechanics of solids and structures. *Computers & Structures*, 8:391–397.

Hughes, T. J. R., Taylor, R. L., and Kanoknukulchai, W. (1977). A simple and effcient finite element for plate bending. *International Journal for Numerical Methods in Engineering*, 11:1529–1543.

Hughes, T. J. R. and Tezduyar, T. E. (1981). Finite elements based upon Mindlin plate theory with particular reference to the four-node bilinear isoparametric element. *Journal of Applied Mechanics*, pages 587–596.

Implicit Function Theorem (Webpage visited 01.03.2007). `http://www.ualberta.ca/dept/math/gauss/fcm/calculus/multvrbl/basic/ImplctFnctns/implct_fnctn_thrm.htm`.

Irons, B. (1966). Engineering application of numerical integration in stiffness methods. *Journal of the American Institute of Aeronautics and Astronautics*, 14:2035–2037.

ISO/TC119 (1995). *ISO 3953: Metallic powders - determination of tap tensity.* ISO, 1995-02 edition.

Jeremic, B. and Xenophontos, C. (1999). Application of the p-version of the finite element method to elasto-plasticity with localization of deformation. *Communications in Numerical Methods in Engineering*, 15, No.12:867–876.

Keller, A. (1989). *Sprachphilosophie.* Alber, 2 edition.

Kim, C.-G. (1998). The effect of quadrature errors in practice. *Kangweon-Kyungki Math. Jour.*, 6(2):195–203.

Kim, C.-G. and Suri, M. (1993). On the p version of the finite element method in the presence of numerical integration. *Numerical Methods for Partial Differential Equations*, 9:593–629.

Királyfalvi, G. and Szabó, B. (1997). Quasi-regional mapping for the p-version of the finite element method. *Finite Elements in Analysis and Design*, 27:85–97.

Koiter, W. (1996). Unrealistic follower forces. *Journal of sound and vibration*, 194:636–636.

Koizumi, M. and Nishihara, M., editors (1991). *Isostatic pressing. Technology and Applications.* Elsevier, New York.

Koschnick, F. (2004). *Geometrische Locking-Effekte bei Finiten Elementen und ein allgemeines Konzept zu ihrer Vermeidung.* PhD thesis, Lehrstuhl für Statik, Technische Universität München.

Koslo, D. and Frazier, G. (1978). Treatment of hourglass pattern in low order finite element codes. *International Journal for Numerical and Analytical Methods in Geomechanics*, 2:57–72.

Krantz, S. and Parks, H. (2003). *The implicit function theorem.* Birkhäuser.

Krause, R. (1996). *Multiscale computations with a combined h- and p-version of the finite-element method.* PhD thesis, Fach Numerische Methoden und Informationsverarbeitung, Universität Dortmund.

Krause, R., Mücke, R., and Rank, E. (1995). hp-version finite elements for geometrically nonlinear problems. *Communications in Numerical Methods in Engineering*, 11:887–897.

Krause, R. and Rank, E. (2003). Multiscale computations with a combination of the h- and p-versions of the finite-element method. *Computer Methods in Applied Mechanics and Engineering*, 192:3959–3983.

Kreisselmeier, G. and Steinhauser, R. (1979). Systematische Auslegung von Reglern durch Optimierung eines vektoriellen Gütekriteriums. *Regelungstechnik*, 3:76–79.

Kutta, W. (1901). Beitrag zur näherungsweisen Integration totaler Differentialgleichungen. *Zeitschrift für Mathematik und Physik*, 46:435–453.

Lassner, E. and Schubert, W.-D. (1999). *Tungsten: Properties, Chemistry, Technology of the Element, Alloys, and Chemical Compounds*. Springer, New York, 1 edition.

Li, A.-M., Simon, U., and Zhao, G. (1993). *Global Affine Differential Geometry of Hypersurfaces*. De Gruyter Expositions in Mathematics. Walter de Gruyter, Berlin, 1 edition.

Luenberger, D. G. (2003). *Linear and Nonlinear Programming*. Kluwer, Norwell, Massachsetts, USA, 2 edition.

Lüthering, G. and Williams, J. C. (2003). *Titanium*. Springer, Berlin, 1 edition.

Malkus, D. and Hughes, T. J. R. (1978). Mixed finite element methods – reduced and selective integration techniques: a unification of concepts. *Computer Methods in Applied Mechanics and Engineering*, 15:63–81.

