

FE–Umsetzung von nichtlinearem Materialverhalten mit ABAQUS

TU Darmstadt — FB 13 / Festkörpermechanik
Sommersemester 2011

apl. Prof. Dr.–Ing. HERBERT BAASER
Herbert@BaaserWeb.de

Stand: 21. Juni 2011
Bingen & Darmstadt

— *Arbeitsversion – nicht korrigierte Fassung* —

Inhaltsverzeichnis

1	Einleitung	4
2	Struktur der Lehrveranstaltung	5
2.1	Ablauf & Bewertung	5
2.2	Zeitplan & Themen	5
3	Verwendung von ABAQUS	7
3.1	Login und Verzeichnisse	7
3.2	Struktur von ABAQUS	7
3.2.1	Kurz-Tutorium am Beispiel eines O-Rings	7
3.2.2	Direkter Aufruf von der Konsole	12
I	Theorie — Grundlagen	13
4	Mathematische Grundlagen	14
4.1	Skalare, Vektoren und Tensoren	14
4.2	Hauptachsentransformation	17
4.3	Notation, Matrix-Darstellung	17
4.4	Polare Zerlegung	17
5	Elemente der Kontinuumsmechanik — Grundlagen der Festkörpermechanik	18
5.1	Deformationsgradient und Verzerrungsmaße	18
5.1.1	Beispiele	20
5.2	Verzerrungsraten	22
5.3	Deformationsinvarianten	22
5.4	Mechanische Spannung — Konjugierte Spannungstensoren	24
5.4.1	VOIGT-Notation	24
5.4.2	Spannungsleistung & konjugierte Spannungstensoren	26
5.4.3	Vorgehen in FEM-Programmsystemen	26

5.4.4	Spannungsraten	27
5.5	Verknüpfung von Spannung und Dehnung — Materialmodul	28
6	Elastisches Materialverhalten	29
6.1	Linear-elastisches Materialverhalten — HOOKEsches Gesetz	29
6.2	Hyperelastizität	30
6.2.1	Quasi-inkompressible Darstellung	31
7	Inelastisches Materialverhalten	34
7.1	Motivation. 1-D Reibmodell	34
7.2	Plastisches Materialverhalten	35
7.2.1	Integrationsalgorithmus für ratenunabhängige Plastizität	35
7.2.2	Return-Mapping-Algorithmus	36
7.3	Klassische J_2 -Plastizität	40
7.3.1	Exakte Linearisierung des Algorithmus	43
7.4	Modellierung duktiler Schädigung	44
7.4.1	Phänomene von Schädigung in metallischen Werkstoffen	44
7.4.2	Kontinuumsschädigungsmodell	44
7.4.3	Konstitutivgleichungen nach GURSON und TVERGAARD	45
7.4.4	Numerische Umsetzung nach ARAVAS	47
7.5	Behandlung finiter Inelastizität	48
8	Grundlagen der Methode der finiten Elemente	53
8.1	Schwache Form — Prinzip der virtuellen Verschiebungen	53
8.2	Linearisierung und Diskretisierung	54
8.2.1	Typen finiter Elemente in der Strukturmechanik	56
8.3	Iteratives Vorgehen	58
8.3.1	Globales NEWTON-Verfahren	58
8.3.2	Behandlung großer, linearer Gleichungssysteme	59
8.3.3	Iterative Lösung des globalen Gleichungssystems	60
II	Anwendungen	63
9	Parameter-Identifikation	64
9.1	Beispiel: Hyperelastische Werkstoffe	64
9.1.1	Einaxiale Darstellung des neo-HOOKE-Modells	64
9.1.2	YEOH-Modell	64
9.2	Versuchsanordnungen	65
9.3	<i>Fehler-Quadrat-Minimierung</i>	66

10 Implementierung von Materialmodellen	67
10.1 Hyperelastische Modelle über UHYPER	67
10.1.1 Schnittstelle	67
10.1.2 Zwangsbedingung InkompRESSibilität	68
10.1.3 Aktivieren / Ansprechen in ABAQUS	68
10.2 Allgemeine Material-Schnittstelle UMAT	68
10.2.1 Schnittstelle	68
10.2.2 Bestimmung des Moduls \mathbf{D}	68
III Übungsaufgaben	71
11 Übungsaufgaben	72
11.1 Berechnung von Deformationsmaßen	72
11.1.1 Hauptachsenzerlegung für <i>simple shear</i>	72
11.2 Einaxiale Darstellung und Ableitung des YEOH-Modells	73
11.3 Parameter-Anpassung — <i>Fehler-Quadrat-Minimierung</i>	74
11.4 Anwendung von ABAQUS CAE	75
11.4.1 Simulation einer axialsymmetrischen Struktur	75
11.4.2 HERTZsche Pressung — Linienkontakt — $a = l$	76
11.4.3 Drei-Punkt-Biegung — Plastische Zone	77
11.5 Programmierbeispiel	78
11.5.1 YEOH in UHYPER	78
Anhang	78
A Fortran77-Beispiel	79
B Aufstellung einiger Linux-/Unix-Befehle	80
Literatur	80

Kapitel 1

Einleitung

Die „Methode der finiten Elemente“ (FEM) ist in der zweiten Hälfte des 20. Jhdt. entwickelt worden und hat seitdem parallel mit den Innovationsschüben im *Hardware*-Bereich der Computer-Technologie eine rasante Weiterentwicklung erfahren.

Die Möglichkeiten einer allgemeinen Betrachtungsweise verschiedenster mechanischer Problemstellungen auf der Grundlage moderner mathematischer Werkzeuge (Matrixoperationen, Variationsformulierung, ...) haben die FEM neben anderen numerischen Verfahren (Differenzen-Schemata, Randelement-Methode/BEM, ...) zu dem bedeutendsten computergestützten Berechnungsverfahren gemacht. Der aktuelle Stand der *Hard*- und *Software*-Technologie ermöglicht heute jedem Entwicklungsingenieur und Wissenschaftler an einem Einzelarbeitsplatz auch schon größere (Anfangs-) Randwertprobleme zu lösen, wo vor mehreren Jahren noch Großrechenanlagen vonnöten waren. Die Anbindung und Anpassung eines jeden mechanisch-mathematischen Simulationsmodells findet mit der Auswahl eines geeigneten Materialmodells über die Materialparameter statt.

Die Materialmodellierung stellt somit das Bindeglied zwischen Modell und praktischer Anwendung dar. Dabei müssen die Materialparameter in eindeutiger Weise durch Experimente und entsprechende Modellrechnungen bestimmbar sein.

Eine zunehmende Verfeinerung und Präzisierung der Materialmodellierung setzt ein tieferes Verständnis der Materialtheorie und der algorithmischen Umsetzung dieser Modelle voraus, um möglichst alle wesentlichen Effekte des Materialverhaltens berücksichtigen zu können. Das Anliegen dieser Vorlesung zielt genau in diese Richtung, wobei hier ein besonderes Augenmerk auf einer korrekten und effizienten numerischen Umsetzung der angesprochenen Materialmodelle liegt.

Kapitel 2

Struktur der Lehrveranstaltung

2.1 Ablauf & Bewertung

In diesem Jahr finden die Lehrveranstaltungen donnerstags 8:00 Uhr – 10:30 Uhr in Raum S1/03/272 statt. Folgende Termine sind vorgesehen: 21.04.2011, 05.05., 19.05., 27.05., 09.06., 16.06. und 30.06..

Während der Veranstaltungen werden 2–3 Übungsaufgaben ausgegeben, die anschließend zu bearbeiten sind. Eine entsprechend ausführliche Dokumentation der Ausarbeitung wird bis zum Beginn des kommenden Wintersemesters bewertet und ergibt – wenn gewünscht – die Note für den Kurs.

2.2 Zeitplan & Themen

Unter <http://www.baaserweb.de/TUDarmstadt/SoSe11> werden zusätzliche Informationen zur Verfügung gestellt.

Folgende Themen und Inhalte sind in diesem Jahr vorgesehen und können bei Bedarf / Wunsch modifiziert und erweitert werden:

- 21.04.2011
 - Begrüßung, Teilnehmer–Liste
 - Abklären der Erwartungen
 - Kurzvorstellung FEM
 - * Wie funktioniert FEM ? Was ist FEM ?
 - * Ablauf einer FEM–Analyse
 - Vorbereitung („preprocessing“)
 - (möglicherweise iterative) Lösung („solver“ / „solution“)

- Nachbereitung („postprocessing“)
 - * Wir behandeln Themen der Festkörper- bzw. Strukturmechanik.
 - * FEM *heute*: Nichtlinearitäten in (1.) Geometrie (große Deformationen, exakte Kinematik) und (2.) Kontakt- und sonstige Restriktionen (Koppelungen, Kinematiken, ...)
- Überblick ABAQUS-CAE
- Beispiel RWDR, siehe Abschn. 11.4.1.
- Kinematik großer Deformationen – siehe Abschn. 5.1–5.3
- 05.05.
 - Spannungskonzept, siehe Abschn. 5.4
 - Übungen mit/an ABAQUS-CAE
- 19.05.
 - Parameteranpassung
- 26.05.
 - Übung zu Parameteranpassung, siehe Kap. 9
- 09.06.
 - HERTZsche Pressung — Linienkontakt, siehe Abschn. 11.4.2
 - Vergleich mit Literaturwerten
- 16.06.
 - FEM als (Entwicklungs-)Werkzeug des Ingenieurs
 - * Kundenbeziehung
 - * Anfrage – Angebot – Auftrag – Bearbeitung – Dokumentation
 - Nachbearbeitung/-betrachtung – Controlling – Kommunikation
- 30.06.
 - FEM als Forschungsgegenstand, aktuelle Herausforderungen
 - * Elementformulierungen, *Locking*-Phänome
 - * Effizienz, Parallelisierung
 - * V&V, Aussagequalität, Fehlerquellen
 - * Materialmodellierung

Kapitel 3

Verwendung von ABAQUS

3.1 Login und Verzeichnisse

Nach dem Start der Rechner steht jeweils der „Account/Login“ `femint` zur Verfügung. Das zugehörige Passwort wird während der Veranstaltung genannt. Bitte Arbeitsverzeichnisse mit `mkdir [name]` der einzelnen Gruppen anlegen, dorthin wechseln mit `cd [name]` und ausschließlich dort arbeiten.

3.2 Struktur von ABAQUS

Aufruf Hilfe / Manual auf <http://lame:2080/>

3.2.1 Kurz-Tutorium am Beispiel eines O-Rings

Seit Anfang des Jahrtausends, in etwa zeitgleich mit den Versionen $> 6.x$ wird für ABAQUS auch die grafische Oberfläche ABAQUS/CAE angeboten, die sowohl als *Präprozessor* als auch zum *Postprocessing* eingesetzt werden kann. Dabei bedeutet *CAE* hier *Complete Abaqus Environment* und integriert damit auch den einzeln verfügbaren ABAQUS/VIEWER in einer einheitlichen Umgebung. Ebenfalls kann damit der Ablauf einzelner FEM-Rechnungen gesteuert und beobachtet werden, wenn diese auf dem gleichen Rechner ausgeführt werden. Wir betrachten hier die Anwendung der FE-Methode aus *klassischer* Sicht und damit auch Vorbereitung *Preprocessing*, Lösung *Solver* und Nachbereitung *Postprocessing* einer FE-Analyse als getrennte Aufgaben, die – jede für sich – besondere Herausforderungen aufweist.

Hier konzentrieren wir uns zunächst auf den Umgang mit ABAQUS/CAE und weisen auf einige Merkmale hin. Dies kann in keiner Weise eine Schulung

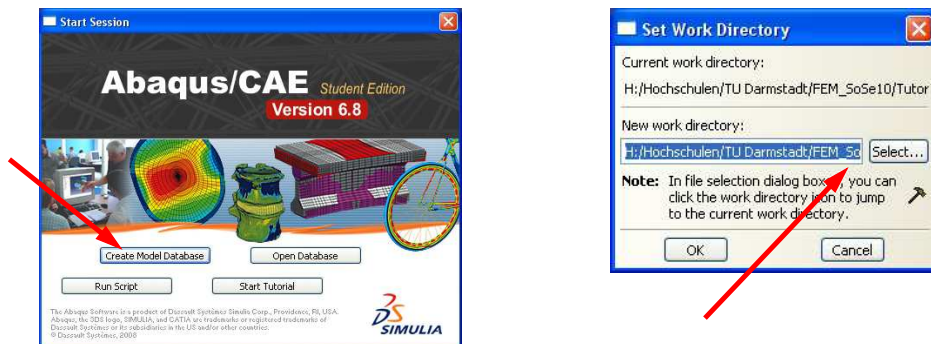


Abbildung 3.1: ABAQUS CAE-Eröffnung und anschließende Festlegung eines Arbeitsverzeichnis

mit diesem Programmsystem oder zumindest ein intensives Durcharbeiten eines Beispiels aus dem *Users Manual* ersetzen !

Nach dem Start von ABAQUS/CAE, siehe Abb. 3.1, empfiehlt es sich, direkt das aktuelle Arbeitsverzeichnis innerhalb des Dateisystems festzulegen, um dort alle Ein- und Ausgabedaten leicht (wieder) zu finden.

Das Erstellen eines FEM-Modells beginnt in dem Modul Part oder aber vorgeschaltet im Sketch-Modus. Beidesmal kann entweder eine CAD-Grafik importiert werden oder aber es können mit einfachsten CAD-Hilfsmitteln eigene Modell konstruiert werden. Man stellt also zunächst das Modul ein, siehe Abb. 3.2, und kann nun eine Konstruktion beginnen, einlesen oder erweitern.

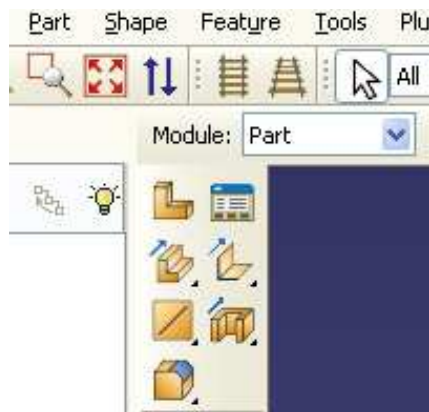


Abbildung 3.2: Arbeitsmodul auswählen

tionsbaums, durch den man sich von oben nach unten *durchhangeln* muss, siehe Abb. 3.3. Ein ähnliches Vorgehen ist schon seit mehreren Jahren z.B. von der Benutzeroberfläche von ANSYS (heute in der sog. *Workbench*) bekannt.

Im linken Fensterteil der CAE-Anwendung erkennt man das Grundgerüst des *Deklara-*

Wir beginnen für dieses einfache Beispiel nun mit der Defintion der Grundstrukturen unserer Modellierung indem wir die Einzelteile zeichnen. Ein O-Ring wird in der 2D-axialsymmetrischen Modellierung im Querschnitt als Kreis dargestellt. Dazu legen wir ein Part an, siehe Abb. 3.4, und zeichnen einen Kreis, bestenfalls schon an der richtigen Position im zunächst frei zu wählenden Koordinatensystem, siehe Abb. 3.5. Später muss die Rotationsachse des Modells bei $x = r = 0$ liegen.

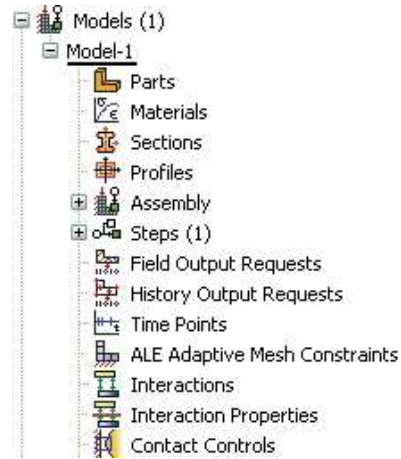


Abbildung 3.3: „Deklarationsbaum“

Einzelteile lassen sich auch nachträglich noch positionieren. Ziel soll sein, den Ring auf eine Welle „aufzuziehen“ und die Situation dann zu analysie-

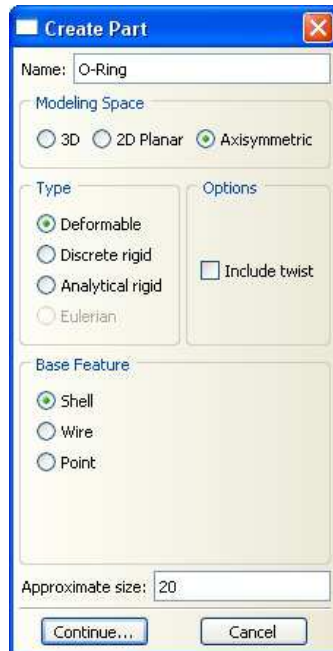


Abbildung 3.4: Einzelteile definieren: hier O-Ring

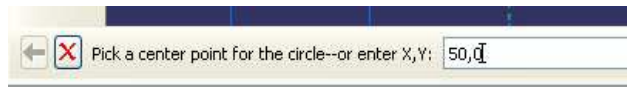


Abbildung 3.5: O-Ring-Querschnitt zeichnen



Abbildung 3.6: Einzelteil „Welle“

ren. Dazu legen wir nun zunächst die Welle als starre Gegenfläche (*analytical rigid*) an, siehe Abb. 3.6, und schieben diese später über die Definition von Verschiebungsrandbedingungen an einem daran gekoppelten Referenzknoten radial nach außen. Der nächste Schritt entlang der nötigen Deklarationen ist die Festlegung eines Materials inkl. Parameter für den Elastomer-Ring, was über das Feld in Abb. 3.7(a) möglich ist. Im Abschnitt **Section** wird einer elementierten **Part**-Struktur ein bereits ausgewähltes Material(modell) zugeordnet, siehe Abb. 3.7(b). Im weiteren Verlauf werden innerhalb der **Steps** die Randbedingungen und Belastungen deklariert, so dass damit der *Preprocessing*-Teil innerhalb ABAQUS/CAE abgeschlossen ist. Der Abschnitt **Jobs** stellt den FEM-Lösungsprozess dar, der ebenfalls recht elegant über den **Job Monitor** gestartet und verfolgt werden kann. Die Auswertung der Berechnungsergebnisse erfolgt im ABAQUS/VIEWER nach Einlesen

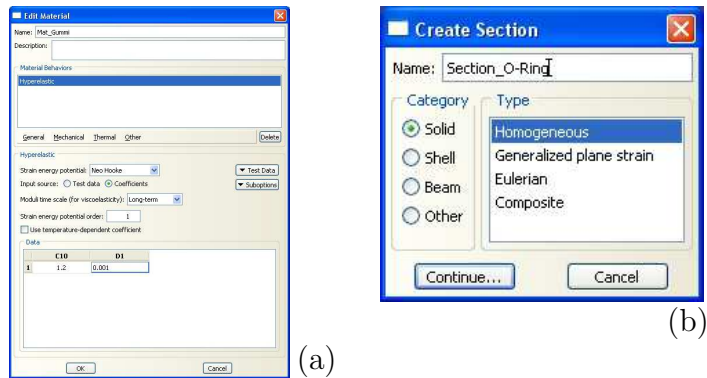


Abbildung 3.7: (a) Material definieren und dieses (b) einer Section zuordnen

der erzeugten Ausgabedatei *.odb, die alle Modellinformationen und Ergebnisse enthält. In Abb. 3.8 ist nur ein kleiner Ausschnitt der sehr leistungsfähigen Werkzeuge zum *Postprocessing* dargestellt. Auch der ABAQUS/VIEWER ist wieder durch eine Werkzeugleiste links und einen Grafikbildschirm aufgebaut.



Abbildung 3.8: Ergebnis–Auswertung im VIEWER: (Un–)verformte Struktur und Feldgrößen

3.2.2 Direkter Aufruf von der Konsole

Präprozessor ABAQUS CAE

Aufruf mit `abaqus cae &`

FEM-Löser ABAQUS

Aufruf mit `abaqus job=[jobname] interactive`

Aufruf und Einbinden einer F77-subroutine

Aufruf mit `abaqus job=[jobname] user=[routine].f interactive`

Postprozessor ABAQUS VIEWER

Aufruf mit `abaqus viewer &`

Teil I
Theorie — Grundlagen

Kapitel 4

Mathematische Grundlagen

Wir verwenden folgende Vektor- und Matrix-Operationen, teilweise in Indexnotation, wie Sie auch in modernen Textbüchern über nichtlineare Kontinuumsmechanik angegeben wird, siehe z.B. HOLZAPFEL [2000].

Die *Transponierte* eines Vektors oder einer tensoriellen Größe in Matrixdarstellung wird dargestellt durch $(\bullet)^T$, die *Inverse* einer Matrix oder eines Tensors durch $(\bullet)^{-1}$. Das *Punkt-Produkt*, das einen Indize zusammenzieht, $\mathbf{a} \cdot \mathbf{b}$ oder $\mathbf{A} \cdot \mathbf{B}$ geben wir gemäß der EINSTEINSchen Summationskonvention mit $a_k b_k$ oder $A_{ik} B_{kj}$ an.

Im gleichen Sinne wird die „Doppelkontraktion“ $\mathbf{A} : \mathbf{B}$ zweier Matrizen zu $A_{ij} B_{ij}$ verstanden. Das Vektor- oder Tensorprodukt $\mathbf{C} = \mathbf{a} \otimes \mathbf{b}$ wird demgemäß $C_{ij} = a_i b_j$ angegeben und führt auf eine Größe höherer Ordnung. Zusätzlich wird der *Spur-Operator* mit $\text{tr}(\mathbf{A}) = A_{ii} = A_{11} + A_{22} + A_{33}$ notiert und angegeben.

Solange es möglich ist und die Darstellung eindeutig, verwenden wir für Tensoren erster und zweiter Ordnung römische oder griechische Buchstaben in **Fettdruck**, während Tensoren vierter Ordnung mit z.B. \mathbb{D} oder $\mathbb{1}$ angegeben werden.

4.1 Skalare, Vektoren und Tensoren

1. Skalar („Tensor 0. Ordnung“)

Ein Skalar ist eine ungerichtete Größe zur Quantifizierung.