Mandel, J. (1990a). Iterative solvers by substructuring for the p-version finite element method. *Computer Methods in Applied Mechanics and Engineering*, 80:117–128.

Mandel, J. (1990b). On block diagonal and Schur complement preconditioning. *Numerische Mathematik*, 58:79–93.

Mang, H. (1980). Symmetricability of pressure stiffness matrices for shells with loaded free edges. *International Journal for Numerical Methods in Engineering*, 15:981–990.

Mansfeld, J., editor (1983). *Die Vorsokratiker I. Auswahl der Fragmente.*, volume 1. Reclam, Ditzingen.

Mansfeld, J., editor (1986). *Die Vorsokratiker II. Zenon, Empedokles, Anaxagoras, Leukipp, Demokrit. Auswahl der Fragmente.*, volume 2. Reclam, Ditzingen.

Marsden, J. E. and Hughes, T. J. R. (1993). *Mathematical foundations of elasticity*. Prentice-Hall.

Melenk, J., Gerdes, K., and Schwab, C. (2001). Fully discrete hp finite elements: fast quadrature. *Computer Methods in Applied Mechanics and Engineering*, 190:4339–4364.

Meyberg, K. and Vachenauer, P. (2003). *Höhere Mathematik 1: Differential- und Integralrechnung. Vektor- und Matrizenrechnung*, volume 1. Springer, 6 edition.

Meyberg, K. and Vachenauer, P. (2006). *Höhere Mathematik 2: Differentialgleichungen, Funktionentheorie, Fourier-Analysis, Variationsrechnung*, volume 2. Springer, 4 edition.

Mok, D., Wall, W., Bischoff, M., and Ramm, E. (1999). Algorithmic aspects of deformation dependent loads in non-linear static finite element analysis. *Engineering Communications*, 16(5):601–618.

Mooney, M. (1940). Theory of large elastic deformation. *Journal of Applied Physics*, 11:582–592.

NASA (1999). Mars climate orbiter mishap investigation board phase I report. `ftp://ftp.hq.nasa.gov/pub/pao/reports/1999/MCO_report.pdf`.

Nelson, B. and Kirby, R. M. (2006). Ray-tracing polymorphic multi-domain spectral/hp elements for isosurface rendering. *IEEE Transactions on Visualization and Computer Graphics*, 12(1):114–125.

Niggl, A. (2007). *Tragwerksanalyse am volumenorientierten Gesamtmodell*. PhD thesis, Lehrstuhl für Bauinformatik, Technische Universität München.

Niggl, A., Düster, A., and Rank, E. (2003). Coupling 1d and 2d elasticity problems by using the *hp-d* -version of the finite element method. In *Proceedings of the Second M.I.T Conference on Computational Fluid and Solid Mechanics*, Cambridge, USA.

Noel, A. and Szabó, B. (1997). Formulation of geometrically non-linear problems in the spatial reference frame. *International Journal for Numerical Methods in Engineering*, 40:1263–1280.

Nübel, V. (2005). *Die adaptive rp-Methode für elastoplastische Probleme*. PhD thesis, Lehrstuhl für Bauinformatik, Technische Universität München.

Nübel, V., Düster, A., and Rank, E. (2000). Die Methode Vektorintegration bei der p-Version der Finite-Elemente-Methode. Technical Report, Lehrstuhl für Bauinformatik, TU München.

Nübel, V., Düster, A., and Rank, E. (2001). Adaptive vector integration as an efficient quadrature scheme for p-version finite element matrices. In *Proceedings of the European Conference on Computational Mechanics 2001*, Cracow, Poland.

Nübel, V., Düster, A., and Rank, E. (2007). An *rp*-adaptive finite element method for elastoplastic problems. *Computational Mechanics*, 39:557–574.

Ogden, R. (1972). Large deformation isotropic elasticity - on the correlation of theory and experiment for incompressible rubberlike solids. *Proceedings of the Royal Society of London*, Series A 326:565–584.

Papen, E. (1977). Isostatic pressing. In Spain, I. L. and Paauwe, I., editors, *High Pressure Technology*, volume 2, chapter 9, pages 339–390. CRC, New York, 2 edition.

Parisch, H. (2003). *Festkörper-Kontinuumsmechanik. Von den Grundgleichungen zur Lösung mit Finiten Elementen*. Teubner, Stuttgart.