Bsp.: Druck p , Dichte ρ , Temperatur ϑ

2. Vektor („Tensor 1. Ordnung“)

Ein Vektor ist eine gerichtete Grösse, die durch Betrag, Richtung und Orientierung (=Richtungssinn) angegeben wird. Durch einen Vektor wird im EUKLIDischen, dreidimensionalen Punktraum (=physikalischer Raum, der uns umgibt) eindeutig ein Punkt (x_1, x_2, x_3) beschrieben:

$$\mathbf{x} := \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad \text{im } \mathbb{R}^3, \text{ „Komponentendarstellung als einspaltige Matrix“} \quad (4.1)$$

Wobei wir mit $\{\mathbf{e}_1 \times \mathbf{e}_2 \times \mathbf{e}_3\}$ das kartesische Koordinatensystem bezeichnen, das orthogonal und normiert (orthonormiert) ist. Also:

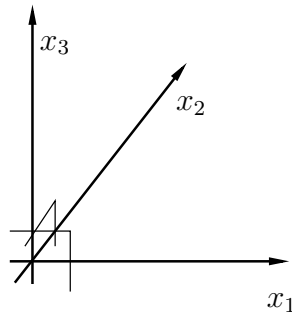


Abbildung 4.1: Kartesisches Koordinatensystem

$$\mathbf{x} = x_1 \mathbf{e}_1 + x_2 \mathbf{e}_2 + x_3 \mathbf{e}_3 = \sum_{i=1}^3 x_i \mathbf{e}_i =: x_i \mathbf{e}_i \quad (4.2)$$

EINSTEIN'sche Summationskonvention:

Summation über doppelt vorkommende lateinische Indizes von 1 bis 3, wenn nichts anderes gesagt wird.

Beachte somit:

$$\begin{aligned} (x_i y_i)^2 &= (x_1 y_1 + x_2 y_2 + x_3 y_3)^2 \\ x_i^2 y_i^2 &= x_1^2 y_1^2 + x_2^2 y_2^2 + x_3^2 y_3^2 \end{aligned} \quad (4.3)$$

Ordnet man sowohl Vektorkomponenten (x_i) als auch Basisvektoren (\mathbf{e}_i) in

Matrixform an, so lässt sich auch

$$\mathbf{x} = (x_i)^T(\mathbf{e}_i) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}^T \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{bmatrix} = [x_1, x_2, x_3] \begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \\ \mathbf{e}_3 \end{bmatrix} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3 \quad (4.4)$$

schreiben.

Zur Darstellung eines Vektors ist die Angabe der Basis notwendig!

- Matrizen sind im Rahmen dieses Kalküls also „nur“ Hilfsmittel zur Darstellung. Mit ihnen lassen sich die Komponentendarstellungen der Gleichungen numerisch auswerten.
- Sehr wohl vermischen sich die Darstellungsformen leicht durch eine Vernachlässigung der Angabe des Bezugssystems.
- Im allgemeinen sind folgende Angaben identisch, wenn auch aus oben genannten Gründen nicht ganz konsequent:

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3 = \dots \quad (4.5)$$

3. Tensoren höherer Ordnung

Ein Tensor n -ter Stufe ist ein n -fach lineares Funktional (=Operator, der jedem Element seines Definitionsbereiches eindeutig eine reelle Zahl zuordnet), dessen 3^n Komponenten $T_{i_1 \dots i_n} \in \mathbb{R}^3$ sich bei der Drehung oder Spiegelung des kartesischen Koordinatensystems nach

$$T_{i_1 \dots i_n} = c_{i_1 k_1} c_{i_2 k_2} \dots c_{i_n k_n} \bar{T}_{k_1 \dots k_n} \quad (4.6)$$

transformieren; mit $c_{ij} = \cos \angle(\mathbf{e}_i, \mathbf{e}_j)$ als Transformationskoeffizienten.

Darstellung von Tensoren in Analogie zu Vektoren (Bsp. für Tensor zweiter Stufe):

$$\begin{aligned} \mathbf{T} = \mathbf{a} \otimes \mathbf{b} &= (a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{e}_3) \otimes (b_1\mathbf{e}_1 + b_2\mathbf{e}_2 + b_3\mathbf{e}_3) \\ &= a_1b_1\mathbf{e}_1 \otimes \mathbf{e}_1 + \dots + a_1b_3\mathbf{e}_1 \otimes \mathbf{e}_3 \\ &+ a_2b_1\mathbf{e}_2 \otimes \mathbf{e}_1 + \dots + a_2b_3\mathbf{e}_2 \otimes \mathbf{e}_3 \\ &+ a_3b_1\mathbf{e}_3 \otimes \mathbf{e}_1 + \dots + a_3b_3\mathbf{e}_3 \otimes \mathbf{e}_3 \\ &= T_{ij} \mathbf{e}_i \otimes \mathbf{e}_j \end{aligned} \quad (4.7)$$

Die $3^2 = 9$ Komponenten eines Tensors 2. Stufe im \mathbb{R}^3 lassen sich als 3×3 -Matrix anordnen. So vermischt sich auch hier wieder die Darstellung:

$$(T_{ij}) = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} = \mathbf{T} \quad (4.8)$$

- Achtung: Dyadenprodukt $\mathbf{a} \otimes \mathbf{b} = \mathbf{ab} \neq \mathbf{a}^T \mathbf{b}$ Skalarprodukt
- Rechenregeln analog EINSTEIN'scher Summenkonvention: $a_{ij} = T_{ijkl} b_{kl}$

4.2 Hauptachsentransformation

4.3 Notation, Matrix-Darstellung

4.4 Polare Zerlegung

Siehe (5.5). Wichtig für Behandlung finiter Inelastizität in Abschnitt 7.5.

Kapitel 5

Elemente der Kontinuumsmechanik — Grundlagen der Festkörpermechanik

Umfassende Literaturquellen: ALTENBACH & ALTENBACH [1994] , BECKER & BÜRGER [1975] , HOLZAPFEL [2000], ITSKOV [2009]

5.1 Deformationsgradient und Verzerrungsmaße

Wir betrachten den Körper \mathcal{B} im Sinne der Kontinuumsmechanik als eine zusammenhängende Menge von Materie im dreidimensionalen Anschauungsraum \mathbb{E}^3 (EUKLIDISCHER Punktraum).

Die Lage eines materiellen Punktes im Raum zur Zeit t wird durch eine eindeutige Vektorzuordnung χ mit \mathbf{X} in der Ausgangskonfiguration und \mathbf{x} in der Momentankonfiguration bezeichnet:

$$\mathbf{x} = \chi(\mathbf{X}, t) \tag{5.1}$$

Der Verschiebungsvektor $\mathbf{u} = \mathbf{x} - \mathbf{X}$ beschreibt das zeitliche Aufeinanderfolgen von Konfigurationen, also die Bewegung. Mit dem Deformationsgradienten

$$\mathbf{F}(\mathbf{X}) = \text{Grad}(\chi) = \frac{\partial \chi}{\partial \mathbf{X}} = \mathbf{I} + \text{Grad } \mathbf{u}(\mathbf{X}) \tag{5.2}$$

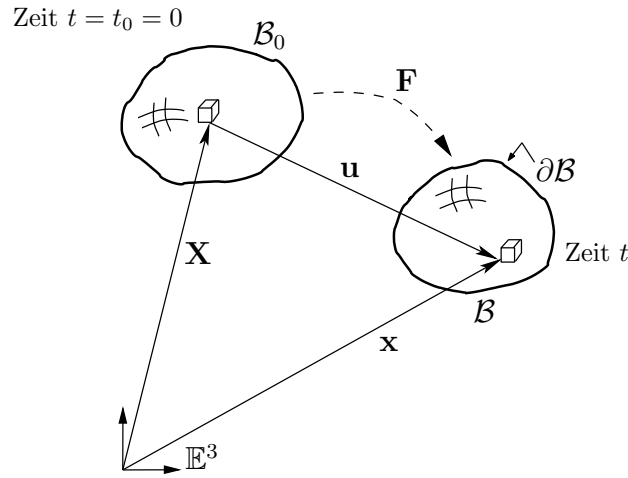


Abbildung 5.1: Ausgangs- und Momentankonfiguration im EUKLIDISCHEN Raum \mathbb{E}^3

wird der Zusammenhang zwischen den Konfigurationen hergestellt. Infinitesimale Linienelemente transformieren sich demnach durch:

$$d\mathbf{x} = \mathbf{F} \cdot d\mathbf{X} . \quad (5.3)$$

Physikalisch zulässige Deformationen sind durch

$$J := \det \mathbf{F} > 0 \quad \text{mit} \quad J = \frac{dv}{dV} = \frac{\varrho_0}{\varrho} \quad (5.4)$$

gekennzeichnet.

Mit dieser Eigenschaft $J \neq 0$ läßt sich \mathbf{F} eindeutig *polar* zerlegen in

$$\mathbf{F} = \mathbf{R} \cdot \mathbf{U} = \mathbf{v} \cdot \mathbf{R} \quad (5.5)$$

wobei \mathbf{U} und \mathbf{v} die Rechts- bzw. Linksstrecktensoren und \mathbf{R} ein eigentlich orthogonaler Drehtensor ($\mathbf{R}^T \cdot \mathbf{R} = \mathbf{I}$, $\det \mathbf{R} = 1$) darstellen.

Damit lassen sich sinnvolle Verzerrungsmaße einführen, siehe auch KAHN & HUANG [1995]

- GREEN-LAGRANGE-Verzerrungstensor

$$\mathbf{E} := \frac{1}{2}(\mathbf{C} - \mathbf{I}) \quad \text{mit} \quad \mathbf{C} = \mathbf{F}^T \cdot \mathbf{F} = \mathbf{U}^2 \quad (5.6)$$

mit Größen in der Ausgangskonfiguration (materielle Beschreibung). Dabei wird die Größe \mathbf{C} als *rechter CAUCHY-GREEN-Tensor* bezeichnet.

Dehnungen / Verzerrungen – 1D-Beispiel

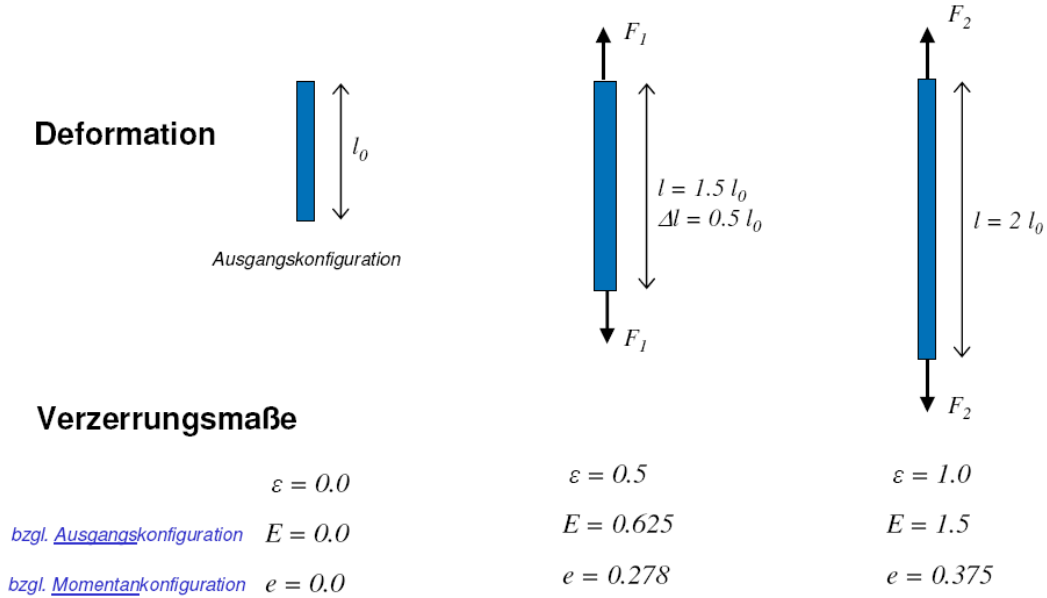


Abbildung 5.2: Vergleich unterschiedlicher Verzerrungsmaße für 1ax-Zug

- ALMANSI-Verzerrungstensor

$$\mathbf{e} := \frac{1}{2}(\mathbf{I} - \mathbf{b}^{-1}) \quad \text{mit } \mathbf{b} = \mathbf{F} \cdot \mathbf{F}^T = \mathbf{v}^2 \quad (5.7)$$

mit Größen in der aktuellen Konfiguration (räumliche Darstellung) und \mathbf{b} als *linker CAUCHY-GREEN-Tensor*.

Nachrechnen liefert einen Zusammenhang zwischen \mathbf{E} und \mathbf{e} :

$$\mathbf{e} = \mathbf{F}^{-T} \cdot \mathbf{E} \cdot \mathbf{F}^{-1} \quad (5.8)$$

Dies entspricht der Vorwärtstransformation (push-forward) des kovarianten Tensors \mathbf{E} in die Momentankonfiguration.

5.1.1 Beispiele

Die beiden Verzerrungsmaße \mathbf{E} und \mathbf{e} aus (5.6) und (5.7) sind für den einaxialen Fall in Abb. 5.2 berechnet und angegeben. An Abb. 5.3 kann man sich — ebenfalls für den einaxialen Fall skizziert — sehr gut die „Definition“ der (Haupt-)Streckungen $\lambda_i = \frac{l_i}{L_i}$ (mit $i = 1, 2, 3$) veranschaulichen. Für diesen

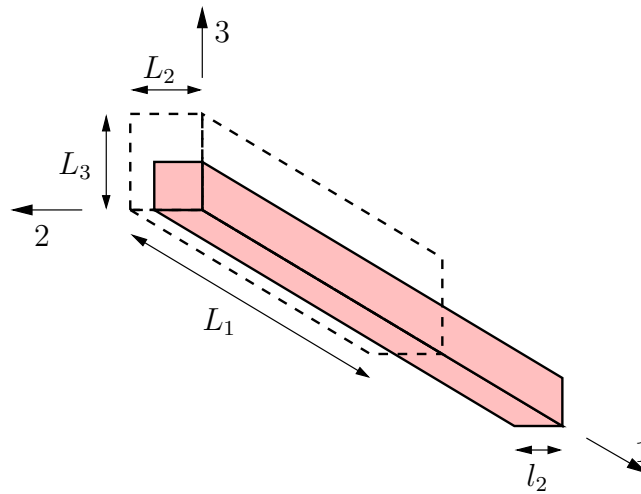


Abbildung 5.3: Deformationsmode „Einaxialer Zug“ (Prototyp 1)

Fall (Prototyp-Deformation 1) stellt sich der Deformationsgradient als

$$\mathbf{F}_{1ax} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \quad (5.9)$$

dar. Dies stellt demnach eine Deformation im Hauptachsensystem dar, welches sich währenddessen nicht verändert. Eine weitere Prototyp-Deformation ist durch den Fall in Abb. 5.4 gegeben. Dies wird durch den Deformations-

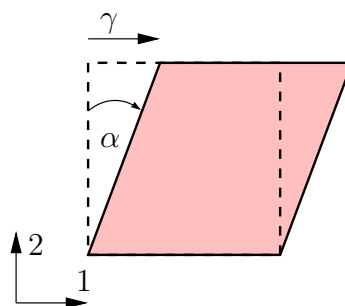


Abbildung 5.4: Deformationsmode „Einfache Scherung“ (*Simple Shear* – Prototyp 4)

gradient

$$\mathbf{F}_{Sish} = \begin{bmatrix} 1 & \gamma & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \text{mit } \gamma = \tan \alpha \quad (5.10)$$

repräsentiert.

5.2 Verzerrungsraten

Räumliches Geschwindigkeitsfeld

$$\mathbf{v} = \dot{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial t} \quad (5.11)$$

und dessen Gradient

$$\text{grad } \mathbf{v} = \frac{\partial \mathbf{v}}{\partial \mathbf{x}} =: \mathbf{l} \quad (5.12)$$

Daraus:

Symmetrischer Anteil des räumlichen Geschwindigkeitsgradienten

$$\mathbf{d} := \frac{1}{2}(\mathbf{l} + \mathbf{l}^T) \quad (5.13)$$

Schiefsymmetrischer Anteil des räumlichen Geschwindigkeitsgradienten

$$\mathbf{w} := \frac{1}{2}(\mathbf{l} - \mathbf{l}^T) \quad (5.14)$$

5.3 Deformationsinvarianten

Aus den *Eigenwertproblemen* für \mathbf{C} und \mathbf{b}

$$\det[\mathbf{C} - \mu \mathbf{I}] \stackrel{!}{=} 0 \quad \text{bzw.} \quad \det[\mathbf{b} - \mu \mathbf{I}] \stackrel{!}{=} 0 \quad (5.15)$$

ergeben sich über das *charakteristische Polynom*

$$\mu^3 - I_1 \mu^2 + I_2 \mu - I_3 = 0 \quad (5.16)$$

die drei Eigenwerte $\mu_{1,2,3}$ von \mathbf{C} und \mathbf{b} , die sich als die Quadrate der Hauptstreckungen $\lambda_{1,2,3}$ aus Abschn. 5.1.1 darstellen. Damit gilt $\mu_{1,2,3} = \lambda_{1,2,3}^2$.

Die Koeffizienten I_1, I_2 und I_3 aus (5.16) stellen die sog. (*Deformations*)-*Invarianten* dar und sind bei gegebener Deformation – wie der Name schon sagt – *invariant* gegenüber Rotationen (des Koordinatensystems und damit

Beobachters). Damit kommt ihnen bei der Formulierung von Materialgesetzen eine besondere Bedeutung zu, siehe Abschn. 6.2. Sie ergeben sich aus (5.15) bzw. (5.16) zu

$$I_1 := \text{Spur}(\mathbf{C}) = \text{Spur}(\mathbf{b}) = \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \quad (5.17)$$

$$I_2 := \lambda_1^2 \lambda_2^2 + \lambda_1^2 \lambda_3^2 + \lambda_2^2 \lambda_3^2 \quad (5.18)$$

$$I_3 := \det(\mathbf{C}) = \det(\mathbf{b}) = \lambda_1^2 \lambda_2^2 \lambda_3^2 = J^2. \quad (5.19)$$

Für den Fall *idealer Inkompressibilität* des Werkstoffs gilt $I_3 \equiv 1$, siehe (5.19)₃. Damit lassen sich die Zusammenhänge in (5.19) sehr schön veranschaulichen, siehe BAASER [2010b] und dazu Abb. 5.5, die auch schon in TRELOAR [1975] zu finden ist.

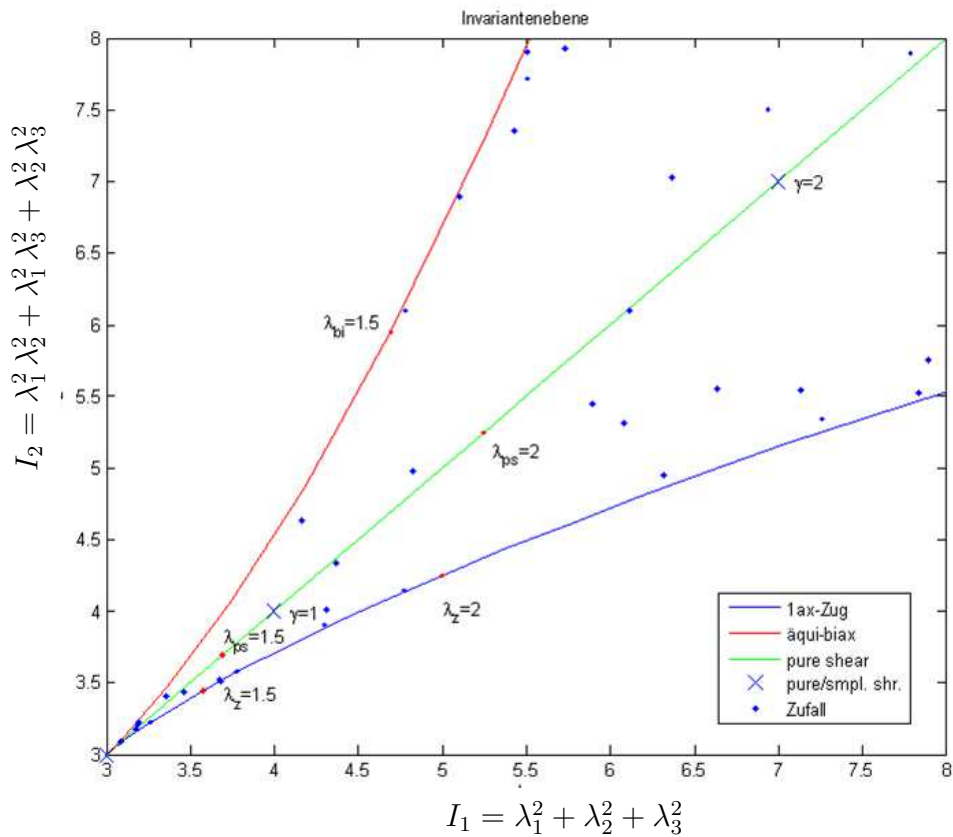


Abbildung 5.5: Invarianten-Ebene für inkompressibles Werkstoffverhalten

5.4 Mechanische Spannung — Konjugierte Spannungstensoren

Aus Sicht der Festkörpermechanik mag ein Körper \mathcal{B} belastet sein durch *äußere Kräfte* \mathbf{b} , die als *Volumenkräfte* auf *jeden Partikel* des Körper wirken, und durch *Oberflächenlasten* \mathbf{t} . Beide Arten bewirken *innere Kräfte* \mathbf{F}_{int} im Körper durch Interaktion der Partikel im Volumen. Schneiden wir den Körper \mathcal{B} gedanklich entlang $S - S$ auf, sprechen wir von *flächenbezogenen Kräften* oder *Spannungen*, siehe Abb. 5.6, und definieren den *Spannungsvektor*

$$\mathbf{t} = \lim_{\Delta a \rightarrow 0} \frac{\mathbf{F}_{int}}{\Delta a} \quad (5.20)$$

im Grenzübergang bzgl. dem Flächenelement Δa . Dabei beschreibt $\mathbf{n}_{\Delta a}$ die nach außen gerichtete Flächennormale.

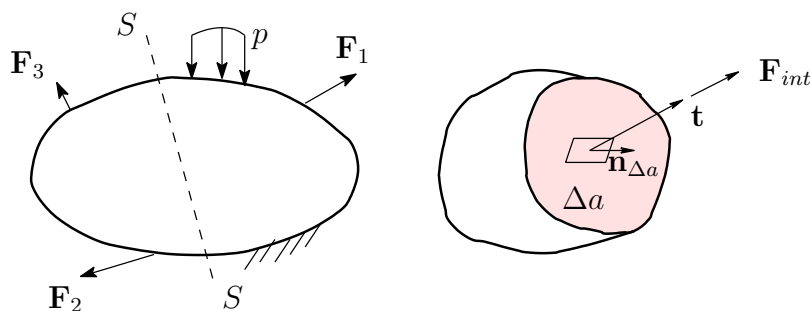


Abbildung 5.6: Beliebiger Schnitt $S - S$ durch einen belasteten Körper \mathcal{B}

Gleichzeitig können wir den Spannungsvektor \mathbf{t} nach dem sog. *CAUCHY-Theorem*

$$\mathbf{t} = \boldsymbol{\sigma} \cdot \mathbf{n}_{\Delta a} \quad (5.21)$$

auch durch den *Spannungstensor* (Tensor 2. Stufe) $\boldsymbol{\sigma}$ in jedem beliebigen Punkte des Körpers unter der Schnittebene $\mathbf{n}_{\Delta a}$ darstellen.

Im Rahmen dieser Betrachtung beschränken wir uns auf eine symmetrische Darstellung des CAUCHYSchen Spannungstensors $\boldsymbol{\sigma}$, so dass wir mit sechs unterschiedlichen Komponenten der ursprünglich $3^2 = 9$ Koeffizienten des Tensors 2. Stufe im \mathbb{E}^3 arbeiten.

5.4.1 VOIGT-Notation

Gemäß der sog. VOIGT-Notation werden die Koeffizienten der (symmetrischen !) Spannungstensoren in einem Spaltenvektor abgelegt, um Speicherplatz und Rechenoperationen bei z. B. Multiplikation dieser Tensoren mit

anderen Größen einzusparen. In vielen Programmsystemen hat sich eine Darstellung der Form

$$\check{\boldsymbol{\sigma}} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix} \quad (5.22)$$

etabliert.

Anmerkung 1:

Bei dieser Darstellung ist darauf zu achten, dass insbesondere die Einträge 5 und 6 in $\check{\boldsymbol{\sigma}}$ in der dem Programmsystem entsprechenden Reihenfolge angegeben werden. Hier gibt es Unterschiede z. B. in FEAP, DAEDALON, ABAQUS/STANDARD, ABAQUS/EXPLICIT, ANSYS oder MSC.MARC usw. Jeweilige Handbücher beachten !

Anmerkung 2:

Die sechs Koeffizienten der symmetrischen Dehnungstensoren $\boldsymbol{\varepsilon}$, beispielsweise als linearisierte Form von \mathbf{E} und \mathbf{e} , können in analoger Weise abgelegt werden:

$$\check{\boldsymbol{\varepsilon}} = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2 \varepsilon_{12} \\ 2 \varepsilon_{13} \\ 2 \varepsilon_{23} \end{bmatrix}, \quad (5.23)$$

wobei hier oftmals in den *Schubtermen* zusätzlich der Faktor 2 mitgeführt wird. Dies hat den Vorteil, dass ein Energieausdruck der Form $W^* = \boldsymbol{\sigma} \cdot \boldsymbol{\varepsilon}$ durch eine reine *Vektormultiplikation* $\check{\boldsymbol{\sigma}} \cdot \check{\boldsymbol{\varepsilon}}$ sofort erhalten wird.

Anmerkung 3:

Aus den hier angegebenen Reihenfolgen der Anordnung der Tensorkoeffizienten ergibt sich dann ebenfalls in konsequenter Erweiterung das Schema für die Speicherung der Koeffizienten von Tensoren 4. Stufe (z. B. des Materialmoduls \mathbf{D} in $\check{\mathbf{D}}$) als 6×6 -Matrix. Dadurch ergibt sich beispielsweise ein „Materialgesetz“ $\boldsymbol{\sigma} = \mathbf{D} : \boldsymbol{\varepsilon}$ als *einfache Matrix-Vektor-Multiplikation* der Form $\check{\boldsymbol{\sigma}} = \check{\mathbf{D}} \cdot \check{\boldsymbol{\varepsilon}}$.

5.4.2 Spannungsleistung & konjugierte Spannungstensoren

Die Rate der inneren, mechanischen Arbeit („Spannungsleistung“) ist gegeben durch

$$\mathcal{P}_{\text{int}} = \int_V \mathbf{P} : \dot{\mathbf{F}} \, dV \quad (5.24)$$

oder durch

$$\begin{aligned} \mathcal{P}_{\text{int}} = \dot{w} &= \int_{\mathcal{B}} \boldsymbol{\sigma} : \mathbf{d} \, dv = \int_{\mathcal{B}} \left(\frac{1}{J} \mathbf{P} \cdot \mathbf{F}^T \right) : (\mathbf{F}^{-T} \cdot \dot{\mathbf{E}} \cdot \mathbf{F}^{-1}) \, dv = \\ &= \int_{\mathcal{B}} \frac{1}{J} (\mathbf{F} \cdot \mathbf{S}) : (\dot{\mathbf{E}} \cdot \mathbf{F}^{-1}) \, dv = \int_{\mathcal{B}_0} \mathbf{S} : \dot{\mathbf{E}} \, dV \end{aligned} \quad (5.25)$$

- \mathbf{S} und \mathbf{E} sind ebenfalls konjugierte, symmetrische Größen
- Analog lassen sich aus dieser Herleitung weitere konjugierte Spannungs- und Verzerrungsmaße angeben, z.B. $\mathbf{P} \leftrightarrow \dot{\mathbf{F}}$ aus (5.24)

Zusammenstellung:

2. PIOLA-KIRCHHOFF-Spg.tensor	$\mathbf{S} = J\mathbf{F}^{-1} \cdot \boldsymbol{\sigma} \cdot \mathbf{F}^{-T}$	Ausgangskonfig.
KIRCHHOFF-Spg.tensor	$\boldsymbol{\tau} = J\boldsymbol{\sigma} = \mathbf{F} \cdot \mathbf{S} \cdot \mathbf{F}^T$	Momentankonfig.
1. PIOLA-KIRCHHOFF-Spg.tensor	$\mathbf{P} = J\boldsymbol{\sigma} \cdot \mathbf{F}^{-T} = \boldsymbol{\tau} \cdot \mathbf{F}^{-T}$	2-Feldtensor

(5.26)

Mit $\mathbf{P} \cdot \mathbf{F}^T = \mathbf{F} \cdot \mathbf{P}^T = J\boldsymbol{\sigma} = \boldsymbol{\tau} = \mathbf{F} \cdot \mathbf{S} \cdot \mathbf{F}^T$ und $\dot{\mathbf{F}} = \mathbf{l} \cdot \mathbf{F}$ können wir daraus auch äquivalente *spannungsleistungskonjugierte* Paare zusammenstellen.

$$\dot{\mathbf{F}} \longleftrightarrow \mathbf{P} \quad (5.27)$$

$$\dot{\mathbf{E}} \longleftrightarrow \mathbf{S} \quad (5.28)$$

$$J\boldsymbol{\sigma} = \boldsymbol{\tau} \longleftrightarrow \mathbf{d} = \mathbf{F}^{-T} \cdot \dot{\mathbf{E}} \cdot \mathbf{F}^{-1} = \mathbf{R} \cdot [\dot{\mathbf{U}} \cdot \mathbf{U}^{-1}]_{\text{sym}} \cdot \mathbf{R}^T \quad (5.29)$$

5.4.3 Vorgehen in FEM-Programmsystemen

Für das weitere Vorgehen wird im Hinblick auf die Materialschnittstelle noch ein zusätzliches konjugiertes Paar benötigt:

relativer oder rotierter Spannungstensor $\bar{\boldsymbol{\tau}} = \mathbf{R}^T \cdot \boldsymbol{\tau} \cdot \mathbf{R} \leftrightarrow$

logarithmische HENCKY-Verzerrungen und deren Rate $\mathbf{R}^T \cdot \mathbf{d} \cdot \mathbf{R}$

$$\begin{aligned} \mathbf{R}^T \cdot \mathbf{d} \cdot \mathbf{R} &= \frac{1}{2} \mathbf{R}^T \cdot \left([\dot{\mathbf{R}} \cdot \mathbf{U} + \mathbf{R} \cdot \dot{\mathbf{U}}] \cdot \mathbf{U}^{-1} \cdot \mathbf{R}^{-1} + \mathbf{R}^{-T} \cdot \mathbf{U}^{-T} \cdot [\mathbf{U}^T \cdot \dot{\mathbf{R}}^T + \dot{\mathbf{U}}^T \cdot \mathbf{R}^T] \right) \cdot \mathbf{R} \\ &= \frac{1}{2} \left(\mathbf{R}^T \cdot \dot{\mathbf{R}} + \dot{\mathbf{U}} \cdot \mathbf{U}^{-1} + \dot{\mathbf{R}}^T \cdot \mathbf{R} + \mathbf{U}^{-T} \cdot \dot{\mathbf{U}}^T \right) = \frac{1}{2} \left(\dot{\mathbf{U}} \cdot \mathbf{U}^{-1} + \mathbf{U}^{-T} \cdot \dot{\mathbf{U}}^T \right) \end{aligned} \quad (5.30)$$

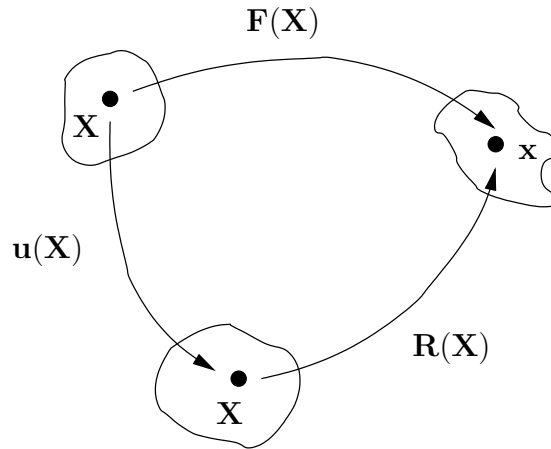


Abbildung 5.7: Polare Zerlegung

Wie kommt man nun zu logarithmischen Dehnraten?

Der Geschwindigkeitsgradient \mathbf{l} bildet $d\mathbf{x}$ auf seine materielle Zeitableitung ab:

$$\frac{D}{Dt}d\mathbf{x} = \dot{\mathbf{F}} \cdot d\mathbf{X} = \mathbf{l} \cdot d\mathbf{x} \quad (5.31)$$

Die Länge von $d\mathbf{x}$ sei $ds = \sqrt{d\mathbf{x}^T d\mathbf{x}}$ und die Zeitableitung davon

$$\frac{D}{Dt}ds = \frac{1}{2ds} \frac{D}{Dt}(d\mathbf{x}^T d\mathbf{x}) = \frac{1}{2ds} (d\mathbf{x}^T \cdot \mathbf{l}^T \cdot d\mathbf{x} + d\mathbf{x}^T \cdot \mathbf{l} \cdot d\mathbf{x}) = \frac{1}{ds} d\mathbf{x}^T \cdot \mathbf{d} \cdot d\mathbf{x} \quad (5.32)$$

mit $d\mathbf{x} = \mathbf{n}ds$ (\mathbf{n} -Einheitsvektor)

$$\frac{D}{Dt}ds = \frac{1}{ds} (\mathbf{n}^T \cdot \mathbf{d} \cdot \mathbf{n}) ds^2 = \mathbf{n}^T \cdot \mathbf{d} \cdot \mathbf{n} ds \quad (5.33)$$

$$\text{vgl. DGL: } \dot{x} = Kx \quad \rightarrow \quad x = e^{Kt} \quad \rightarrow \quad \ln x = Kt \quad \rightarrow \quad \frac{D}{Dt} \ln x = K \quad (5.34)$$

$$\Rightarrow \frac{D}{Dt} \ln(ds) = \mathbf{n}^T \cdot \mathbf{d} \cdot \mathbf{n} \quad (5.35)$$

Daraus ergibt sich, dass $d_{(ii)}$ logarithmische Dehnraten sind.

Konjugiertes Paar: $\bar{\boldsymbol{\tau}}$ und $\ln \mathbf{U} = \ln \mathbf{v}$

5.4.4 Spannungsraten

JAUMANNsche Spannungsrate

$$\frac{d^\nabla}{dt}(J\boldsymbol{\sigma}) = \frac{d}{dt}(J\boldsymbol{\sigma}) - J(\mathbf{w} \cdot \boldsymbol{\sigma} - \boldsymbol{\sigma} \cdot \mathbf{w}) \quad (5.36)$$

mit \mathbf{w} aus (5.14).

5.5 Verknüpfung von Spannung und Dehnung — Materialmodul

Die Verknüpfung der in Abschnitt 5.4 dargestellten Spannungsmaßen und den in Abschnitt 5.1 gezeigten kinematischen Größen wird durch ein *Materialmodell* realisiert, das in diesem Rahmen durch den 4-stufigen Tensor \mathbb{C} oder \mathbb{D} repräsentiert werden soll.

Es kann hier *nicht* auf die *mechanische Materialtheorie* eingegangen werden, die gewisse Restriktionen an diese Modell vorgibt, so dass es *thermomechanisch konsistenten* Relationen kommt.

Zunächst soll hier nur angemerkt sein, dass mit einem solchen Zusammenhang die Verhältnisse in (5.25) nicht gestört werden dürfen. Wir fokussieren in den beiden folgenden Kapiteln 6 und 7 auf die numerische Umsetzung einiger, beispielhafter Materialformulierungen.

In einer einfachen und hier auch ratenunabhängigen Form geben wir beispielsweise den Zusammenhang zwischen 2. PIOLA–KIRCHHOFFSchem Spannungs- und GREEN–LAGRANGESchem Verzerrungstensor zu

$$\mathbf{S} = \mathbb{C} : \mathbf{E} \quad \longleftrightarrow \quad S_{ij} = \mathbb{C}_{ijkl} E_{kl} \quad (5.37)$$

an.

Kapitel 6

Elastisches Materialverhalten

6.1 Linear–elastisches Materialverhalten — HOOKEsches Gesetz

Im Rahmen der linearen Elastizitätstheorie stellt sich bei rein *isotropem* Materialverhalten die Verknüpfung von Spannungstensor (5.22) und Verzerrungstensor (5.23) in der Form

$$\boldsymbol{\sigma} = \mathbf{C} : \boldsymbol{\varepsilon} \quad (6.1)$$

dar, wobei der Steifigkeitstensor \mathbf{C} in diesem Fall aus nur zwei unabhängigen Materialparametern angegeben werden kann. Diese sind die LAMESchen Konstanten λ und μ , die wiederum – alternativ – mit dem Elastizitätsmodul (YOUNG–Modul) E und der Querkontraktionszahl (POISSON–Zahl) ν im Verhältnis stehen:

$$E = \frac{\mu(2\mu + 3\lambda)}{\mu + \lambda} \quad (6.2)$$

$$\nu = \frac{\lambda}{2(\mu + \lambda)} \quad \text{und} \quad (6.3)$$

$$\mu = G \quad \text{als Schubmodul.} \quad (6.4)$$

Eine recht elegante und für die numerische Umsetzung effektive Form lässt sich schreiben, wenn man in (6.1)

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{11}^* & \mathbf{C}_{12}^* \\ \mathbf{C}_{21}^* & \mathbf{C}_{22}^* \end{bmatrix} \quad (6.5)$$

als Blockmatrix notiert. Damit ergeben sich die Diagonaleinträge des Spannungstensors aus \mathbf{C}_{11}^* zu

$$\begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \end{bmatrix} = \underbrace{\begin{bmatrix} \lambda + 2\mu & \lambda & \lambda \\ \lambda & \lambda + 2\mu & \lambda \\ \lambda & \lambda & \lambda + 2\mu \end{bmatrix}}_{\mathbf{C}_{11}^*} \cdot \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \end{bmatrix} \quad (6.6)$$

und die Neben-Diagonaleinträge analog aus \mathbf{C}_{22}^* zu

$$\begin{bmatrix} \sigma_{12} \\ \sigma_{13} \\ \sigma_{23} \end{bmatrix} = \underbrace{\begin{bmatrix} \mu & 0 & 0 \\ 0 & \mu & 0 \\ 0 & 0 & \mu \end{bmatrix}}_{\mathbf{C}_{22}^*} \cdot \begin{bmatrix} 2\varepsilon_{12} \\ 2\varepsilon_{13} \\ 2\varepsilon_{23} \end{bmatrix}, \quad (6.7)$$

wobei $\mathbf{C}_{12}^* = \mathbf{C}_{21}^* \equiv \mathbf{0}$ sind – wohlgemerkt unter obiger Annahme der Isotropie.

6.2 Hyperelastizität

Materialverhalten wird als *CAUCHY-elastic* bezeichnet, wenn sich der *CAUCHY-Spannungstensor* allein als Funktion der aktuellen Deformation gemäß

$$\boldsymbol{\sigma} = \mathcal{F}(\mathbf{b}) = \alpha_0 \mathbf{I} + \alpha_1 \mathbf{b} + \alpha_2 \mathbf{b}^2 \quad (6.8)$$

darstellen lässt. Dabei entspricht (6.8)₂ dem Ergebnis des *RIVLIN-ERICKSEN-Theorems* (1955) mit $\alpha_{0,1,2} = \alpha_{0,1,2}[I_1(\mathbf{b}), I_2(\mathbf{b}), I_3(\mathbf{b})]$ für isotropes Verhalten.

Wir beschränken uns wiederum auf die Beschreibung *zeitunabhängiger* und *isothermer* Prozesse. Mehr Details zu diesem ganzen Komplex finden sich in *HOLZAPFEL* [2000] in ausgezeichneter Form dargestellt.

Als Spezialfall zu oben nennen wir Materialverhalten *hyperelastisch* oder *GREEN-elastic*, bei dem eine *freie HELMHOLTZ-Energie(dichte)funktion* (pro Einheitsmasse und Einheitsvolumen) als *Potential* U für die Spannungen darstellen lässt. Damit folgt beispielsweise

$$\mathbf{P} = \frac{\partial U(\mathbf{F})}{\partial \mathbf{F}} \quad (6.9)$$

oder

$$\mathbf{S} = \frac{\partial U(\mathbf{E})}{\partial \mathbf{E}} = 2 \frac{\partial U(\mathbf{C})}{\partial \mathbf{C}} \quad (6.10)$$

$$= 2 \left[\left(\frac{\partial U}{\partial I_1} + I_1 \frac{\partial U}{\partial I_2} \right) \mathbf{I} - \frac{\partial U}{\partial I_2} \mathbf{C} + I_3 \frac{\partial U}{\partial I_3} \mathbf{C}^{-1} \right] \quad (6.11)$$

mit beispielsweise

$$\frac{\partial I_1}{\partial \mathbf{C}} = \frac{\partial \text{Spur}(\mathbf{C})}{\partial \mathbf{C}} = \frac{\partial (\mathbf{I} : \mathbf{C})}{\partial \mathbf{C}} = \mathbf{I} \quad (6.12)$$

$$\frac{\partial I_2}{\partial \mathbf{C}} = I_1 \mathbf{I} - \mathbf{C} \quad (6.13)$$

$$\frac{\partial I_3}{\partial \mathbf{C}} = I_3 \mathbf{C}^{-1} . \quad (6.14)$$

6.2.1 Quasi–inkompressible Darstellung

Mit Hinblick auf eine Implementierung und Umsetzung hyperelastischer Materialmodelle im Rahmen der FE–Methode formuliert man diese Modelle in einer Darstellung, in der die *volumetrischen* und *isochoren* (*gestaltändernden*) Anteile additiv getrennt werden. Insbesondere für Materialien, die (nahezu) inkompressibel (Beispiel technische Elastomere, Gummi: Kompressionsmodul K um drei Größenordnungen höher als der Schubmodul G) sind, kann man dadurch mit speziellen FE–Formulierungen die *Zwangsbedingung* $\det \mathbf{F} = J \simeq 1$ bis zu einer Genauigkeit von G/K erreichen und konvergente Lösungen bekommen.

Grundlegend für diese Darstellung ist die Definition eines modifizierten Deformationsgradienten

$$\bar{\mathbf{F}} := J^{-1/3} \mathbf{F} , \quad (6.15)$$

womit $\det \bar{\mathbf{F}} \equiv 1$ erfüllt ist und beispielsweise $\bar{\mathbf{C}} := J^{-2/3} \mathbf{C}$ folgt.

Mit der Annahme der *additiven Zerlegung* der Verzerrungsenergie U in

$$U(\mathbf{C}) = U_{iso}(\bar{\mathbf{C}}) + U_{vol}(J) = U_{iso}[\bar{I}_1(\bar{\mathbf{C}}), \bar{I}_2(\bar{\mathbf{C}})] + U_{vol}(J) \quad (6.16)$$

kommt man aus (6.11) zu einer äquivalenten Darstellung für den CAUCHY–Spannungstensor

$$\boldsymbol{\sigma} = 2J^{-1} \mathbf{b} \frac{\partial U(\mathbf{b})}{\partial \mathbf{b}} \quad (6.17)$$

$$= \boldsymbol{\sigma}_{iso} + \boldsymbol{\sigma}_{vol} = 2J^{-1} \mathbf{b} \frac{\partial U_{iso}(\bar{\mathbf{b}})}{\partial \bar{\mathbf{b}}} + p \mathbf{I} \quad (6.18)$$

in der der hydrostatische Druck $p = -\frac{1}{3} \text{Spur}(\boldsymbol{\sigma}) = -\frac{1}{3}(\sigma_{11} + \sigma_{22} + \sigma_{33})$ als volumetrischer Anteil abgespalten ist.

Anmerkung:

Für die Herleitung von (6.18) aus (6.11) sind u.a. folgende Beziehungen hilfreich:

$$\frac{\partial J}{\partial \mathbf{C}} = \frac{J}{2} \mathbf{C}^{-1} \quad \text{und} \quad \frac{\partial J^{-2/3}}{\partial \mathbf{C}} = -\frac{1}{3} J^{-2/3} \mathbf{C}^{-1} . \quad (6.19)$$

One of the simplest hyperelastic material models — often used for simple rubber representation and similar material behavior like soft tissues in bio-mechanical applications — is given by a *free energy density function* of the typical neo-HOOKEian type form

$$U_{NH} = \underbrace{\frac{1}{2}\mu(\bar{I}_1 - 3)}_{U_{NH\,iso}} + \underbrace{\frac{1}{2}K(J - 1)^2}_{U_{NH\,vol}}, \quad (6.20)$$

where we already split U_{NH} off into a purely isochoric part $U_{NH\,iso}$ and a purely volumetric part $U_{NH\,vol}$. Hereby, the isochoric part is driven by \bar{I}_1 , the first invariant of the modified right CAUCHY-GREEN deformation tensor $\bar{\mathbf{C}} = J^{-2/3}\mathbf{C}$, see Sec. ??, and the volumetric part is just given as function of the third invariant of the deformation gradient $J = \det(\mathbf{F}) = \sqrt{I_3(\mathbf{C})}$, respectively. Here, nonlinearity is given by the nonlinear deformation measure respecting for finite strains.