Phan-Thien, N. (2002). *Understanding Viscoelasticity. Basics of Rheology.* Advanced Texts in Physics. Springer, Berlin, 1 edition.

Popper, K. R. S. (1959). *The logic of scientific discovery.* Hutchinson, London.

Price, P. E. (1998). Cold Isostatic Pressing. In *ASM Handbook, Vol. 07: Powder Metal Technologies*, pages 382 – 388. ASM International, 2 edition.

Rabbat, G. N. B., Sangiovanni-Vincentelli, A., and Hsieh, H. (1979). A multilevel Newton algorithm with macromodelling and latency for the analysis of large-scale nonlinear cicuits in the time domain. *IEEE transactions on cicuits and systems*, 26(9):733–741.

Rank, E. (1992a). Adaptive remeshing and h-p domain decomposition. *Computer Methods in Applied Mechanics and Engineering*, 101:299–313.

Rank, E. (1992b). A combination of hp-version finite elements and a domain decomposition method. In *Proc. of the First European Conference on Numerical Methods in Engineering*, Brüssel.

Rank, E. (1993). A zooming-technique using a hierarchical *hp*-version of the finite element method. In Whiteman, J., editor, *The Mathematics of Finite Elements and Applications - Highlights 1993*. Elsevier, Uxbridge.

Rank, E., Bröker, H., Düster, A., Krause, R., and Rücker, M. (2002). The p-version of the finite element method for structural problems. In Stein, E., editor, *Error-controlled Adaptive Finite Elements in Solid Mechanics*, chapter 8, pages 263–307. John Wiley & Sons.

Rank, E., Bröker, H., Düster, A., and Rücker, M. (1998a). Neue Modellierungskonzepte für FE-Berechnungen mit Ansätzen höherer Ordnung. In Wriggers, P., Meißner, U., Stein, E., and Wunderlich, W., editors, *Finite Elemente in der Baupraxis - Modellierung, Berechnung und Konstruktion*, pages 421–430. Ernst & Sohn.

Rank, E., Bröker, H., Düster, A., and Rücker, M. (2001). Integrierte Modellierungs- und Berechnungssoftware für den konstruktiven Ingenieurbau: Die p-Version und geometrische Elemente. *Bauingenieur*, 76:53–61.

Rank, E. and Krause, R. (1997). A multiscale finite-element-method. *Computers & Structures*, 64:139–144.

Rank, E., Krause, R., and Preusch, K. (1998b). On the accuracy of p-version elements for the Reissner-Mindlin plate problem. *International Journal for Numerical Methods in Engineering*, 43:51–67.

Rank, E. and Zienkiewicz, O. (1987). A simple error estimator in the finite element method. *Communications in Applied Numerical Methods*, 3:243–249.

Reese, S., Küssner, M., and Reddy, B. (1999). A new stabilization technique for finite elements in non-linear elasticity. *International Journal for Numerical Methods in Engineering*, 44:1617–1652.

Reese, S. and Wriggers, P. (2000). A stabilization technique to avoid hourglassing in finite elasticity. *International Journal for Numerical Methods in Engineering*, 48:79–109.

Reissner, E. (1950). On a variational theorem in elasticity. *Journal of Mathematics and Physics*, 29(8):90–95.

Richerson, D. W. (2006). *Modern ceramic engineering: properties, processing, and use in design*. Taylor & Francis Group.

Rivlin, R. (1948). Large elastic deformations of isotropic materials I. Fundamental concepts. *Philosophical Transaction of the Royal Society of London*, Series A 240:459–490.

Rivlin, R. and Saunders, D. (1951). Large elastic deformations of isotropic materials VII. Experiments on the deformation of rubber. *Philosophical Transaction of the Royal Society of London*, Series A 243:251–288.

Rumpel, T. and Schweizerhof, K. (2004). Hydrostatic fluid loading in non-linear finite element analysis. *International Journal for Numerical Methods in Engineering*, 59:849–850.

Runge, C. (1895). Über die numerische Auflösung von Differentialgleichungen. *Mathematische Annalen*, 46:167–178.

Sagawa, M., Nagata, H., Watanabe, T., and Itani, O. (2000). Rubber isostatic pressing (RIP) of powders for magnets and other materials. *Materials & Design*, 21:243–249.

Schneider, H. (1990). *Flüssigkeitsbelastete Membranen unter großen Deformationen*. PhD thesis, Institut für Mechanik (Bauwesen), University of Stuttgart.