In addition, the function U_{NH} is determined by only two free material parameters $\mu = G$ and K , which can be clearly identified as *shear modulus* and *compression modulus*, respectively, due to the additive representation in (6.20).

For the simulation of e.g. rubber materials, which are known to react with no volumetric deformation, the part $U_{NH\,vol}$ can be seen as incompressibility constraint. In such cases the compression modulus K penalizes the material response and is in the order of $K \cong 2000G$ for typical rubber materials.

From the representation (6.20) the stress response in terms of the second PIOLA-KIRCHHOFF stress is given by

$$\mathbf{S} = 2 \frac{\partial U_{NH}(\mathbf{C})}{\partial \mathbf{C}} = \mathbf{S}_{iso} + \mathbf{S}_{vol} = 2 \left[\frac{\partial U_{NH\,iso}(\bar{\mathbf{C}})}{\partial \mathbf{C}} + \frac{\partial U_{NH\,vol}(J)}{\partial \mathbf{C}} \right]. \quad (6.21)$$

The belonging modulus can be obtained after some costly manipulations by

$$\mathbb{C} = 2 \frac{\partial \mathbf{S}(\mathbf{C})}{\partial \mathbf{C}} = \mathbb{C}_{iso} + \mathbb{C}_{vol} = 2 \frac{\partial \mathbf{S}_{iso}}{\partial \mathbf{C}} + 2 \frac{\partial \mathbf{S}_{vol}}{\partial \mathbf{C}} \quad (6.22)$$

with

$$\mathbb{C}_{iso} = -\frac{2}{3} J^{-2/3} \mu \left[(\mathbf{C}^{-1} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{C}^{-1}) - \frac{1}{3} \text{tr}(\mathbf{C}) \mathbf{C}^{-1} \otimes \mathbf{C}^{-1} + \text{tr}(\mathbf{C}) \frac{\partial \mathbf{C}^{-1}}{\partial \mathbf{C}} \right]$$

and

$$\mathbb{C}_{vol} = KJ \left[(2J - 1) \mathbf{C}^{-1} \otimes \mathbf{C}^{-1} + 2(J - 1) \frac{\partial \mathbf{C}^{-1}}{\partial \mathbf{C}} \right]. \quad (6.23)$$

Here, again the decoupled form as in (6.20) is visible, which leads to a representation in mainly two term depending on the parameters G and K and the deformation due to \mathbf{C} and $J = \det(\mathbf{F})$.

For more details, we refer to HOLZAPFEL [2000] and to MIEHE [1994], especially for a transformation onto the actual configuration. This representation of a fourth order tensor can be treated as 6×6 array as given before, where our preliminaries of the VOIGT notation are applied, see Sec. ??, and — at last — it remains symmetric because of the nature of the constitutive framework. A more enhanced model for rubber elasticity is the well-known YEOH model with

$$U_{YeoH} = c_1 (\bar{I}_1 - 3) + c_2 (\bar{I}_1 - 3)^2 + c_3 (\bar{I}_1 - 3)^3 + \frac{1}{2} K (J - 1)^2 \quad (6.24)$$

as strain energy density function incorporating three material parameters c_1, c_2 and c_3 . This model is able to represent the typical upturn behavior in rubber materials for elongated stretches.

Kapitel 7

Inelastisches Materialverhalten

7.1 Motivation. 1-D Reibmodell

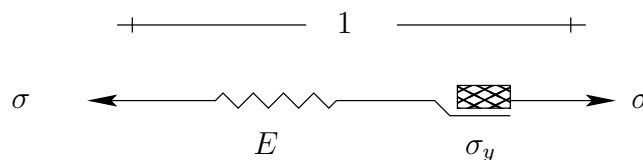


Abbildung 7.1: Reibelement

Gesamtdehnung: $\varepsilon = \varepsilon^e + \varepsilon^p$

Gleichgewichtsbetrachtung: $\sigma = E\varepsilon^e \equiv E(\varepsilon - \varepsilon^p)$

Betrachtung des Reiblements:

Maximal mögliche Spannung ist dort durch σ_y begrenzt (Fließbedingung):

$$f(\sigma) := |\sigma| - \sigma_y \leq 0 \quad (7.1)$$

Fallunterscheidung:

1. $|\sigma| < \sigma_y \rightsquigarrow \dot{\varepsilon}^p = 0$ keine Änderung ($\dot{\cdot}$) möglich!

Also: $\dot{\varepsilon}^p = 0$, wenn $f(\sigma) = |\sigma| - \sigma_y < 0$

Rein elastische Antwort. $f(\sigma < 0)$ ist der elastische Bereich

2. $|\sigma| = \sigma_y \rightsquigarrow \dot{\varepsilon}^p \neq 0$ Verschiebung des Reiblements

$$\begin{aligned} \dot{\varepsilon}^p &= \gamma > 0 & , \text{wenn } \sigma &= \sigma_y > 0 \\ \dot{\varepsilon}^p &= -\gamma > 0 & , \text{wenn } \sigma &= -\sigma_y < 0 \end{aligned} \quad (7.2)$$

$$\rightsquigarrow \dot{\varepsilon}^p = \gamma \cdot \text{sign}(\sigma) \quad , \text{wenn } f(\sigma) = |\sigma| - \sigma_y = 0$$

zusammenfassend:

$$\left. \begin{array}{l} f(\sigma) < 0 \Rightarrow \gamma = 0 \\ \gamma > 0 \Rightarrow f(\sigma) = 0 \end{array} \right\} \longleftrightarrow \gamma \cdot f(\sigma) = 0 \quad \text{KUHN-TUCKER-Bedingungen} \quad (7.3)$$

alternativ:

$$\left. \begin{array}{l} \dot{f}(\sigma) < 0 \Rightarrow \gamma = 0 \\ \gamma > 0 \Rightarrow \dot{f}(\sigma) = 0 \end{array} \right\} \longleftrightarrow \gamma \cdot \dot{f}(\sigma) = 0 \quad \text{Konsistenz-Bedingung} \quad (7.4)$$

7.2 Plastisches Materialverhalten

7.2.1 Integrationsalgorithmus für ratenunabhängige Plastizität

„dehnungsgesteuerter Prozess im Zeitintervall $[t_n, t_n + \Delta t]$ “
allgemeines Anfangswertproblem:

$$\left. \begin{array}{l} \dot{x}(t) = f(x(t)) \\ x(0) = x_n \end{array} \right\} \quad (7.5)$$

Zeitintegration:

$$\left. \begin{array}{l} x_{n+1} = x_n + \Delta t \cdot f(x_{n+\theta}) \\ x_{n+\theta} = \theta \cdot x_{n+1} + (1 - \theta)x_n \end{array} \right\} \quad \text{mit } \theta \in [0, 1] \quad (7.6)$$

Dabei ist $x_{n+1} = x(t_{n+1})$ und $t_{n+1} = t_n + \Delta t$ und

$$\left. \begin{array}{l} \theta = 0 \longrightarrow \text{EULER vorwärts („explizit“)} \\ \theta = \frac{1}{2} \longrightarrow \text{Mittelpunktsregel} \\ \theta = 1 \longrightarrow \text{EULER rückwärts („implizit“)} \end{array} \right\} \quad (7.7)$$

Siehe entsprechende mathematische Handbücher zur Genauigkeit und Stabilität der Integrationsregeln.

Wie wählen $\theta = 1$ und erhalten für vorangegangenes 1-D Beispiel:

$$\left. \begin{array}{l} \varepsilon_{n+1}^p = \varepsilon_n^p + \Delta\gamma \cdot \text{sign}(\sigma_{n+1}) \\ \alpha_{n+1} = \alpha_n + \Delta\gamma \end{array} \right\} \quad (7.8)$$

Hierbei ist α eine zusätzliche „innere Voraussetzung“ zur Beschreibung isotroper Verfestigung ($\sigma_y = \sigma_y(\alpha)$). $\Delta\gamma = \gamma_{n+1} \cdot \Delta t \geq 0$ ist hierbei ein

LAGRANGE-Parameter und algorithmisches Geegstück zum Konsistenz-Parameter $\gamma \geq 0$.

$$\left. \begin{aligned} \sigma_{n+1} &= E(\varepsilon_{n+1}^p - \varepsilon_n^p) \\ \varepsilon_{n+1} &= \varepsilon_n + \Delta\varepsilon_n \end{aligned} \right\} \quad (7.9)$$

Die Variablen $(\sigma_{n+1}, \alpha_{n+1})$ und $\Delta\gamma$ sind begrenzt/eingeschränkt durch die „diskreten“ KUHN-TUCKER-Bedingungen:

$$\left. \begin{aligned} f_{n+1} &:= |\sigma_{n+1}| - (\sigma_y + K \cdot \alpha_{n+1}) \leq 0 \\ \Delta\gamma &\geq 0 \\ \Delta\gamma f_{n+1} &= 0 \end{aligned} \right\} \quad K \text{ ist der Verfestigungsmodul} \quad (7.10)$$

7.2.2 Return-Mapping-Algorithmus

Für einen beliebigen Zustand (σ, \mathbf{q}) sind folgende Situationen möglich:

$$\left\{ \begin{array}{l} f < 0 \Leftrightarrow (\sigma, \mathbf{q}) \in \text{int}(\mathbb{E}) \Rightarrow \gamma = 0 \text{ „elastisch“} \\ f = 0 \Leftrightarrow (\sigma, \mathbf{q}) \in \partial\mathbb{E} \left\{ \begin{array}{l} \dot{f} < 0 \Rightarrow \gamma = 0 \text{ „elastische Entlastung“} \\ \dot{f} = 0 \text{ und } \gamma = 0 \text{ „neutrale Belastung“} \\ \dot{f} < 0 \text{ und } \gamma = 0 \text{ „plastische Belastung“} \end{array} \right. \end{array} \right. \quad (7.11)$$

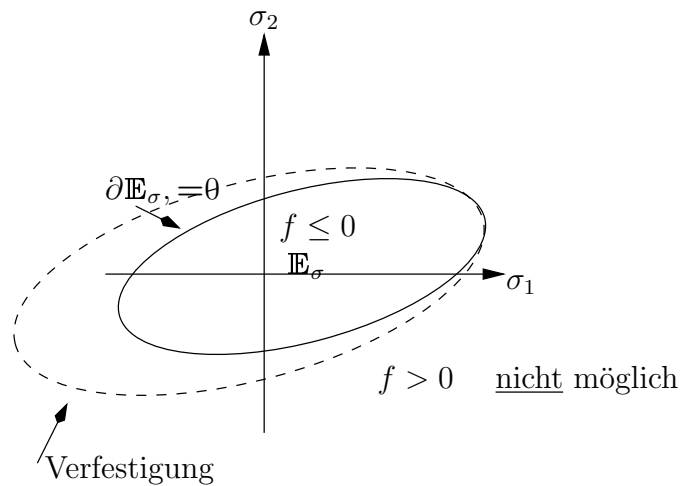


Abbildung 7.2: Verfestigung

Elastische „trial“-Schritt (=„Prädiktor“)

→ rein elastischer Hilfszustand, der möglicherweise physikalisch nie eingenommen wird, bei eingefrorenen inneren Variablen:

$$\left. \begin{aligned} \sigma_{n+1}^{\text{trial}} &:= E(\varepsilon_{n+1} - \varepsilon_n^p) \equiv \sigma_n + E \cdot \Delta\varepsilon_n \\ \varepsilon_{n+1}^{p\text{trial}} &:= \varepsilon_n^p \\ \alpha_{n+1}^{\text{trial}} &:= \alpha_n \\ f_{n+1}^{\text{trial}} &:= |\sigma_{n+1}^{\text{trial}}| - (\sigma_y + K\alpha_n) \end{aligned} \right\} \quad (7.12)$$

mit bekannten Anfangswerten $\{\varepsilon_n, \varepsilon_n^p, \alpha_n\}$ und Inkrement $\Delta\varepsilon_n$
Fallunterscheidung:

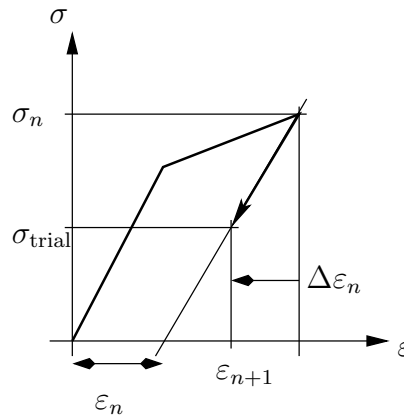


Abbildung 7.3: Elastische Antwort

- $f_{n+1}^{\text{trial}} \leq 0$ (elastische Antwort)

$$\rightsquigarrow \varepsilon_{n+1}^p := \varepsilon_n^p, \quad \alpha_{n+1} := \alpha_n, \quad \sigma_{n+1} := \sigma_{n+1}^{\text{trial}} \quad (7.13)$$

- Spannungs–Dehnungs–Zusammenhang
- Fließregel und Verfestigungsgesetz mit $\Delta\gamma \equiv 0$
- KUHN–TUCKER–Bedingung mit $f_{n+1} = f_{n+1}^{\text{trial}} \leq 0$ erfüllt

- $f_{n+1}^{\text{trial}} > 0$ (Verletzen der Fließbedingung) $f(\sigma, \alpha) \leq 0$

$$\rightsquigarrow \Delta\gamma \stackrel{!}{>} 0, \quad \varepsilon_{n+1}^p \neq \varepsilon_n^p \rightsquigarrow \sigma_{n+1} \neq \sigma_{n+1}^{\text{trial}} \quad (7.14)$$

$\Delta\gamma \stackrel{!}{>} 0$ ist das eigentliche Fließkriterium. $f = 0$ ist notwendig, aber nicht hinreichend!

Return-Mapping (=„Korrektor“)

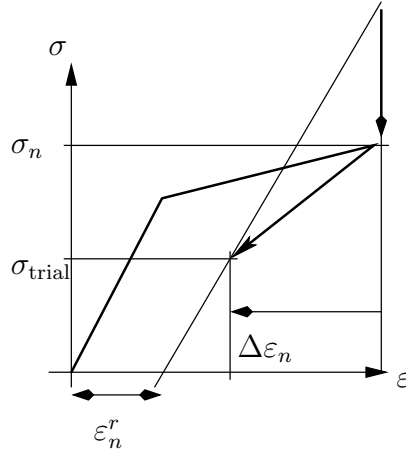


Abbildung 7.4: „Return Mapping“

es gilt:

$$\begin{aligned}
 \sigma_{n+1} &= E(\varepsilon_{n+1} - \varepsilon_{n+1}^p) \\
 &= E(\varepsilon_{n+1} - \varepsilon_n^p) - E(\varepsilon_{n+1}^p - \varepsilon_{n+1}^p) \\
 &= \sigma_{n+1}^{\text{trial}} - E \cdot \Delta\gamma \cdot \text{sign}(\sigma_{n+1}) \\
 \varepsilon_{n+1}^p &= \varepsilon_n^p + \Delta\gamma \cdot \text{sign}(\sigma_{n+1}) \\
 \alpha_{n+1} &= \alpha_n + \Delta\gamma \\
 f_{n+1} &\equiv |\sigma_{n+1}| - (\sigma_y + K\alpha_n)
 \end{aligned} \tag{7.15}$$

Bestimmung von $\Delta\gamma$:

aus (i):

$$\begin{aligned}
 |\sigma_{n+1}| \text{sign}(\sigma_{n+1}) &= |\sigma_{n+1}^{\text{trial}}| \text{sign}(\sigma_{n+1}^{\text{trial}}) - \Delta\gamma \cdot E \cdot \text{sign}(\sigma_{n+1}) \\
 (|\sigma_{n+1}| + \Delta\gamma \cdot E) \text{sign}(\sigma_{n+1}) &= |\sigma_{n+1}^{\text{trial}}| \text{sign}(\sigma_{n+1}^{\text{trial}})
 \end{aligned} \tag{7.16}$$

da $\Delta\gamma > 0$ und $E > 0$ folgt daraus:

$$\text{sign}(\sigma_{n+1}) = \text{sign}(\sigma_{n+1}^{\text{trial}}) \tag{7.17}$$

und

$$|\sigma_{n+1}| + \Delta\gamma \cdot E \text{sign}(\sigma_{n+1}) = |\sigma_{n+1}^{\text{trial}}| \tag{7.18}$$

(iii) in (ii):

$$\begin{aligned}
 f_{n+1} &= |\sigma_{n+1}^{\text{trial}}| - \Delta\gamma \cdot E - (\sigma_y + K\alpha_n) - K(\alpha_{n+1} - \alpha_n) \\
 &= f_{n+1}^{\text{trial}} - \Delta\gamma(E + K) \\
 &\stackrel{!}{=} 0
 \end{aligned} \tag{7.19}$$

$$\Rightarrow \Delta\gamma = \frac{f_{n+1}^{\text{trial}}}{E + K} > 0 \tag{7.20}$$

\Rightarrow insgesamt

$$\begin{aligned}
 \sigma_{n+1} &= \sigma_{n+1}^{\text{trial}} - \Delta\gamma \cdot E \cdot \text{sign}(\sigma_{n+1}^{\text{trial}}) \\
 \varepsilon_{n+1}^p &= \varepsilon_n^p + \Delta\gamma \cdot E \text{sign}(\sigma_{n+1}^{\text{trial}}) \\
 \alpha_{n+1} &= \alpha_n + \Delta\gamma
 \end{aligned} \tag{7.21}$$

Der zugehörige „algorithmische“ Tangentenmodul

1-D, isotrope Verfestigung

$$\mathbf{C}_{n+1} = \frac{\partial \sigma_{n+1}}{\partial \varepsilon_{n+1}} \tag{7.22}$$

auf Element-/Integrationspunktebene für optimale Konvergenz des NEWTON-Verfahrens

\leadsto „Ableiten des Algorithmus (*)“ ($\varepsilon_n^p, \alpha_n$ sind konstant!)

a)

$$\frac{\partial \sigma_{n+1}^{\text{trial}}}{\partial \varepsilon_{n+1}} = E \tag{7.23}$$

b)

$$\frac{\partial \Delta\gamma}{\partial \varepsilon_{n+1}} = \frac{1}{E + K} \cdot \frac{\partial f_{n+1}^{\text{trial}}}{\partial \varepsilon_{n+1}} = \frac{E}{E + K} \cdot \text{sign}(\sigma_{n+1}^{\text{trial}}) \tag{7.24}$$

c)

$$\begin{aligned}
 \sigma_{n+1} &= (|\sigma_{n+1}^{\text{trial}}| - \Delta\gamma \cdot E) \text{sign}(\sigma_{n+1}^{\text{trial}}) \\
 &= \left[1 - \frac{\Delta\gamma \cdot E}{|\sigma_{n+1}^{\text{trial}}|} \right] \cdot \sigma_{n+1}^{\text{trial}}
 \end{aligned} \tag{7.25}$$

Aus c) wird gebildet

$$\begin{aligned}
\frac{\partial \sigma_{n+1}}{\partial \varepsilon_{n+1}} &= \left[-\frac{E^2}{E+K} \frac{\text{sign}(\sigma_{n+1}^{\text{trial}}) |\sigma_{n+1}^{\text{trial}}| - \Delta\gamma E^2 \text{sign}(\sigma_{n+1}^{\text{trial}})}{|\sigma_{n+1}^{\text{trial}}|^2} \right] \cdot \sigma_{n+1}^{\text{trial}} + \left[1 - \frac{\Delta\gamma E}{|\sigma_{n+1}^{\text{trial}}|} \right] \cdot E \\
&= \frac{E^2}{E+K} \frac{|\sigma_{n+1}^{\text{trial}}| + \Delta\gamma E^2}{|\sigma_{n+1}^{\text{trial}}|^2} \cdot |\sigma_{n+1}^{\text{trial}}| + \left[1 - \frac{\Delta\gamma E}{|\sigma_{n+1}^{\text{trial}}|} \right] \cdot E \\
&= \frac{E \cdot K}{E + K} \quad \text{für } f_{n+1}^{\text{trial}} > 0
\end{aligned} \tag{7.26}$$

Dieser „algorithmische 1-D Tangentenmodul“ ist exakt derselbe wie der entsprechende elastoplastische Modul. Für höhere Dimensionen trifft dieses nicht mehr zu, siehe SIMO & HUGHES [1998].

7.3 Klassische J_2 -Plastizität

Grundproblem dehnungsgesteuert:

gegeben zum Zeitpunkt t_n :

$$\{\varepsilon_n, \varepsilon_n^p\} \tag{7.27}$$

mit abhängigen Größen $\{\varepsilon_n^e, \sigma_n\}$, die immer aus (7.27) bestimmt werden können mit:

$$\varepsilon_n^e := \varepsilon_n - \varepsilon_n^p \quad \text{und} \quad \sigma_n = \nabla W(\varepsilon_n^e) \tag{7.28}$$

Hierbei besteht das Ziel die Gleichung (7.27) zum Zeitpunkt t_{n+1} mit Hilfe der elastoplastischen Konstitutivgleichungen zu aktualisieren.

$$\begin{aligned}
\dot{\varepsilon}^p &= \gamma \mathbf{r}(\boldsymbol{\sigma}) && \text{Anm: „assoziert“ für } \mathbf{r} = \frac{\partial f(\boldsymbol{\sigma})}{\partial \boldsymbol{\sigma}} \\
\boldsymbol{\alpha} &= \boldsymbol{\alpha}(\gamma) && \text{allgemeiner: Satz innerer Variablen} \\
f(\boldsymbol{\sigma}) &\leq 0 && \tag{7.29}
\end{aligned}$$

und $\gamma \geq 0$ KUHN-TUCKER-Bedingung

$$\gamma f(\boldsymbol{\sigma}) = 0$$

und $\{\varepsilon, \varepsilon^p\}|_{t=t_n} = \{\varepsilon_n, \varepsilon_n^p\}$ Anfangsbedingungen

Damit ergibt sich in deviatorischen Größen:

$$f(\boldsymbol{\sigma}) = \|\text{dev} \boldsymbol{\sigma}\| - \sqrt{\frac{2}{3}} \sigma_y(\alpha) \quad \dot{\varepsilon}^p = \gamma \frac{\text{dev} \boldsymbol{\sigma}}{\|\text{dev} \boldsymbol{\sigma}\|} \quad \dot{\alpha} = \gamma \frac{2}{3} \tag{7.30}$$

Dies impliziert für $\|\dot{\boldsymbol{\epsilon}}^p\| = \gamma$:

$$\alpha(t) := \int_0^t \frac{2}{3} \|\dot{\boldsymbol{\epsilon}}^p(\tau)\| d\tau \quad \text{akkumuliert plastische Vergleichsdehnung} \quad (7.31)$$

Hierbei ist $\sigma_Y(\alpha) := \sigma_0 + K\alpha$ die sogenannte "lineare Verfestigung" und K der Verfestigungsmodul.

Algorithmische Umsetzung mit der VON MISES–Fließbedingung

Sinnvollerweise wird dieser Algorithmus in deviatorischen Größen formuliert. Integration im Intervall $[t_n, t_{n+1}]$. Wir führen noch $\mathbf{n}_{n+1} := \frac{\mathbf{s}_{n+1}}{\|\mathbf{s}_{n+1}\|}$ als *normierten Spannungsdeviator* ein. Für einen zweidimensionalen Spannungszustand in der Hauptachsendarstellung lässt sich \mathbf{n} als Vektor normal zur VON MISES–Fließfläche zeichnen, siehe Abb. 7.5. Damit lässt sich auch motivieren, dass $\mathbf{s}_{n+1}^{\text{trial}}$ *kolinear* mit \mathbf{s}_{n+1} ist.

Somit haben wir:

$$\begin{aligned} \boldsymbol{\epsilon}_{n+1}^p &= \boldsymbol{\epsilon}_n^p + \Delta\gamma \mathbf{n}_{n+1} \\ \alpha_{n+1} &= \alpha_n + \frac{2}{3} \Delta\gamma \\ \mathbf{s}_{n+1} &= \mathbf{s}_{n+1}^{\text{trial}} - 2\mu \Delta\gamma \mathbf{n}_{n+1} \end{aligned} \quad (7.32)$$

$$\mathbf{n}_{n+1} = \frac{\mathbf{s}_{n+1}}{\|\mathbf{s}_{n+1}\|} \stackrel{!}{=} \frac{\mathbf{s}_{n+1}^{\text{trial}}}{\|\mathbf{s}_{n+1}^{\text{trial}}\|} \quad (7.33)$$

Bestimmung von $\Delta\gamma$: $(*) \cdot \mathbf{n}_{n+1}$ ergibt

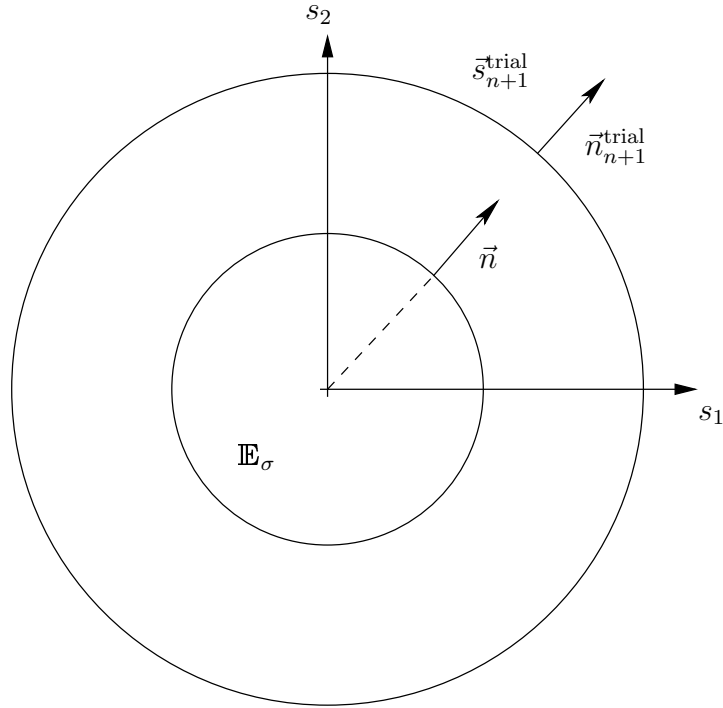


Abbildung 7.5: Anschauung 2D: \mathbf{s} und \mathbf{n} als Vektoren im Deviatorraum. Die von Mises-Fließfläche stellt hier einen Kreis dar.

$$\mathbf{s}_{n+1} \cdot \frac{\mathbf{s}_{n+1}^{\text{trial}}}{\|\mathbf{s}_{n+1}^{\text{trial}}\|} = \frac{\mathbf{s}_{n+1}^{\text{trial}} \cdot \mathbf{s}_{n+1}^{\text{trial}}}{\|\mathbf{s}_{n+1}^{\text{trial}}\|} - 2\mu\Delta\gamma \frac{\mathbf{s}_{n+1}^{\text{trial}} \cdot \mathbf{s}_{n+1}^{\text{trial}}}{\|\mathbf{s}_{n+1}^{\text{trial}}\|^2}$$

$$\frac{\|\mathbf{s}_{n+1}\| \frac{\mathbf{s}_{n+1}^{\text{trial}}}{\|\mathbf{s}_{n+1}^{\text{trial}}\|} \cdot \mathbf{s}_{n+1}^{\text{trial}}}{\|\mathbf{s}_{n+1}^{\text{trial}}\|} = \|\mathbf{s}_{n+1}^{\text{trial}}\| - 2\mu\Delta\gamma$$

$$\|\mathbf{s}_{n+1}\| = \|\mathbf{s}_{n+1}^{\text{trial}}\| - 2\mu\Delta\gamma$$

$$\underbrace{\|\mathbf{s}_{n+1}\| - \sqrt{\frac{2}{3}}\sigma_y(\alpha)}_{\triangleq f=0} = \|\mathbf{s}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}\sigma_y(\alpha) - 2\mu\Delta\gamma$$

$$\Rightarrow \|\mathbf{s}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}\sigma_y(\alpha) - 2\mu\Delta\gamma \stackrel{!}{=} 0 = g(\Delta\gamma) \quad \text{algorithmische Konsistenzbedingung}$$

$$\text{und } \alpha_{n+1} = \alpha_n + \frac{2}{3}\Delta\gamma$$

Diese Bestimmungsgleichung für $\Delta\gamma$ ist im allgemeinen nicht linear, aufgrund ihrer Konvexität aber garantiert mit einem NEWTON-Verfahren lösbar.

Spezialfall: $\sigma_y(\alpha) = \sigma_0 + K\alpha$ „linear“

$$\begin{aligned}
g(\Delta\gamma) = 0 &= \|\mathbf{s}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}(\sigma_0 + K\alpha + \sqrt{\frac{2}{3}}K\Delta\gamma) - 2\mu\Delta\gamma \\
0 &= f_{n+1}^{\text{trial}} - \frac{2}{3}K\Delta\gamma - 2\mu\Delta\gamma \\
\Rightarrow \Delta\gamma &= \frac{f_{n+1}^{\text{trial}}}{2\mu + \frac{2}{3}K} \quad 2\mu\Delta\gamma = \frac{f_{n+1}^{\text{trial}}}{1 + \frac{K}{3\mu}}
\end{aligned} \tag{7.35}$$

NEWTON-Verfahren zur Bestimmung von $\Delta\gamma$:

1. Initialisierung

$$\Delta\gamma^{(0)} = 0 \quad \alpha_{n+1}^{(0)} = \alpha_n \tag{7.36}$$

2. Iteration bis $|g(\Delta\gamma^{(k)})| < \text{TOL}$

$$\begin{aligned}
g(\Delta\gamma^{(k)}) &= \|\mathbf{s}_{n+1}^{\text{trial}}\| - \sqrt{\frac{2}{3}}\sigma_y(\alpha_{n+1}^{(k)}) - 2\mu\Delta\gamma^{(k)} \\
Dg(\Delta\gamma^{(k)}) &= -2\mu - \sqrt{\frac{2}{3}} \frac{\partial\sigma_y[\alpha_{n+1}^{(k)}(\Delta\gamma)]}{\partial\Delta\gamma} \\
\Rightarrow \Delta\gamma^{(k+1)} &= \Delta\gamma^{(k)} - \frac{g(\Delta\gamma^{(k)})}{Dg(\Delta\gamma^{(k)})} \\
\rightsquigarrow \alpha_{n+1}^{(k+1)} &= \alpha_n + \sqrt{\frac{2}{3}}\Delta\gamma^{(k+1)}
\end{aligned} \tag{7.37}$$

7.3.1 Exakte Linearisierung des Algorithmus

Für die algorithmische Linearisierung des Spannungstensors gilt:

$$\boldsymbol{\sigma}_{n+1} = \underbrace{\kappa(\text{tr}[\boldsymbol{\varepsilon}_{n+1}])\mathbf{I}}_{\text{hydrostatischer Anteil}} + \underbrace{2\mu(\text{dev}[\boldsymbol{\varepsilon}_{n+1}] - \Delta\gamma\mathbf{n}_{n+1})}_{\text{deviatorischer Anteil}} \tag{7.38}$$

mit dem Elastizitätstensor

$$\begin{aligned}
\mathbb{C} &= \kappa\mathbf{I} \otimes \mathbf{I} + 2\mu(\mathbb{1} - \frac{1}{3}\mathbf{I} \otimes \mathbf{I}) \\
C_{ijkl} &= \kappa\delta_{ij}\delta_{kl} + 2\mu(\frac{1}{2}\{\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}\} - \frac{1}{3}\delta_{ij}\delta_{kl}) \\
&= \lambda\delta_{ij}\delta_{kl} + \mu(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk})
\end{aligned} \tag{7.39}$$

ergibt sich durch Bildung des Differentials:

$$\begin{aligned}
 d\boldsymbol{\sigma}_{n+1} &= \mathbb{C} : d\boldsymbol{\varepsilon} - 2\mu[d\Delta\gamma\mathbf{n}_{n+1} + \Delta\gamma d\mathbf{n}_{n+1}] \\
 &= \left\{ \mathbb{C} - 2\mu\mathbf{n}_{n+1} \otimes \frac{\partial\Delta\gamma}{\partial\boldsymbol{\varepsilon}_{n+1}} - 2\mu\Delta\gamma \frac{\partial\mathbf{n}_{n+1}}{\partial\boldsymbol{\varepsilon}_{n+1}} : d\boldsymbol{\varepsilon}_{n+1} \right\}
 \end{aligned} \tag{7.40}$$

7.4 Modellierung duktiler Schädigung

- GURSON [1977] → TVERGAARD [1989]
- ROUSSELIER *et al.* [1989]
- LEMAITRE & CHABOCHE [1990]

7.4.1 Phänomene von Schädigung in metallischen Werkstoffen

Belastet man Bauteile aus metallischen Werkstoffen deutlich über ihre Fließgrenze σ_0 , stellt man nach einem relativ großen Bereich metallischer Verfestigung auch nach Erreichen eines Maximalwertes eine deutliche Resttragfähigkeit fest. In diesem Bereich fällt allerdings die maximal aufnehmbare Last bei weiterer Deformation ab, wobei sich das betrachtete Bauteil an einer bestimmten Stelle einschnürt (Dehnungslokalisierung). Mikroskopisch betrachtet geht dieser globale Versagensvorgang auf eine zunehmende Konzentration von Mikroporen zurück, die den tragenden Querschnitt in Bauteilen dieser Materialklasse deutlich reduzieren. Das Auftreten von Mikroporen soll hier mit „Schädigung“ identifiziert werden und läuft im wesentlichen in drei Schritten ab. Zuerst entstehen Mikroporen bei entsprechender Belastung an sogenannten Keimstellen (z.B. Fehlstellen, Kongruenzen, Fremdeinschlüsse/Größenordnung in Stählen: μm). Dieser Prozeß wird als „Nukleation“ bezeichnet und wird maßgeblich vom aktuellen Druckzustand $p = -\frac{1}{3}\sigma_{ij}\delta_{ij}$ beeinflusst. Weitere, zunehmende Belastung führt zu einem Wachstum bestehender Poren, bis sich diese Poren wiederum vereinigen („Koaleszenz“) und damit im folgenden auch einen mit bloßem Auge erkennbaren Makroriss bilden können.

7.4.2 Kontinuumsschädigungsmodell

Um diesen mikroskopischen Prozeß nun mathematisch beschreiben zu können, verschmiert man gedanklich die inhomogenen, diskreten Materialeigenschaften

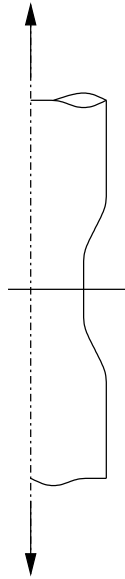


Abbildung 7.6: Dehnungslokalisierung

ten der Mikroebene zu einer kontinuierlichen Beschreibung auf globaler Makroebene (Voraussetzung RVE!). Dies geschieht durch Einführen einer skalaren Größe f , die im kontinuierlichen Modell dem Porenvolumenanteil des realen Materials repräsentieren soll und im weiteren die Schädigung beschreiben wird. Das die Poren umgebende Material wird als Matrixmaterial bezeichnet. Alle Spannungs- und Verzerrungsgrößen stellen jetzt im kontinuierlichen Modell effektive Größen zur Beschreibung des Materialzustandes dar.

7.4.3 Konstitutivgleichungen nach GURSON und TVERGAARD

Kontinuumsmodell zur Beschreibung *duktiler Schädigung* (GURSON [1977], TVERGAARD [1989]) als verallgemeinertes Plastizitätsgesetz

- Fließbedingung:

$$\Phi = \left(\frac{q}{\sigma_\mu} \right)^2 + 2q_1 f^* \cosh \left(-\frac{3}{2} \frac{p}{\sigma_\mu} q_2 \right) - 1 + q_1 f^{*2} = 0 \quad (7.41)$$

wobei

$$\sigma_\mu = \sigma_0 \left(\frac{\varepsilon_\mu^p}{\varepsilon_0} + 1 \right)^{\frac{1}{N}}, \quad \varepsilon_0 = \frac{\sigma_0}{E} \quad \text{Matrixspannung} \quad (7.42)$$

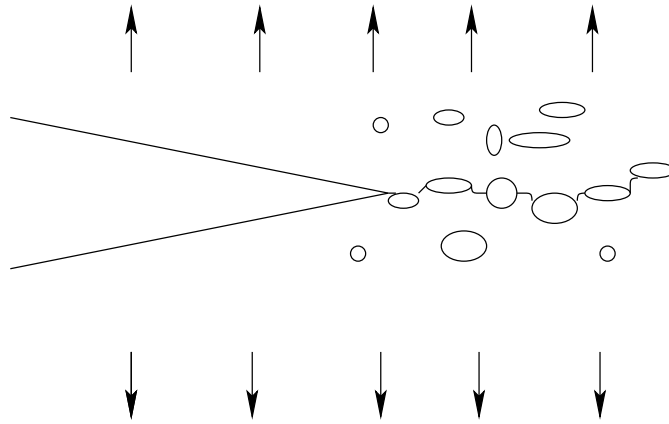


Abbildung 7.7: Makroriss

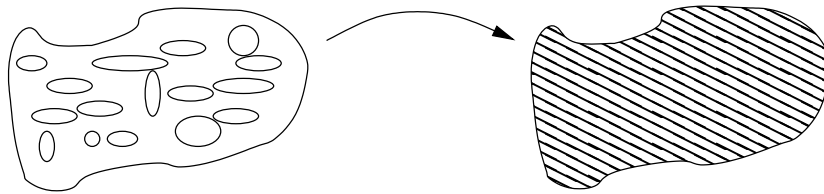


Abbildung 7.8: Verschmieren

$$f^*(t) = \begin{cases} f & ; f \leq f_c \\ f_c + \frac{q_1 - f_c}{f_F - f_c} (f - f_c) & ; f > f_c \end{cases} \quad (7.43)$$

$$q = \left(\frac{3}{2} s_{ij} s_{ij} \right)^{\frac{1}{2}} \quad \text{makroskopische Vergleichsspannung} \quad (7.44)$$

$$f_c, f_F, q_1, q_2 \quad \text{Materialparameter} \quad (7.45)$$

- elastische Verzerrung $\boldsymbol{\sigma} = \mathbb{C}\boldsymbol{\epsilon}_e$
- plastische Verzerrung, Evolution nach Normalenregel: $\dot{\boldsymbol{\epsilon}}_p = \lambda \frac{\partial \Phi}{\partial \boldsymbol{\sigma}}$
- $\dot{\boldsymbol{\epsilon}} = \dot{\boldsymbol{\epsilon}}_e + \dot{\boldsymbol{\epsilon}}_p$
- Konsistenzbedingung $\lambda \dot{\Phi} = 0$
- Evolution der Vergleichsdehnung des Matrixmaterials:

$$(1 - f) \sigma_M \dot{\epsilon}_M^p = \boldsymbol{\sigma} : \dot{\boldsymbol{\epsilon}}_p \quad (7.46)$$

Spannungleistung auf Mikro = Spannungleistung auf Makroebene

- Schädigungsevolution: $\dot{f} = \dot{f}_{Wachstum} + \dot{f}_{Neubildung}$ wobei (vgl. CHU & NEEDLEMAN [1980])

$$\begin{aligned} \dot{f}_{Wachstum} &= (1 - f)\text{tr}(\dot{\epsilon}_p) \\ \dot{f}_{Neubildung} &= \left\{ \frac{f_N}{s_N \sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{\epsilon_\mu^p - \epsilon_N}{s_N} \right)^2 \right] \right\} \dot{\epsilon}_\mu^p \end{aligned} \quad (7.47)$$

7.4.4 Numerische Umsetzung nach ARAVAS

- grundsätzliches Vorgehen:

$$p = -\frac{1}{3}\sigma_{ij}\delta_{ij}, \quad q = \sqrt{\frac{3}{2}s_{ij}s_{ij}}, \quad \mathbf{n} = \frac{3}{2q}\mathbf{s} \quad (7.48)$$

$$\boldsymbol{\sigma} = -p\mathbf{I} + \frac{2}{3}q\mathbf{n} \quad \dot{\epsilon}_p = \Delta\epsilon_p = \frac{1}{3}\Delta\epsilon_p\mathbf{I} + \Delta\epsilon_q\mathbf{n} \quad (7.49)$$

- aus Normalenregel:

$$\Delta\epsilon_p = -\lambda \frac{\partial\Phi}{\partial p}, \quad \Delta\epsilon_q = \lambda \frac{\partial\Phi}{\partial q} \quad (7.50)$$

algebraische Elimination von λ liefert:

$$\Delta\epsilon_p \frac{\partial\Phi}{\partial q} + \Delta\epsilon_q \frac{\partial\Phi}{\partial p} = 0 \quad (7.51)$$

- Insgesamt: lokales nichtlineares Gleichungssystem für $\Delta\epsilon_p, \Delta\epsilon_q, \Delta\epsilon_M^p, \Delta f$

<p>(1) $\Phi = 0$</p> <p>(2) (7.51)</p> <p>(3) Evolutionsgleichung für $\Delta\epsilon_\mu^p$</p> <p>(4) Evolutionsgleichung für Δf</p>	\rightsquigarrow	Lösung mit NEWTON-Verfahren
--	--------------------	-----------------------------

(7.52)

- Linearisierung

$$\boldsymbol{\sigma} = \mathbb{C}^e : \left(\boldsymbol{\epsilon} - \frac{1}{3} \Delta \epsilon_p \mathbf{I} - \Delta \epsilon_q \mathbf{n} \right) \quad (7.53)$$

$$\delta \boldsymbol{\sigma} = \mathbb{C}^e : \left(\delta \boldsymbol{\epsilon} - \frac{1}{3} \delta \Delta \epsilon_p \mathbf{I} - \delta \Delta \epsilon_q \mathbf{n} - \Delta \epsilon_q \frac{\partial \mathbf{n}}{\partial \boldsymbol{\sigma}} : \delta \boldsymbol{\sigma} \right) \quad (7.54)$$

Problematik bei Beschreibung von entfestigendem Materialverhalten mit der FEM: „Verlust der Elliptizität“

7.5 Behandlung finiter Inelastizität

Kinematics

The framework of multiplicative elastoplasticity is used. Its kinematic key assumption is the multiplicative split of the deformation gradient, which maps material points \mathbf{X} onto the current configuration \mathbf{x} ,

$$\mathbf{F} := \partial \mathbf{x} / \partial \mathbf{X} = \mathbf{F}_e \cdot \mathbf{F}_p \quad (7.55)$$

into an elastic and an inelastic part, providing the basis of a geometrically exact theory and avoiding linearization of any measure of deformation. Note that

$$d\hat{\mathbf{x}} = \mathbf{F}_p \cdot d\mathbf{X} = \mathbf{F}_e^{-1} \cdot d\mathbf{x} \quad (7.56)$$

introduces a so-called *intermediate configuration*, which quantities are labeled by $(\hat{\bullet})$. As a further advantage, fast and numerically stable iterative algorithms, proposed and described in SIMO [1992], can be used. In the following, only a brief summary of the integration algorithm for a time step $[t_n; t_{n+1}]$ in the context of a FE-implementation is given. Note that in the following the index $n + 1$ is suppressed for brevity if misunderstanding is unlikely to occur.

The essential aspect of the multiplicative decomposition is the resulting additive structure of the current logarithmic principal strains within the return mapping scheme as $\boldsymbol{\epsilon}^e = \boldsymbol{\epsilon}^{tr} - \Delta \boldsymbol{\epsilon}^p$. Here, $\boldsymbol{\epsilon}^e$ and $\boldsymbol{\epsilon}^{tr}$ stand for a vector representation with the components $\epsilon_i^e = \ln \mu_i^e$ and $\epsilon_i^{tr} = \ln \mu_i^{tr}$, respectively, strictly connected with the spectral decomposition of the elastic left CAUCHY–GREEN tensor.