Scholz, D. (2006). *An anisotropic p-adaptive method for linear elastostatic and elastodynamic analysis of thin-walled and massive structures*. Dissertation, Lehrstuhl für Bauinformatik, Fakultät für Bauingenieur- und Vermessungswesen, Technische Universität München.

Schwab, C. (1998). *p- and hp-finite element methods, theory and applications in solid and fluid mechanics*. Oxford University Press.

Schwarz, H. (2004). *Numerische Mathematik*. B.G. Teubner, 5. edition.

Schweizerhof, K. (1982). Nichtlineare Berechnung von Tragwerken unter verformungsabhängigen Belastungen mit finiten Elementen. technical report 1982-2, Institut für Baustatik, Stuttgart.

Schweizerhof, K. and Ramm, E. (1984). Displacement dependent pressure loads in nonlinear finite element analyses. *Computers & Structures*, 18:1099–1114.

Sederberg, T. W., Cardon, D. L., Finnigan, G. T., North, N. S., Zheng, J., and Lyche, T. (2004). T-spline simplification and local refinement. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 276–283, New York, NY, USA. ACM Press.

Sederberg, T. W., Zheng, J., Bakenov, A., and Nasri, A. (2003). T-splines and T-NURCCs. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, pages 477–484, New York, NY, USA. ACM Press.

Sewell, M. (1967). On configuration-dependent loading. *Archives of Rational Mechanics*, 23:321–351.

Simo, J. and Armero, F. (1992). Geometrically non-linear enhanced strain mixed methods and the method of incompatible modes. *International Journal for Numerical Methods in Engineering*, 33:1413–1449.

Simo, J., Armero, F., and Taylor, R. (1993). Improved versions of assumed enhanced strain trilinear elements for 3D finite deformation problems. *Computer Methods in Applied Mechanics and Engineering*, 110:359–386.

Simo, J. and Rifai, M. (1990). A class of mixed assumed strain methods and the method of incompatible modes. *International Journal for Numerical Methods in Engineering*, 29:1595–1638.

Simo, J. and Taylor, R. (1985). Consistent tangent operators for rate-depedent elastoplasticity. *Computer Methods in Applied Mechanics and Engineering*, 48:101–118.

Simo, J., Taylor, R., and Wriggers, P. (1991). A note on finite-element implementation of pressure boundary loading. *Communications in Applied Numerical Methods*, 7:513–525.

Simo, J. C. and Hughes, T. J. R. (1986). On variational foundations of assumed strain methods. *Journal of Applied Mechanics*, 53:51–54.

Simo, J. C. and Pister, K. S. (1984). Remarks on rate constitutive equations for finite deformation problems: Computational implications. *Computer Methods in Applied Mechanics and Engineering*, 46:201–215.

Solín, P., Segeth, K., and Doležel, I. (2003). *Higher-order finite element methods*. Studies in Advanced Mathematics. Chapman & Hall, Boca Raton.

Suri, M. (1996). Analytical and computational assessment of locking in the hp finite element method. *Computer Methods in Applied Mechanics and Engineering*, 133:347–371.

Szabó, B., Actis, R., and Holzer, S. (1995). Solution of elastic-plastic stress analysis problems by the p-version of the finite element method. In Babuška, I. and Flaherty, J., editors, *Modeling, mesh generation, and adaptive numerical methods for partial differential equations*, IMA Volumes in Mathematics and its Applications, Vol. 75, pages 395–416. Springer, New York.

Szabó, B. and Babuška, I. (1991). *Finite element analysis*. John Wiley & Sons.

Szabó, B., Düster, A., and Rank, E. (2004). The p-version of the Finite Element Method. In Stein, E., de Borst, R., and Hughes, T. J. R., editors, *Encyclopedia of Computational Mechanics*, volume 1, chapter 5, pages 119–139. John Wiley & Sons.

Szanto, M., Bier, W., Frage, N., Hartmann, S., and Yosibash, Z. (in press 2007). Experimental based finite element simulation of cold isostatic pressing of metal powders. *International Journal of Mechanical Science*.

Tonti, E. (1972). On the mathematical structure of a large class of physical theories. Accademia Nazionale dei Lincei, estratto dai Rendiconti della Classe di Scienze fisiche, matematiche e naturali, Serie VIII, Vol. LII, fasc. 1, Gennaio.

Tonti, E. (2003). A classification diagram for physical variables. `http://discretephysics.dic.units.it/papers/diagrammi.pdf`. preliminary draft.

Trefethen, L. N. (2007). Is Gauss-quadrature better than Clenshaw-Curtis? *SIAM review, to appear. preprint available at* ***http://web.comlab.ox.ac.uk/oucl/work/nick.trefethen/CC_trefethen_revised2.pdf***.

Truesdell, C. and Noll, W. (1965). The non-linear field theories of mechanics. In Flügge, S., editor, *Encyclopedia of Physics*, volume III/3. Springer Verlag, Berlin.

Überhuber, C. W. (2001). *Computer-Numerik II*. Springer, Berlin.

Vogelius, M. (1983). An analysis of the p-version of the finite element method for nearly incompressible materials; uniformly optimal error estimates. *Numerische Mathematik*, 41:39–53.

Weisstein, E. W. (2004). Affine transformation. MathWorld–A Wolfram Web Resource. `http://mathworld.wolfram.com/AffineTransformation.html`.

Weisstein, E. W. (2005). L'hospital's rule. MathWorld–A Wolfram Web Resource. `http://mathworld.wolfram.com/LHospitalsRule.html`.

Westermann, T. (1997). *Mathematik für Ingenieure mit Maple*. Springer, Berlin, 1 edition.

Wilson, E. (1965). Structural analysis of axisymmetric solids. *AIAA Journal*, 3(12):2269–2274.

Wilson, E., Taylor, R., Docherty, W., and Ghaboussi, J. (1973). Incompatible displacement models. In S.J., F., editor, *Numerical and computer models in structural mechanics*. Academic Press, New York.

Woo, K., Hong, C., and Basu, P. (2003). Materially and geometrically nonlinear analysis of laminated anisotropic plates by the *p*-version of fem. *Computers & Structures*, 81:1653–1662.

Wood, W., Bossak, M., and Zienkiewicz, O. (1981). An alpha modification of newmark's method. *International Journal for Numerical Methods in Engineering*, 5:1562–1566.

Wriggers, P. (2001). *Nichtlineare Finite-Element-Methoden*. Springer-Verlag.

Wriggers, P. (2006). *Compuational Contact Mechanics*. Springer, Berlin.

Wriggers, P. and Korelc, J. (1996). On enhanced strain methods for small and finite deformations of solids. *Computational Mechanics*, 18:413–428.

Wriggers, P. and Reese, S. (1996). A note on enhanced strain methods for large deformations. *Computer Methods in Applied Mechanics and Engineering*, 135:201–209.

Yang, H., Kim, J., and Kim, K. (2004). Rubber isostatic pressing and cold isostatic pressing of metal powder. *Materials Science and Engineering A*, 382(1–2):41–49.

Yosibash, Z., Hartmann, S., Heisserer, U., Düster, A., Rank, E., and Szanto, M. (2007). Axisymmetric pressure boundary loading for finite deformation analysis using p-FEM. *Computer Methods in Applied Mechanics and Engineering*, 196:1261–1277.

Zienkiewicz, O. and Craig, A. (1986). Adaptive refinement, error estimates, multigrid solution, and hierarchic finite element method concepts. In Babuška, I., Zienkiewicz, O., Gago, J., and Olivera, E. d. A., editors, *Accuracy estimates and adaptive refinements in finite element computations*, pages 25–59. John Wiley & Sons.

Zienkiewicz, O. and Taylor, R. (1989). *The Finite Element Method – Basic Formulations and Linear Problems*, volume 1. McGraw-Hill Book Company, 4th edition.

Zienkiewicz, O. and Taylor, R. (2000a). *The Finite Element Method – Solid Mechanics*, volume 2. Butterworth-Heinemann, 5th edition.

Zienkiewicz, O. and Taylor, R. (2000b). *The Finite Element Method – The Basis*, volume 1. Butterworth-Heinemann, 5th edition.

Zienkiewicz, O., Taylor, R., Sherwin, S., and Peiro, J. (2003). On discontinuous Galerkin methods. *International Journal for Numerical Methods in Engineering*, 58:1119–1148.

Zienkiewicz, O., Taylor, R., and Too, J. (1971). Reduced integration technique in general analysis of plates and shells. *International Journal for Numerical Methods in Engineering*, 3:275–290.