The elastic left CAUCHY–GREEN tensor can be specified with the multiplicative decomposition (7.55) as

$$\mathbf{b}_e = \mathbf{F}_e \cdot \mathbf{F}_e^T = \mathbf{F} \cdot \mathbf{C}_p^{-1} \cdot \mathbf{F}^T, \quad (7.57)$$

where the superscripts “-1” and “T” denote the inverse and the transpose of a tensor, respectively. That relation clearly shows the “connection” between the elastic and inelastic deformation measure by the occurrence of the inelastic right CAUCHY–GREEN tensor $\mathbf{C}_p = \mathbf{F}_p^T \cdot \mathbf{F}_p$. By means of the relative deformation gradient, see SIMO [1992] and TSAKMAKIS & WILLUWEIT [2003],

$$\Delta \mathbf{F} = \partial \mathbf{x}_{n+1} / \partial \mathbf{x}_n = \mathbf{F}_{n+1} \cdot \mathbf{F}_n^{-1} , \quad (7.58)$$

which relates the current configuration \mathbf{x}_{n+1} to the configuration belonging to the previous time step at t_n , an elastic *trial*-state $\mathbf{b}_e^{tr} = \Delta \mathbf{F} \cdot \mathbf{b}_n \cdot \Delta \mathbf{F}^T$ is calculated for the current configuration with frozen internal variables at state t_n .

In the considered case of isotropy, \mathbf{b}_e commutes with $\boldsymbol{\tau}$, see REESE & WRIGGERS [1997], SIMO [1992]. We assume to fix the principle axes of \mathbf{b}_e during the return mapping scheme described in the previous section, so that the spectral decomposition

$$\mathbf{b}_e^{tr} = \sum_{i=1}^3 \mu_i^{tr2} \mathbf{n}_i^{tr} \otimes \mathbf{n}_i^{tr} \quad (7.59)$$

is given and the eigenvectors \mathbf{n}_i^{tr} can also be used to compose the stress tensor $\boldsymbol{\tau} = \sum_{i=1}^3 \tau_i \mathbf{n}_i^{tr} \otimes \mathbf{n}_i^{tr}$. That motivates the evaluation of the constitutive equations given in the previous section in principle axes, which means additionally a time saving compared to an evaluation of all six (symmetric) tensor components.

Furthermore, for the elastic part of the material description any hyper-elastic behaviour can be used.

Remark

The general concept of LIE time derivative $\mathcal{L}_{\mathbf{v}}(\bullet)$ characterizing the change of a spatial field in the direction of the vector \mathbf{v} and known to yield objective spatial fields, see HOLZAPFEL [2000], leads in this case to the OLDROYD rate of the elastic left CAUCHY–GREEN tensor

$$\mathcal{L}_{\mathbf{v}} \mathbf{b}_e = \overset{\nabla}{\mathbf{b}}_e = \dot{\mathbf{b}}_e - \mathbf{l} \cdot \mathbf{b}_e - \mathbf{b}_e \cdot \mathbf{l}^T \quad (7.60)$$

where $\dot{\mathbf{b}}_e$ denotes the material time derivative and $\mathbf{l} = \text{grad } \dot{\mathbf{x}} = \dot{\mathbf{F}} \cdot \mathbf{F}^{-1}$ the spatial velocity gradient. In this case \mathbf{v} is identified as velocity vector $\mathbf{v} = \dot{\mathbf{x}} = \partial \mathbf{x} / \partial t$. The decomposition of $\mathbf{l} = \mathbf{d} + \mathbf{w}$ in its symmetric part $\mathbf{d} = \text{sym}(\mathbf{l}) = \frac{1}{2}(\mathbf{l} + \mathbf{l}^T)$ and its antimetric part \mathbf{w} , known as spin tensor, respectively, plays

a crucial role in the definition of the inelastic flow rule. Some basic algebraic manipulations let us also obtain the expressions in (7.60) as

$$\mathcal{L}_{\mathbf{v}}\mathbf{b}_e = -2\mathbf{F}_e \cdot \text{sym}(\hat{\mathbf{l}}_p) \cdot \mathbf{F}_e^{\text{T}} = -2\text{sym}(\mathbf{l}_p \cdot \mathbf{b}_e) , \quad (7.61)$$

where $\hat{\mathbf{l}}_p$ is defined by $\hat{\mathbf{l}}_p = \dot{\mathbf{F}}_p \cdot \mathbf{F}_p^{-1}$ acting on the intermediate configuration. Please note, that we do *not* make any assumption concerning the antisymmetric part \mathbf{w} of \mathbf{l} . Because of the restriction to isotropic material behaviour, the focus is just directed to the symmetric part \mathbf{d} of \mathbf{l} . So, the additive decomposition $\mathbf{d} = \mathbf{d}_e + \mathbf{d}_p$ results from the multiplicative decomposition $\mathbf{F} = \mathbf{F}_e \cdot \mathbf{F}_p$.

The definition of the associated flow rule in that finite strain regime as

$$\mathbf{d}_p := \gamma \frac{\partial \Phi}{\partial \boldsymbol{\tau}} \quad (7.62)$$

enables with (7.60) and (7.61) the formulation

$$\mathcal{L}_{\mathbf{v}}\mathbf{b}_e = -2\text{sym}\left(\gamma \frac{\partial \Phi}{\partial \boldsymbol{\tau}} \cdot \mathbf{b}_e\right) , \quad (7.63)$$

where γ here is known as “consistency parameter”, see SIMO [1992].

Stress Representation

Following ARAVAS [1987], we write the KIRCHHOFF stress tensor $\boldsymbol{\tau}$ as the weighted CAUCHY stress tensor as

$$\boldsymbol{\tau} = J \boldsymbol{\sigma} = -p^\tau \mathbf{I} + 2/3 q^\tau \hat{\mathbf{n}} , \quad (7.64)$$

where $J := \det \mathbf{F}$ is the determinant of the deformation gradient. The scalar $p^\tau = -\tau_{ij} \delta_{ij} / 3$ defines the hydrostatic pressure, $q^\tau = \sqrt{3/2 t_{ij} t_{ij}}$ is the equivalent KIRCHHOFF stress and $t_{ij} = \tau_{ij} + p^\tau \delta_{ij}$ are the components of the KIRCHHOFF stress deviator¹. These quantities can also be obtained for the CAUCHY stress tensor, whose deviatoric stress is $\mathbf{s} = \boldsymbol{\sigma} + p^\sigma \mathbf{I}$. In this notation, an additional important quantity is the normalized and dimensionless stress deviator

$$\hat{\mathbf{n}} = 3/(2q^\tau) \mathbf{t} = 3/(2q^\sigma) \mathbf{s} . \quad (7.65)$$

The second order unit tensor \mathbf{I} is defined as the KRONECKER symbol by its components δ_{ij} in the cartesian frame.

¹EINSTEIN summation: $i, j = 1, 2, 3$

Internal System of Equations

Analogous to (7.64), the inelastic strain rate can be written as

$$\Delta \boldsymbol{\epsilon}^p = \frac{1}{3} \Delta \varepsilon_p \mathbf{I} + \Delta \varepsilon_q \hat{\mathbf{n}} , \quad (7.66)$$

where $\Delta \varepsilon_p$ and $\Delta \varepsilon_q$ describe scalar rate quantities which are defined below. Note, that again the dimensionless tensor quantities \mathbf{I} and $\hat{\mathbf{n}}$ are used in this notation.

The two scalar valued quantities $\Delta \varepsilon_p$ and $\Delta \varepsilon_q$ are given with (7.62) as

$$\Delta \varepsilon_p = -\gamma \left(\frac{\partial \Phi}{\partial p} \right) \quad \text{and} \quad \Delta \varepsilon_q = \gamma \left(\frac{\partial \Phi}{\partial q} \right) . \quad (7.67)$$

The algebraic elimination² of γ in (7.67) leads to

$$\Delta \varepsilon_p \left(\frac{\partial \Phi}{\partial q} \right) + \Delta \varepsilon_q \left(\frac{\partial \Phi}{\partial p} \right) = 0 . \quad (7.68)$$

With that and a yield surface of the type

$$\Phi(p, q, \varepsilon_M^p, f) = 0 \quad (7.69)$$

the internal set of equations is completed by an isotropic hardening rule

$$k_{iso} = k_{iso}(s) , \quad (7.70)$$

where s is the plastic arc-length and its increment $\dot{s} \equiv \gamma$.

If the condition $\Phi \leq 0$ (see (7.69)) is fulfilled by the current stress state $\boldsymbol{\tau}$, this state is possible and it is the solution. If, on the other hand, $\Phi \leq 0$ is violated by the trial-state, the trial stresses must be projected back on the yield surface $\Phi = 0$ in an additional step, often called “exponential return mapping”.

In that case, $\mathbf{x} = \mathbf{x}_{n+1}$ is fixed and (7.63) results in

$$\overset{\nabla}{\mathbf{b}}_e = \dot{\mathbf{b}}_e = -2\gamma \text{sym} \left(\frac{\partial \Phi}{\partial \boldsymbol{\tau}} \cdot \mathbf{b}_e \right) , \quad (7.71)$$

with $\mathbf{l} \equiv \mathbf{0}$. The solution of the first order differential equation (7.71) is given by

$$\mathbf{b}_{e,n+1} = \sum_{i=1}^3 \underbrace{\exp [2\epsilon_i^e]}_{\mu_i^{e,2}} \mathbf{n}_{(i)n+1}^{tr} \otimes \mathbf{n}_{(i)n+1}^{tr} , \quad (7.72)$$

²Please note: Here, we consider the very general case, where the plastic strain rate is pressure dependent, which allows that algebraic manipulation. Otherwise, the “consistency parameter” γ has to be determined instead of $\Delta \varepsilon_p$. In that case, we obtain $\Delta s = \sqrt{2/3} \Delta \varepsilon_q$ as algorithmic counterpart to \dot{s} .

where the elastic logarithmic strains $\boldsymbol{\epsilon}^e$ are obtained in principle axes, see (7.73) below, so that $\mathbf{b}_{e,n+1}$ is known and $\mathbf{C}_p^{-1} = \mathbf{F}^{-1} \cdot \mathbf{b}_{e,n+1} \cdot \mathbf{F}^{-T}$ can be stored as *history variable* for the next time step.

HISTORY–Storage

With these results of the integration procedure above for $\Delta\boldsymbol{\epsilon}_{n+1}^p$, the actual elastic left CAUCHY–GREEN tensor is computed by

$$\boldsymbol{\epsilon}_{n+1}^e = \boldsymbol{\epsilon}^{tr} - \Delta\boldsymbol{\epsilon}_{n+1}^p \quad (7.73)$$

and (7.72) in reversal of the spectral decomposition (7.59), see REESE & WRIGGERS [1997].

Kapitel 8

Grundlagen der Methode der finiten Elemente

Literatur:

HUGHES [2000], WRIGGERS [2001], SIMO & HUGHES [1998], BAASER [2010a]

8.1 Schwache Form — Prinzip der virtuellen Verschiebungen

$$\int_{\mathcal{B}} \boldsymbol{\sigma} : \delta \boldsymbol{\varepsilon} \, dV - \int_A \mathbf{t}^T \cdot \delta \mathbf{u} \, da \stackrel{!}{=} 0 \quad (8.1)$$

Aus dem Erhaltungssatz des Impulses folgt durch partielle Integration mit Hilfe des CAUCHY-Theorems $\mathbf{t} = \boldsymbol{\sigma} \mathbf{n}$ und unter Vernachlässigung von Volumenlasten die lokale Impulsbilanz

$$\operatorname{div} \boldsymbol{\sigma}^T = \mathbf{0} . \quad (8.2)$$

Multipliziert man diese mit einer Testfunktion $\boldsymbol{\eta}$, die später als Vektor der zulässigen virtuellen Verschiebungen $\delta \mathbf{u}$ identifiziert wird und die kinematische Randbedingung des Feldproblems erfüllt, und integriert über das Körpervolumen, so erhält man unter Verwendung des GAUSSSchen Satzes, d.h. Einarbeitung der Spannungsrandbedingungen, das Prinzip der virtuellen Verschiebungen als Funktional

$$g(\mathbf{u}, \delta \mathbf{u}) := \delta \Pi = \int_{\mathcal{B}} \boldsymbol{\sigma} \operatorname{grad} \delta \mathbf{u} \, dv - \int_{\partial \mathcal{B}_\sigma} \mathbf{t}^{\text{Rand}} \delta \mathbf{u} \, da \stackrel{!}{=} 0 . \quad (8.3)$$

Damit wird der Gleichgewichtszustand beschrieben. Unbekannt ist das Verschiebungsfeld: $\mathbf{u} := \mathbf{x} - \mathbf{X}$

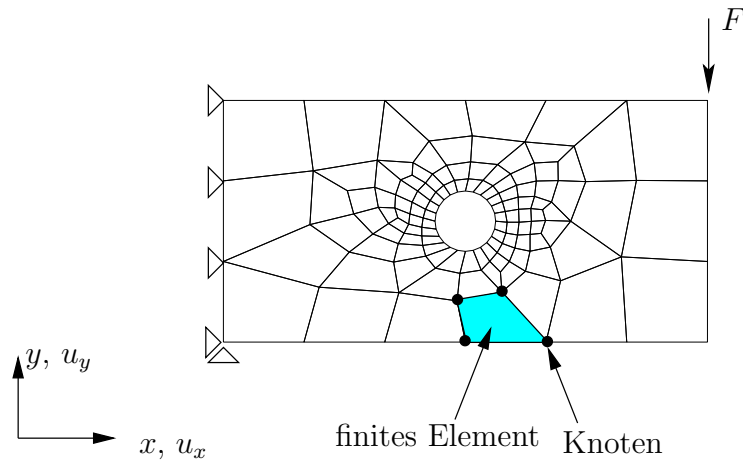


Abbildung 8.1: Diskretisierte Struktur mit Rand-/Lagerbedingungen und Belastung F

8.2 Linearisierung und Diskretisierung

Die Entwicklung des Prinzips der virtuellen Verschiebungen (in der Referenzkonfiguration) geschieht um eine bekannte Gleichgewichtslage $\hat{\mathbf{u}} := \mathbf{u} - \Delta\mathbf{u}$ in eine TAYLOR-Reihe, wobei nach dem linearen Glied abgebrochen wird.

$$G(\mathbf{u}, \delta\mathbf{u}) \Big|_{\mathbf{u}=\hat{\mathbf{u}}} \approx G(\hat{\mathbf{u}}, \delta\mathbf{u}) + \mathcal{D}G(\hat{\mathbf{u}}, \delta\mathbf{u}) \cdot \Delta\mathbf{u} \stackrel{!}{=} 0 \quad (8.4)$$

$$\text{mit } \mathcal{D}G(\hat{\mathbf{u}}, \delta\mathbf{u}) := \frac{d}{d\epsilon} [G(\hat{\mathbf{u}} + \epsilon\Delta\mathbf{u}, \delta\mathbf{u})]_{\epsilon \rightarrow 0} \quad (8.5)$$

Das bedeutet nach einer Rücktransformation in die Momentankonfiguration:

$$\begin{aligned} \mathcal{D}g(\hat{\mathbf{u}}, \delta\mathbf{u}) &= \Delta \int_{\mathcal{B}_0} J \boldsymbol{\sigma} \delta \mathbf{e} \, dV \\ &= \int_{\mathcal{B}_0} J (\Delta \boldsymbol{\sigma} + \text{grad} \Delta \mathbf{u} \boldsymbol{\sigma}) \text{grad} \delta \mathbf{u} \, dV \end{aligned} \quad (8.6)$$

Diskretisierung, „Isoparametrisches Konzept“

- Geometrie:

$$\mathbf{x} = \sum_{i=1}^{nel} N_i(\boldsymbol{\xi}) = \mathbf{x}_i \quad (8.7)$$

- Verschiebungsfeld:

$$\mathbf{u} = \sum_{i=1}^{nel} N_i(\boldsymbol{\xi}) = \mathbf{v}_i \quad (8.8)$$

N_i : „Formfunktionen“ (z.B. bilinear) für ebenes 4-Knotenelement

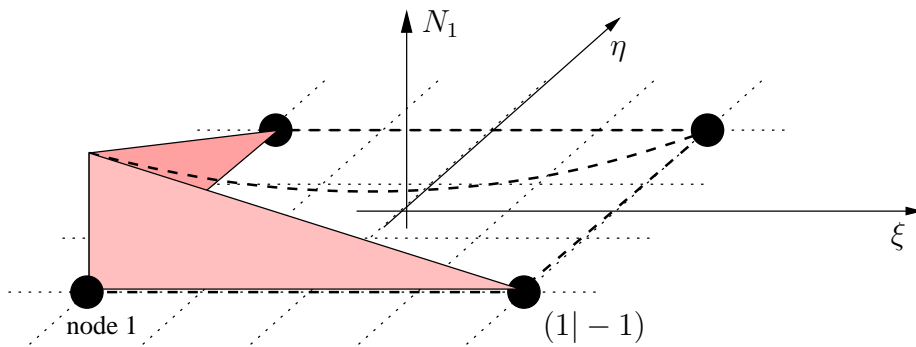


Abbildung 8.2: Ebenes 4-Knoten-Element

$$N_i = \frac{1}{4}(1 + \xi_i \xi)(1 + \eta_i \eta) \quad i = 1, 2, 3, 4; \boldsymbol{\xi} = \begin{bmatrix} \xi \\ \eta \end{bmatrix} \quad (8.9)$$

Numerische Integration am Beispiel eines 1-dimensionalen Integrals:

$$\int_0^l f(x) dx \quad (8.10)$$

Transformation: $x = x(\xi) = \frac{1}{2}(\xi + 1)l$

$$\rightsquigarrow \int_{-1}^{-1} f[x(\xi)] \frac{dx}{d\xi} d\xi = \int_{-1}^{-1} g(\xi) d\xi \quad (8.11)$$

Man kann zeigen:

$$\int_{-1}^{-1} g(\xi) d\xi \approx \sum_{p=1}^n g(\xi_p) \cdot w_p \quad (8.12)$$

Die größte Genauigkeit erreicht man hierbei mit der GAUSSintegration (exakt

	Grad n	„GAUSS-Pkt.“ ξ_p	Gewicht w_p	exakt für q
für Polynome vom Grad $q \leq 2n-1$:	1	0	2	1
	2	$\pm -\frac{1}{\sqrt{3}} \approx \pm 0,5377$	1	3
	3	$\left\{ \begin{matrix} 0 \\ \pm\sqrt{\frac{3}{5}} \end{matrix} \right\}$	$\left\{ \begin{matrix} \frac{8}{9} \\ \frac{5}{9} \end{matrix} \right\}$	5

8.2.1 Typen finiter Elemente in der Strukturmechanik

Abhängig von der Fragestellung und den damit gewählten Ansatzfunktionen ergeben sich als (einfachste) Typen finiter Elemente im 2D-Fall Dreieck- oder Viereck-Elemente, siehe Abb. 8.3. Für Aufgaben, die axialsymmetrisch

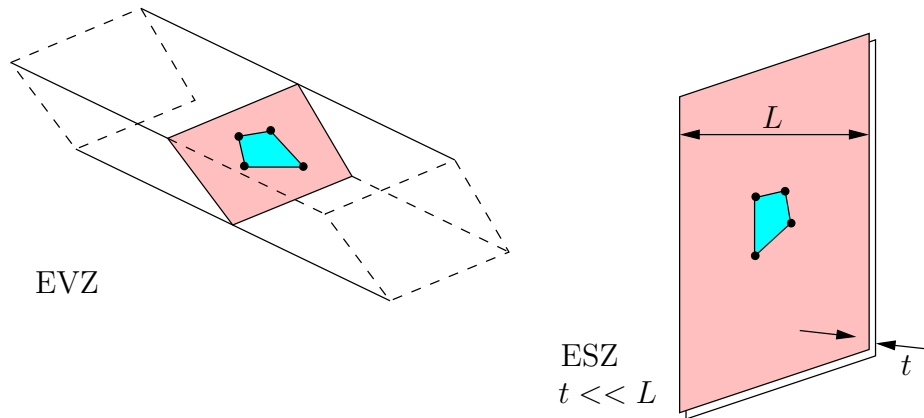


Abbildung 8.3: 2D-Elemente. Idealisiert *ebener Verzerrungszustand (EVZ)* und *ebener Spannungszustand (ESZ)*

sind, lassen sich auf Basis der 2D-Elemente spezielle Formulierungen für axialsymmetrische Elemente angeben, siehe Abb. 8.4. Diese werden oft als $2\frac{1}{2}$ D-Elemente bezeichnet. Ebenso lassen sich für 3D-Modelle entsprechende Elemente als Tetraeder- oder Würfel¹-Elemente mit mind. 4 bzw. mind. 8 Knoten angeben, siehe Abb. 8.5. Durch die Wahl von Ansatzfunktionen höherer Ordnung (*p-Methode*) lassen sich sowohl im 2D- als auch im 3D-Fall Elemente mit entsprechend mehr Knoten (zus. an den Kanten oder im Innern) realisieren. Damit wird die Abbildungsleistung dieser Formulierungen, aber auch der Rechenaufwand deutlich erhöht. Es gilt, einen Kompromiss

¹In den seltensten Fällen sind die Formen dieser Elemente exakte Würfel. Die Bezeichnung hat sich bloß eingebürgert. Im englischen Sprachgebrauch würde man von *Brick-Elementen* reden, was nicht unbedingt besser ist.

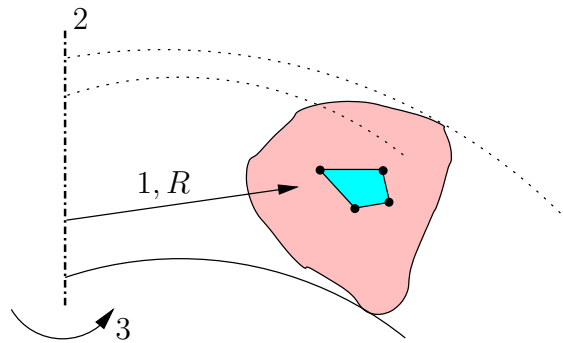


Abbildung 8.4: Axialsymmetrische „2D“-Elemente

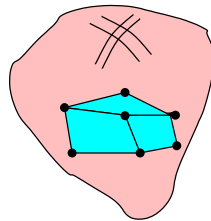


Abbildung 8.5: 3D-Elemente zur Diskretisierung von Volumen

zwischen *mehr Elementen niedrigerer Ordnung (h-Methode)* und *wenigen Elementen höherer Ordnung (p-Methode)* zu finden.

Spezielle Strukturelemente

- Stab: 2-Knoten-Element in 2D oder 3D
- Balken: 2 oder mehr Knoten in 2D oder 3D mit zusätzlichen VerdrehFHG
- Platten-Element(e)
- Schalen-Elemente: Membrane (*biegeschlaff*) und höherwertig mit Biegung
- *Wedge-Elemente*, die als degenerierte 8-Knoten-Elemente entstehen und einem Tortenstück ähneln. Oft verwendet, um 3D-Vernetzungen einfacher an Ecken oder Kanten realisieren zu können.

8.3 Iteratives Vorgehen

8.3.1 Globales NEWTON-Verfahren

[Globales NEWTON-Verfahren zur Lösung von (8.4)] Gleichgewichtsiteration (Bronstein „Nichtlineare Gleichungen“)

→ „direkte Lösung“, NEWTON-Verfahren

Aus $G(\hat{\mathbf{u}}, \delta \mathbf{u}) + \mathcal{D}G(\hat{\mathbf{u}}, \delta \mathbf{u}) \cdot \Delta \mathbf{u} = 0$ mit $\Delta \mathbf{u} = \mathbf{u} - \hat{\mathbf{u}}$ ergibt sich durch Auflösen:

$$\mathbf{u} = \hat{\mathbf{u}} - \underbrace{[\mathcal{D}G(\hat{\mathbf{u}}, \delta \mathbf{u})]^{-1}}_{\text{„Steifigkeit“ } \mathbf{K}} \cdot \underbrace{G(\hat{\mathbf{u}}, \delta \mathbf{u})}_{\text{„rechte Seite“ } -\mathbf{r}} \quad (8.13)$$

- Assemblierung/Zusammenbau des Gesamtsystems \mathbf{K} und \mathbf{r} erfolgt durch FEM-Programmpaket. Hierbei muss die Zeilen- und Spaltenbelegung beachtet werden.
- Die Verknüpfung zwischen den unbekanntem Verschiebungen/Verzerrungen und den resultierenden Spannungen liegt im Materialmodell.
- FEM-Paket löst global $\mathbf{K} \Delta \mathbf{u} = \mathbf{r}$ und aktualisiert damit \mathbf{u} .
- Die Wiederholung dieses Ablaufes erfolgt bis die Lösung ausreichend angenähert ist.

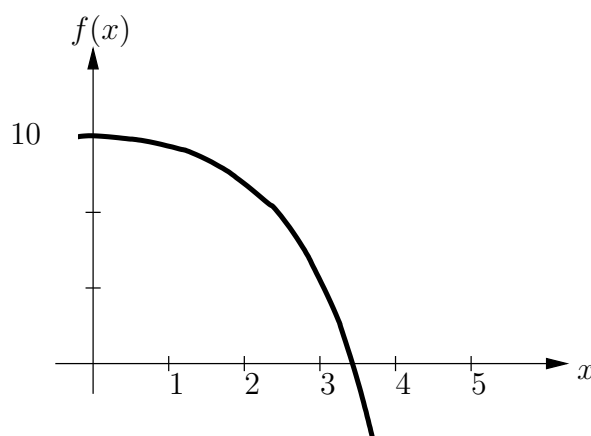


Abbildung 8.6: Nullstellensuche

Das NEWTON-Verfahren wird nun an einem Beispiel erläutert:

$$f(x) = -x^2 - \sin(x) + 10; \quad f'(x) = -2x - \cos(x) \quad (8.14)$$

Nullstellensuche: $f(x) \stackrel{!}{=} 0$

$$\text{Algorithmus: } x^{(m+1)} = x^{(m)} - \frac{f(x^{(m)})}{f'(x^{(m)})}; \quad f'(x^{(m)}) \neq 0$$

$$\begin{aligned} \Rightarrow \quad & x^{(0)} = 0 \\ & x^{(1)} = 10 \qquad f(x^{(1)}) = -89,456 \\ & x^{(2)} = 5,331 \qquad f(x^{(2)}) = -17,6086 \\ & x^{(3)} = 3,76512 \qquad f(x^{(3)}) = -3,59223 \\ & x^{(4)} = 3,230436 \qquad f(x^{(4)}) = -0,34699 \\ & x^{(5)} = 3,1669407 \qquad f(x^{(5)}) = -0,00416807 \\ & x^{(6)} = 3,166159315 \qquad f(x^{(6)}) = -6,17559 \cdot 10^{-7} \end{aligned} \tag{8.15}$$

- MATHEMATICA, „exakt“: 3,166159199192307...
- Konvergenz: „mindestens quadratisch“; Kriterium: $|f(x)|$ oder $|f(x^{(m+1)}) - f(x^{(m)})| < \epsilon$

8.3.2 Behandlung großer, linearer Gleichungssysteme

LDL^T-Zerlegung (direktes Verfahren)

Eine symmetrische Matrix \mathbf{K} (in strukturmechanischen Problemen i.a. die Regel) läßt sich eindeutig in $\mathbf{K} = \mathbf{L} \cdot \mathbf{D} \cdot \mathbf{L}^T$ zerlegen, wenn \mathbf{D} eine Diagonalmatrix und \mathbf{L} eine untere Dreiecksmatrix mit $L_{(ii)} = 1$ darstellt. Als Übung kann man eine beliebige 3×3 -Matrix durch GAUSS-Elimination invertieren. So sind die Einträge L_{ij} leicht zu erkennen.

Vorgehensweise zur Lösung von $\mathbf{K} \cdot \Delta \mathbf{u} = \mathbf{r}$:

$$\begin{aligned} \text{Lösen von } \mathbf{L} \cdot \mathbf{v} = \mathbf{r} & \Rightarrow \mathbf{v} \\ \text{Lösen von } \mathbf{L}^T \cdot \Delta \mathbf{u} = \mathbf{D}^{-1} \cdot \mathbf{v} & \Rightarrow \Delta \mathbf{u} \end{aligned} \tag{8.16}$$

Anmerkungen zur Rechnerimplementierung:

- Ausnutzen der Symmetrie, positive Definitheit, Bandstruktur (Einfluss der Knotennumerierung)
- „Profilspeicherung“/spaltenorientiert/aber: „fill-in“!

Iterative Verfahren

Die iterative Lösung von $\mathbf{K} \Delta \mathbf{u} = \mathbf{r}$ in das vorher beschriebene NEWTON-Verfahren zur Lösung des globalen, möglicherweise nichtlinearen Systems kann zur Bestimmung von $\Delta \mathbf{u}$ ebenso ein wiederum iteratives Verfahren integriert werden. Beispiele dafür sind das sogenannte CG-Verfahren {Conjugated

Gradient} oder das LANCZOS-Verfahren. Dies hat vor allem bei sehr großen Gleichungssystemen Zeit- aber auch Speicherplatzvorteile, weil \mathbf{K}^{-1} dabei nicht bestimmt werden muss.

8.3.3 Iterative Lösung des globalen Gleichungssystems

Wir haben globale Gleichgewichtsiteration mit NEWTON-Verfahren und aktualisieren

$$\begin{aligned} \mathbf{u} &= \hat{\mathbf{u}} + \Delta \mathbf{u} \\ &= - [\mathcal{D}G(\hat{\mathbf{u}}, \delta \mathbf{u})]^{-1} \cdot G(\hat{\mathbf{u}}, \delta \mathbf{u}) \end{aligned} \quad (8.17)$$

durch Lösung $\Delta \mathbf{u}$.

- iterative Lösung ist bei großen „Systemen“ ($n \gg 10.000$ Unbekannte) effizienter als die direkten Methoden!
- „effizienter“ heißt Speicherplatzersparnis, aber auch Zeitersparnis
- Konstruktion eines Verfahrens unter Ausnutzung der typischen Struktur von \mathbf{K} : dünn besetzt, Bandstruktur, eventuell symmetrisch, meist positiv definit

Vorkonditionierung

Die Konvergenzgeschwindigkeit eines iterativen Verfahrens (zur Lösung schwachbesetzter Gleichungssysteme) hängt sehr stark von der Konditionszahl $k(\mathbf{K})$ der Systemmatrix ab:

$$k(\mathbf{K}) := \frac{\|\mathbf{K}\|_2}{\|\mathbf{K}^{-1}\|_2} = \frac{\lambda_{\max}(\mathbf{K})}{\lambda_{\min}(\mathbf{K})} = \frac{\lambda_{\max}(\mathbf{K})}{\lambda_{\max}(\mathbf{K}^{-1})}. \quad (8.18)$$

Es gilt also

$$1 = k(\mathbf{1}) = k(\mathbf{K}\mathbf{K}^{-1}) \leq k(\mathbf{K}) \quad (8.19)$$

\Rightarrow Optimalfall: $k(\mathbf{K}) = 1$; Ziel: $k(\mathbf{K}) \rightarrow 1$

\leadsto Vorkonditionierung = Modifikation

Es wird $\mathbf{C}^{-1} \cdot \mathbf{K} \cdot \Delta \mathbf{u} = \mathbf{C}^{-1} \cdot \mathbf{r}$ gelöst, wobei $k(\mathbf{C}^{-1} \cdot \mathbf{K}) < k(\mathbf{K})$ gelten muss und \mathbf{C} leicht zu finden/bestimmen und leicht zu invertieren sein soll. Für eine geeignete Wahl von \mathbf{C} siehe u.a. BRAESS [1997].

Einfachste Wahl: $\mathbf{C} := \mathbf{D}$, wobei \mathbf{D} eine Diagonalmatrix mit ausschließlich den Diagonalgliedern von \mathbf{K} darstellt. (= Diagonalvorkonditionierung)

\Rightarrow Wir lösen: $\mathbf{D}^{-1} \cdot \mathbf{K} \cdot \Delta \mathbf{u} = \mathbf{D}^{-1} \cdot \mathbf{r}$

$\mathbf{D}^{-1} \cdot \mathbf{K}$ erfüllt offensichtlich unsere Anforderungen!

- Verfahren der konjugierten Gradienten (CG)

allgemein: Lösen des äquivalenten Minimierungsproblems

$$F(\mathbf{z}) = \frac{1}{2}(\mathbf{b} - \mathbf{A} \cdot \mathbf{z})^T \cdot \mathbf{A}^{-1} \cdot (\mathbf{b} - \mathbf{A} \cdot \mathbf{z}) \rightarrow \min_{z \in \mathbb{R}^n} \quad (8.20)$$

für \mathbf{A} symmetrisch und positiv definit ($\mathbf{A} = \mathbf{D}^{-1} \cdot \mathbf{K}$, $\mathbf{b} = \mathbf{D}^{-1} \cdot \mathbf{r}$)

– Initialisierung:

Start mit $\mathbf{x}_0 \in \mathbb{R}^n$, z.B. $\mathbf{x}_0 = \mathbf{0}$, $\mathbf{p}_0 := \mathbf{r}_0 := \mathbf{b} - \mathbf{A} \cdot \mathbf{x}_0$

– Iteration:

solange $\mathbf{p}_k \neq \mathbf{0}$ für $k = 0, 1, 2, \dots$ berechne:

$$\begin{aligned} a_k &:= \frac{\mathbf{r}_k^T \cdot \mathbf{r}_k}{\mathbf{p}_k^T \cdot \mathbf{A} \cdot \mathbf{p}_k} \rightarrow \mathbf{x}_{k+1} := \mathbf{x}_k + a_k \mathbf{p}_k \\ &\rightarrow \mathbf{r}_k - a_k \mathbf{A} \cdot \mathbf{p}_k \rightarrow b_k := \frac{\mathbf{r}_{k+1}^T \cdot \mathbf{r}_{k+1}}{\mathbf{r}_k^T \cdot \mathbf{r}_k} \\ &\rightarrow \mathbf{p}_{k+1} := \mathbf{r}_{k+1} + b_k \mathbf{p}_k \end{aligned} \quad (8.21)$$

- Lanczos-Verfahren

→ siehe Algorithmus nächste Seite (8.22)

Aufwand und Datenstrukturierung

Aufwand für direktes Verfahren ($\mathbf{K} = \mathbf{L} \cdot \mathbf{D} \cdot \mathbf{K}^T$) mit m als halbe Bandbreite und n als Anzahl der Unbekannten, nach BATHE [1996]: $\frac{1}{2}nm^2 + 2nm$

Aufwand eines Iterationsschrittes innerhalb eines iterativen CG-Verfahrens mit γ als Mittelwert der von Null verschiedenen Einträge pro Zeile: $(\gamma + 5)n$

Erfahrungsgemäß liegt die Anzahl der Iterationschritte weit unterhalb von n , je nach Vorkonditionierung.

Günstige Speichertechnik für das CG-Verfahren oder LANCZOS-Verfahren, wo Matrix- Vektor- Multiplikation benötigt wird, für Skalarrechner:

Bsp:

$$\left[\begin{array}{cccccccc} a_{11} & a_{12} & & a_{14} & a_{15} & & & \\ a_{21} & a_{22} & a_{23} & & & a_{26} & & \\ & a_{32} & a_{33} & & a_{35} & & & \\ a_{41} & & & a_{44} & & a_{46} & & \\ a_{51} & & a_{53} & & a_{55} & & & \\ & a_{62} & & a_{64} & & a_{66} & & \end{array} \right]_{\rightarrow n=6} \quad (8.23)$$

A :	a_{11}	a_{21}	a_{22}	a_{32}	a_{33}	a_{41}	a_{44}	a_{51}	a_{53}	a_{55}	a_{62}	a_{64}	a_{66}
k :	1	1	2	2	3	1	4	1	3	5	2	4	6
ξ :	1 3 5 7 10 13						\triangleq Position der Diagonalglieder						

$$\begin{aligned}
z_1 &= a_1 p_1 \\
z_i &= a_{\xi(i)} p_i \text{ für } i = 1, 2, \dots, n \\
z_i &= z_i + a_j p_{k(j)} \text{ für } j = \xi(i-1) + 1, \dots, \xi(i) - 1 \\
z_{k(i)} &= z_{k(i)} + a_i p_i
\end{aligned}$$

Für dieses Beispiel:

Zähler $i = 4$: (2) $z_4 = a_{7 \cdot 4}$

$j = 6(\dots, 6)$

(3) $z_4 = z_4 + a_{6 \cdot 1}$ „unteres Dreieck“

(4) $z_1 = z_1 + a_{6 \cdot 4}$ „oberes Dreieck“

Anmerkung: Weitere Komponente in $z_4 = z_4 + a_{12 \cdot 6}$ bei $i=6$, dann $j = 11, 12$.

\rightsquigarrow insgesamt: 2 Schleifen, 1 Feld für A, 2 Felder für Zeiger

Teil II

Anwendungen

Kapitel 9

Parameter–Identifikation

9.1 Beispiel: Hyperelastische Werkstoffe

Verzerrungsenergie, „freie Energie(dichte)“ U siehe (??)

9.1.1 Einaxiale Darstellung des neo–HOOKE–Modells

$$P_{1ax\ nH} = G \left(\lambda_{1ax} - \frac{1}{\lambda_{1ax}^2} \right), \quad (9.1)$$

wobei mit G der Schubmodul bezeichnet wird.

9.1.2 YEOH–Modell

siehe Übung in Abschn. 11.2

9.2 Versuchsanordnungen

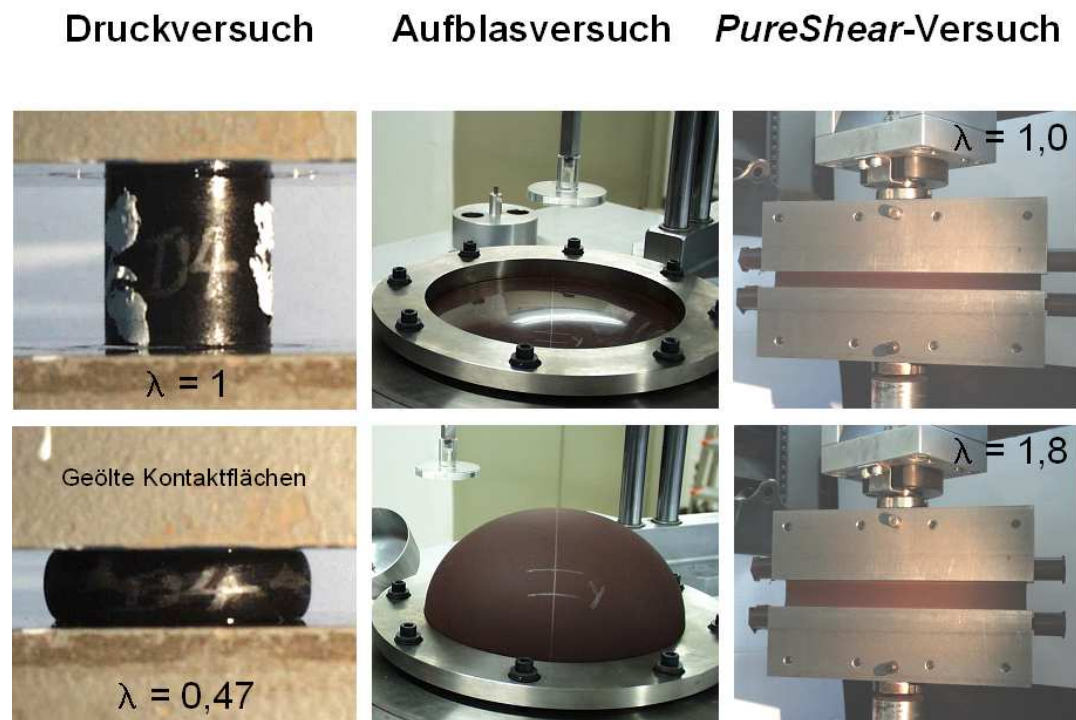


Abbildung 9.1: Versuchstypen zum Anregen unterschiedlicher Deformationen

9.3 Fehler–Quadrat–Minimierung

- Gegeben:
 n Messwerte σ_i^{Mess}
- Berechnet aus gegebenem Modell mit p Parametern:
 n Modellwerte σ_i^{Modell}

Minimieren der Fehlergröße durch Verändern der p freien Parameter:

$$err := \sum_{i=1}^n (\sigma_i^{\text{Mess}} - \sigma_i^{\text{Modell}})^2 \rightsquigarrow \min \quad (9.2)$$

Hier bieten sich MICROSOFT EXCEL (interner *Solver*/PlugIn) oder MATLAB (fminsearch) an:

MATLAB–Beispiel hierzu (Fragmente einer Umsetzung mit m-files):

```
% Startloesung fuer Minimierung
param = [ 1.0 ];

data = dlmread('1ax.dat');          % eps - sigma
data = sort(data,1);
sig_mess = data(:,2);
lambda = data(:,1) + ones(length(sig_mess),1);

% Minimierung
optset = optimset('MaxFunEvals',1.0e6,'TolX',1.0e-8 ...
                 'Display','iter');
param = fminsearch(@(param) minimierung(param,lambda,sig_mess) ...
                  ,param,optset);

Dieser Optimierungsaufruf benötigt die
function dist = minimierung(param,lambda,sig_mess)

sig_nH = neoHooke(param,lambda);

% Residuum
dist = sig_nH - sig_mess;
dist = norm(dist,'fro');

return
```

Kapitel 10

Implementierung von Materialmodellen

Verwendung von ABAQUS repräsentativ, ohne Werbung machen zu wollen !
Analog z.B. ANSYS oder MSC.MARC.

10.1 Hyperelastische Modelle über UHYPER

10.1.1 Schnittstelle

```
SUBROUTINE UHYPER (BI1, BI2, AJ, U, UI1, UI2, UI3, TEMP
X                , NOEL, CMNAME, INCMPLFLAG
X                , NUMSTATEV, STATEV
X                , NUMFIELDV, FIELDV
X                , FIELDVINC, NUMPROPS, PROPS)

      UI1 (1)      UI2 (1)      UI3 (1)
      ∂U/∂I1.      ∂2U/∂I12.      ∂3U/∂I12∂J.
      UI1 (2)      UI2 (2)      UI3 (2)
      ∂U/∂I2.      ∂2U/∂I22.      ∂3U/∂I22∂J.
      UI1 (3)      UI2 (3)      UI3 (3)
      ∂U/∂J.        ∂2U/∂J2.      ∂3U/∂I1∂I2∂J.
      UI2 (4)      UI3 (4)
      ∂2U/∂I1∂I2.      ∂3U/∂I1∂J2.
      UI2 (5)      UI3 (5)
      ∂2U/∂I1∂J.      ∂3U/∂I2∂J2.
      UI2 (6)      UI3 (6)
      ∂2U/∂I2∂J.      ∂3U/∂J3.
```

Abbildung 10.1: ABAQUS-Schnittstelle UHYPER

Auf diese Schnittstelle wird im Beispiel in Abschnitt 11.5 nochmals Bezug genommen.

10.1.2 Zwangsbedingung Inkompressibilität

Verwendung in ABAQUS: $U_{vol} = \frac{1}{D_1} (J - 1)^2$

10.1.3 Aktivieren / Ansprechen in ABAQUS

Eintrag in *.inp-Datei:

```
**
*HYPERELASTIC, USER, TYPE=COMPRESSIBLE, PROPERTIES=numprops
  prop1, prop2, prop3, ...
**
```

In Abschnitt 3.2.2 werden die verschiedenen Möglichkeiten des Aufrufs von ABAQUS dargestellt.

10.2 Allgemeine Material-Schnittstelle UMAT

VOIGT-Notation / ABAQUS-Konvention (siehe Manual 1.2.2-6)

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{11} & \tau_{12} & \tau_{13} \\ \tau_{12} & \sigma_{22} & \tau_{23} \\ \tau_{13} & \tau_{23} & \sigma_{33} \end{bmatrix} \rightsquigarrow \hat{\boldsymbol{\sigma}} = \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \tau_{12} \\ \tau_{13} \\ \tau_{23} \end{bmatrix} = \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{bmatrix} \quad (10.1)$$

10.2.1 Schnittstelle

10.2.2 Bestimmung des Moduls \mathbb{D}

Zur vollständigen Beschreibung des NEWTON-Verfahrens der Gleichgewichtsiteration muss auf Ebene der Integrationspunkte innerhalb jeder Elementformulierung neben der Spannungsantwort ebenfalls noch die Ableitung der Spannung bzgl. des jeweiligen Deformationsmaßes angegeben werden. Diese Materialsteifigkeit bezeichnen wir als Modul \mathbb{D} . Siehe für eine ausführliche Diskussion dazu auch WRIGGERS [2001].

In ABAQUS wird an dieser Stelle die Ableitung der JAUMANNschen Spannungsrate nach der dort verwendeten Dehnrate $\Delta\boldsymbol{\varepsilon}$ verlangt, siehe (5.36) und Theorie-Manual 1.4.3–3 und 4.1.1–4.

Numerische Bestimmung des Moduls \mathbb{D}

Die oben genannte Ableitung der Spannung(srate), die in ABAQUS allerdings für eine (optimale) Konvergenz — vor allem im Rahmen finiter Deformationen (Option NLGEOM) — zwingend erforderlich ist, stellt für die Formulierung und spätere Implementierung von (eigenen) Materialmodellen eine besondere Herausforderung dar.

Aus diesem Grund wird hier eine Möglichkeit zur *numerischen Bestimmung* des Moduls beschrieben, die auch in TSAKMAKIS & WILLUWEIT [2003] beschrieben und inzwischen für eine effiziente Verwendung weiterentwickelt worden ist, siehe BAASER [2006].

Hier betrachten wir die Gedanken dieser Umsetzung für ABAQUS. Eine grundlegende Diskussion zur numerischen Bestimmung von Materialmoduli findet sich auch schon in MIEHE [1996].

ABAQUS stellt in der UMAT-Schnittstelle den Deformationsgradient $\mathbf{F}_1 = \mathbf{F}_{t+\Delta t}$ zum aktuellen Zeitpunkt $t+\Delta t$ und den Deformationsgradient $\mathbf{F}_0 = \mathbf{F}_t$ zum letzten, konvergierten Iterationsschritt t zur Verfügung. Aufgabe ist nun, neben dem aktuellen Spannungszustand $\boldsymbol{\sigma}_{t+\Delta t}$ auch den entsprechenden Modul

$$\mathbb{D}_{t+\Delta t} = \frac{1}{J} \frac{\partial \Delta \boldsymbol{\tau}_{t+\Delta t}}{\partial \Delta \boldsymbol{\varepsilon}_{t+\Delta t}} \quad (10.2)$$

zu bestimmen. Dabei ist

$$\Delta \boldsymbol{\varepsilon}_{t+\Delta t} := \mathbf{ln}[\Delta \mathbf{v}_{t+\Delta t}] \quad (10.3)$$

definiert als der natürliche Logarithmus der Rate des aktuellen *Linksstrecktensors*, siehe (??) und (??).

Idee dieser Umsetzung ist nun, aus $\mathbf{F}_{t+\Delta t}$ die Rate $\Delta \boldsymbol{\varepsilon}_{t+\Delta t}$ zu bestimmen, diese *numerisch zu stören*, um daraus eine Störung $\mathbf{F}_{t+\Delta t}^*$ für den aktuellen Deformationsgradienten zu erhalten. Damit lassen sich *gestörte Spannungsantworten* $\boldsymbol{\tau}_{t+\Delta t}^*$ berechnen, die dann eine konsistente Bestimmung des Moduls ergeben.

Algorithmus zur Bestimmung von $\mathbf{F}_{t+\Delta t}^*$:

$$\Delta \mathbf{F} = \mathbf{F}_1 \cdot \mathbf{F}_0^{-1} \quad (10.4)$$

$$\Delta \mathbf{v} \text{ aus } \Delta \mathbf{F} \quad (\text{polare Zerlegung, } \Delta \mathbf{R} \text{ speichern}) \quad (10.5)$$

$$\Delta \boldsymbol{\varepsilon} = \ln[\Delta \mathbf{v}] \quad (10.6)$$

$$\Delta \boldsymbol{\varepsilon}^* \text{ als Störung von } \Delta \boldsymbol{\varepsilon} \quad (10.7)$$

$$\Delta \mathbf{v}^* = \exp[\Delta \boldsymbol{\varepsilon}^*] \quad (10.8)$$

$$\Delta \mathbf{F}^* = \Delta \mathbf{v}^* \cdot \Delta \mathbf{R} \quad (10.9)$$

$$\mathbf{F}_1^* = \Delta \mathbf{F}^* \cdot \mathbf{F}_0 \quad (10.10)$$

Zur Bestimmung der Exponentialfunktion eines Tensors, siehe BAASER [2004] und Referenzen darin.

Teil III
Übungsaufgaben

Kapitel 11

Übungsaufgaben

11.1 Berechnung von Deformationsmaßen

$$\mathbf{b} = \mathbf{F} \cdot \mathbf{F}^T$$

11.1.1 Hauptachsenzerlegung für *simple shear*

$$\mathbf{F} = \begin{bmatrix} 1 & s & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ mit } s = \tan \gamma, \text{ siehe Abb. 5.4}$$

- Berechnen Sie analytisch die Hauptstreckungen $\lambda_i = \lambda_i(\gamma)$ dieser Deformation
- siehe auch: ABAQUS-Hilfsroutinen `SINV`, `SPRINC` und `SPRIND`

11.2 Einaxiale Darstellung und Ableitung des YEOH-Modells

Geben Sie das hyperelastische Materialmodell nach YEOH mit U_{YeoH} nach (6.24) in einer einaxialen Darstellung für den 1. PIOLA-KIRCHHOFF-Tensor P_{11} als Funktion der Streckung $\lambda = \lambda_{1ax}$ an. Dabei gelten $\bar{I}_1 = \bar{\lambda}_1^2 + \bar{\lambda}_2^2 + \bar{\lambda}_3^2$ und $\bar{\lambda}_i = J^{-\frac{1}{3}} \lambda_i$, $J = \det \mathbf{F}$.

Mögliches Vorgehen:

- Aus (6.11) oder (6.18): Darstellung für \mathbf{S} bzw. $\boldsymbol{\sigma}$
- Bedingung: „einaxial“ $\rightarrow p$
- $\leadsto S_{11}, \sigma_{11}, P_{11}$

Anmerkung:

Im ABAQUS-Manual gibt es ein Kapitel zur Parameter-Anpassung bei hyperelastischen Werkstoffmodellen.

11.3 Parameter-Anpassung

— Fehler-Quadrat-Minimierung

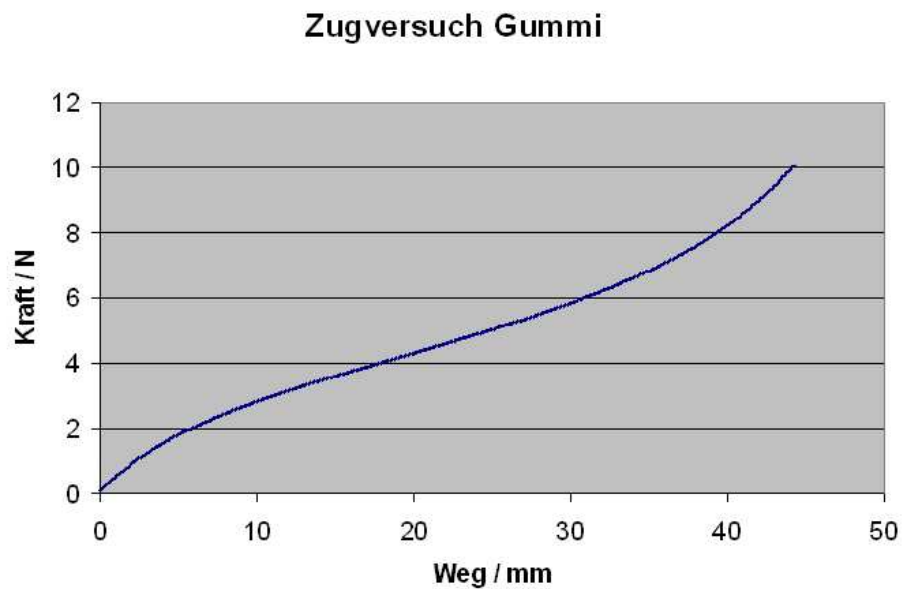


Abbildung 11.1: Einaxialer Zugversuch an Gummi-Probe mit $l_0 = 30$ mm und $A_0 = 4$ mm²

- Anpassen dieser Messdaten das neo-HOOKE- und das YEOH-Werkstoffmodell, siehe Abschn. 9.3
- Welche Parameter liefert eine Anpassung durch ABAQUS/CAE ?

11.4 Anwendung von ABAQUS CAE

11.4.1 Simulation einer axialsymmetrischen Struktur

Wir betrachten den in Abb. 11.2 dargestellten Querschnitt eines Radialwellendichtrings (RWDR).

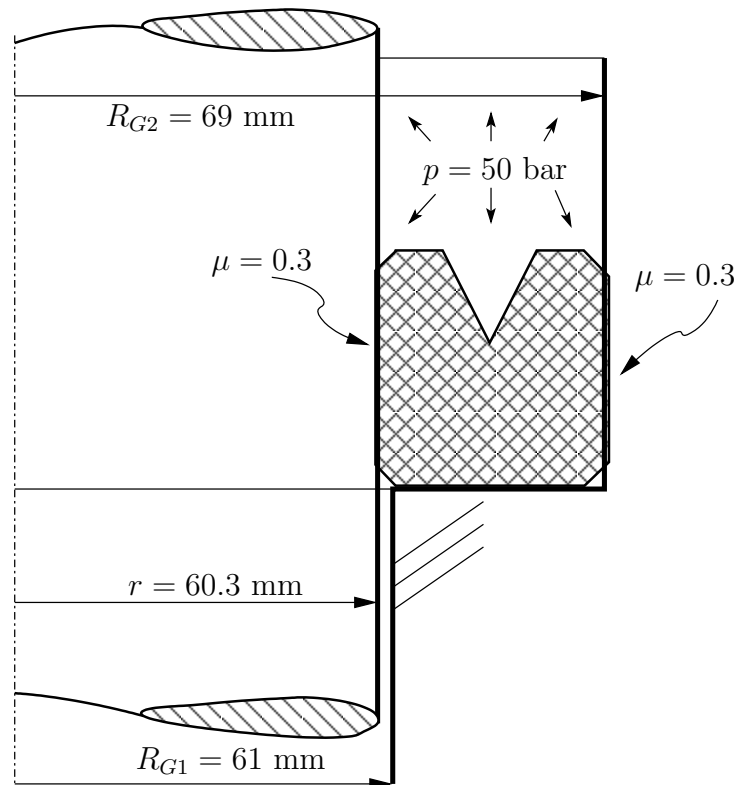


Abbildung 11.2: Querschnitt eines RWDR

Aufgabenstellung

- Einbau, Verpressung (Reibkoeffizient zwischen Gehäuse, Welle und Dichtungsring μ) und Temperaturerhöhung (Wärmeausdehnungskoeffizient $\alpha = 1.8 \cdot 10^{-4} \text{ K}^{-1}$)
- Druckbelastung mit p .
Hinweis: ABAQUS-Befehl *PRESSURE PENETRATION
- Konvergenzuntersuchung mit mindestens zwei unterschiedlichen Diskretisierungen

- Vergleich Materialmodelle neo-HOOKE und YEOH mit Materialdaten aus Abschnitt 11.3
Bestimmung von Betrag und Ort der maximalen Hauptspannung im Querschnitt
- Reaktionskräfte *radial* für unterschiedliche Modellierungen

11.4.2 HERTZsche Pressung — Linienkontakt — $a = l$

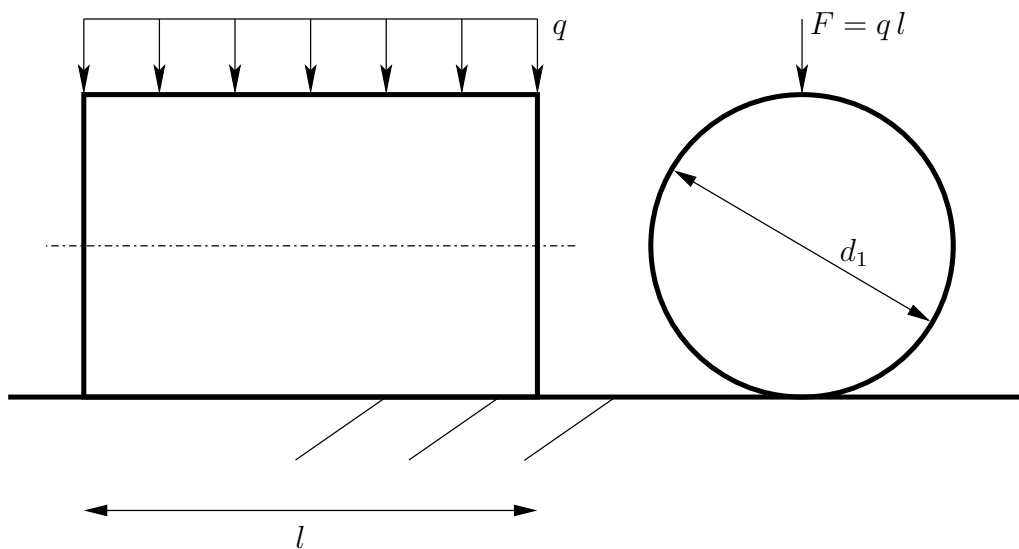


Abbildung 11.3: HERTZsche Pressung — Linienkontakt

Aufgabenstellung

- $r = \frac{d_1}{2} = 25 \text{ mm}$, $l = 10 \text{ mm}$
- Breite der (rechteckigen) Kontaktfläche $A_{\text{Kontakt}} = a b$.
Vergleich mit $b = \sqrt{\frac{4F(1-\nu^2)d_1}{\pi E l}}$

11.4.3 Drei-Punkt-Biegung — Plastische Zone

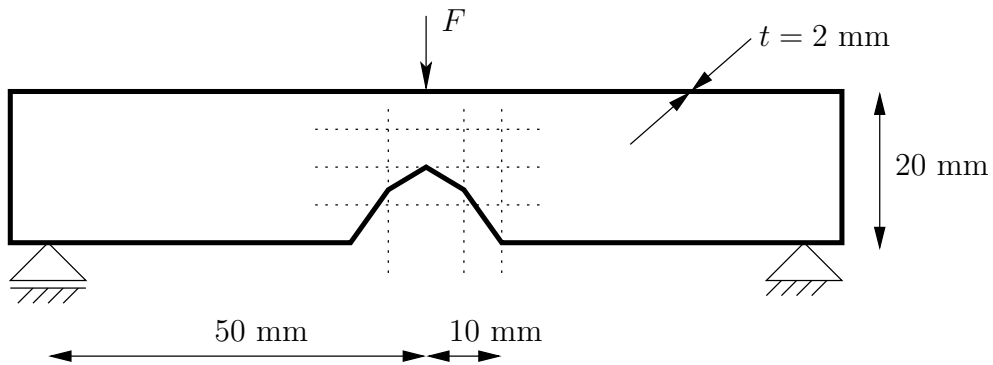


Abbildung 11.4: Drei-Punkt-Biegung an Stahlprobe

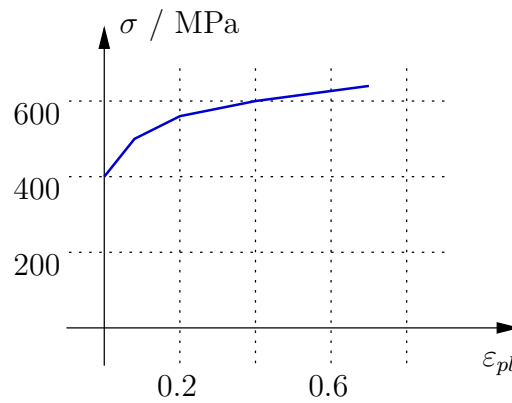


Abbildung 11.5: (isotrope) Materialverfestigung

Aufgabenstellung

- Ausbildung der plastischen Zone (siehe PEEQ als Kontour-Darstellung) abhängig von F ?

11.5 Programmierbeispiel

11.5.1 YEOH in UHYPER

Beispiel: “Original-Quelltext“ aus ABAQUS–Manual

```
      SUBROUTINE UHYPER(BI1,BI2,AJ,U,UI1,UI2,UI3,TEMP,NOEL,CMNAME,
$              INCMPLAG,NUMSTATEV,STATEV,NUMFIELDV,
$              FIELDV,FIELDVINC,NUMPROPS,PROPS)
C
      INCLUDE 'ABA_PARAM.INC'
C
      CHARACTER*80 CMNAME
      DIMENSION UI1(3),UI2(6),UI3(6),STATEV(*),FIELDV(*),
$              FIELDVINC(*),PROPS(*)
C
      PARAMETER (ZERO=0.0D0,ONE=1.0D0, TWO=2.0D0, THREE=3.0D0)
C
      C10 = PROPS(1)
      C01 = PROPS(2)
      D1  = PROPS(3)
      ...
      RETURN
      END
C
C *** ENDE ***
```

Aufgabenstellung

- Vergleich der Ergebnisse zu Beispiel 11.4.1 mit Parametern aus 11.3.
- Konvergiert diese Lösung mit der gleichen Geschwindigkeit wie die Lösung aus 11.4.1 ? Vergleich der Iterationen in *.sta.

Anhang A

Fortran77–Beispiel

Übersetzen / Kompilieren mit `g77 -O3 -o testprog testprog.f`

c (c) HBaa, August 2008: Beispiel eines F77-Programms

```
program testprog
c      1      2      3      4      5      6      7
c23456789#123456789#123456789#123456789#123456789#123456789#12

c... Variablendeklaration
      implicit none
      integer i
      real*8 a1, erg, feld(10) ! "real*8" entspricht "double precision"

c... Schleife
      do i=1,10
         feld(i) = dble(i)
      enddo
      a1 = 4.23d0 ! "double precision"-Zuweisung

c... Aufruf einer "subroutine"
      call quadrat(a1,erg)
      print*, 'Ergebnis: ', erg, feld(4)
      if (erg.gt.10.0d0) then
         print*, 'Ergebnis ist groesser als 10.0 !'
      endif

      end ! program
c
      subroutine quadrat(a,b)
      implicit none
      real*8 a,b

      b = a**2

      return
      end
c
c *** ENDE ***
```


Anhang B

Aufstellung einiger Linux-/Unix-Befehle

- `man [Befehl]` – Hilfe zu [Befehl]
- `dir` – Verzeichnisinhalt anzeigen
- `mkdir [name]` – Verzeichnis erstellen
- `cd [name]` – Wechsel in dieses Verzeichnis
- `cd ..` – in Verzeichnis nach oben wechseln
- `cp [name_1] [name_2]` – Datei kopieren
- `mv [name_1] [name_2]` – Datei umbenennen

Literaturverzeichnis

- ALTENBACH, J. & ALTENBACH, H. (1994). *Einführung in die Kontinuumsmechanik*. Teubner.
- ARAVAS, N. (1987). ‘On the numerical integration of a class of pressure-dependent plasticity models’. *Int. J. Num. Meth. Eng.* **24**, 1395–1416.
- BAASER, H. (2004). ‘The PADÉ–approximation for matrix exponentials applied to an integration algorithm preserving plastic incompressibility’. *Comp. Mechanics* **34**(3), 237–245.
- BAASER, H. (2006). A Material Model Representing Inelasticity of Elastomers. In ‘NAFEMS Materialmodellierung’. Wiesbaden.
- BAASER, H. (2010a). *Development and Application of the Finite Element Method based on MATLAB*. Springer.
- BAASER, H. (2010b). ‘FEM–Simulation von Elastomerbauteilen’. *ATZ – Automobiltechnische Zeitschrift*.
- BATHE, K.-J. (1996). *Finite Element Procedures in Engineering Analysis*. Prentice–Hall, Inc. .
- BECKER, E. & BÜRGER, W. (1975). *Kontinuumsmechanik*. Teubner, Stuttgart.
- BRAESS, D. (1997). *Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie*. Springer.
- CHU, C.C. & NEEDLEMAN, A. (1980). ‘Void nucleation effects in biaxially stretched sheets’. *J. Eng. Mat. Tech.* **102**, 249–256.
- GURSON, A.L. (1977). ‘Continuum theory of ductile rupture by void nucleation and growth: Part I - yield criteria and flow rules for porous ductile media’. *J. Eng. Mat. Tech.* **99**, 2–15.

- HOLZAPFEL, G.A. (2000). *Nonlinear Solid Mechanics*. Wiley.
- HUGHES, T.J.R. (2000). *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications.
- INA (2002). *Technisches Taschenbuch*. 7., veränderter Nachdruck edn. Schaeffler KG.
- ITSKOV, M. (2009). *Tensor Algebra and Tensor Analysis for Engineers: With Applications to Continuum Mechanics*. 2 edn. Springer.
- KAHN, A.S. & HUANG, S. (1995). *Continuum theory of plasticity*. Wiley.
- LEMAITRE, J. & CHABOCHE, J.-L. (1990). *Mechanics of Solid Materials*. Cambridge University Press.
- MIEHE, C. (1994). ‘Aspects of the formulation and finite element implementation of large strain isotropic elasticity’. *Int. J. Num. Meth. Eng.* **37**, 1981–2004.
- MIEHE, C. (1996). ‘Numerical Computation of Algorithmic (Consistent) Tangent Moduli in Large-Strain Computational Inelasticity’. *Comp. Meth. Appl. Mech. Eng.* **134**, 223–240.
- REESE, S. & WRIGGERS, P. (1997). ‘A material model for rubber-like polymers exhibiting plastic deformation: computational aspects and a comparison with experimental results’. *Comp. Meth. Appl. Mech. Eng.* **148**, 279–298.
- ROUSSELIER, G. , DEVAUX, J.-C. , MOTTET, G. & DEVESA, G. (1989). ‘A methodology for ductile fracture analysis based on damage mechanics: An illustration of a local approach of fracture’. *Nonlinear Fracture Mechanics* **2**, 332–354.
- SIMO, J.C. (1992). ‘Algorithms for static and dynamic multiplicative plasticity that preserve the classical return mapping schemes of the infinitesimal theory’. *Comp. Meth. Appl. Mech. Eng.* **99**, 61–112.
- SIMO, J.C. & HUGHES, T.J.R. (1998). *Computational Inelasticity*. Springer. New-York.
- TRELOAR, L.R.G. (1975). *The Physics of Rubber Elasticity*. 3th edn. Oxford University Press.

- TSAKMAKIS, CH. & WILLUWEIT, A. (2003). Use of the elastic predictor–plastic corrector method for integrating finite deformation plasticity laws. In K. HUTTER & H. BAASER (Eds.). ‘Deformation and Failure in Metallic Materials’. Springer.
- TVERGAARD, V. (1989). ‘Material failure by void growth to coalescence’. *Advances in Appl. Mech.* **27**, 83–151.
- WEISS, A. & BONNER, S. (2008). *Generation Doof: Wie blöd sind wir eigentlich?*. Bastei Lübbe.
- WRIGGERS, P. (2001). *Nichtlineare Finite-Element-Methoden*. Springer.