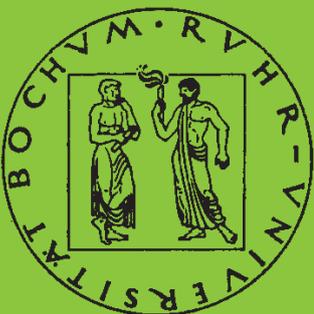


Mitteilungen aus dem Institut für Mechanik

Olaf Schilling

**Über eine implizite Partikelmethode
zur Simulation von Umformprozessen**

Heft Nr. 139



RUHR-UNIVERSITÄT BOCHUM

RUHR-UNIVERSITÄT BOCHUM
Institut für Mechanik

Olaf Schilling

**Über eine implizite Partikelmethode zur Simulation von
Umformprozessen**

Mitteilungen aus dem Institut für Mechanik Nr. 139

Herausgeber:

Institut für Mechanik
— Schriftenreihe —
Ruhr-Universität Bochum
D-44780 Bochum

ISBN 978-3-935892-14-8

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

©2007 Institut für Mechanik der Ruhr-Universität Bochum

Printed in Germany



Über eine implizite Partikelmethode zur Simulation von Umformprozessen

Vorgelegte
Dissertation

zur

Erlangung des Grades
Doktor-Ingenieur (Dr.-Ing.)

der

Fakultät für Bauingenieurwesen der
Ruhr-Universität Bochum

von

Olaf Schilling

Bochum, im Oktober 2005

Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als Wissenschaftlicher Mitarbeiter in der Arbeitsgruppe *Numerische Mechanik und Simulationstechnik* am Institut für Mechanik der Ruhr-Universität Bochum.

Für die Anregung zu dieser Arbeit, die Betreuung und die Übernahme des Hauptreferats danke ich Frau Prof. Dr.-Ing. Reese herzlich. Herrn Prof. Dr.-Ing. Hartmann danke ich für sein Interesse und seine freundliche Bereitschaft zur Übernahme des Koreferats.

Bei allen Mitarbeitern der Arbeitsgruppe bedanke ich mich für das angenehme und produktive Arbeitsklima.

Mein besonderer Dank gilt meiner Familie für Ihre Unterstützung.

Bochum, 21.10.2005

Olaf Schilling

Dissertation eingereicht am: 21.10.2005

Tag der mündlichen Prüfung: 27.02.2006

Erster Referent: Prof. Dr.-Ing. S. Reese

Zweiter Referent: Prof. Dr.-Ing. D. Hartmann

Übersicht

Die Eignung einer Gruppe netzfreier Methoden für die numerische Simulation von Umformprozessen wird untersucht. Den Methoden ist gemeinsam, dass Körper durch Lagrangesche Partikel diskretisiert werden, denen Zustands- und Geschichtsvariable angeheftet sind. Die schwache Form des Impulssatzes wird auf temporären Netzen oder ohne Netz gelöst. Wie bei der FEM sind die Ansätze lokal. Materialgesetze und das implizite Zeitschrittverfahren werden wie in der Finite-Elemente-Methode implementiert. Einer gesonderten Behandlung bedürfen die wesentlichen Randbedingungen und Kontaktbedingungen. Da die Eigenschaften der Ansatzfunktionen der meisten netzfreien Methoden einen direkten Einbau dieser Bedingungen nicht erlauben, werden diese über ein Strafverfahren erzwungen. Die Gemeinsamkeiten zwischen der untersuchten Gruppe netzfreier Methoden und der Finite-Elemente-Methode überwiegen die Unterschiede, sodass alle Methoden effizient innerhalb eines Programmsystems umgesetzt werden können. Abschließend wird das Programmsystem anhand einiger numerischer Beispiele aus dem Bereich der Umformsimulation überprüft und ein Vergleich zwischen einer vollständig netzfreien Methode und der Finite-Elemente-Methode vorgenommen.

Abstract

The applicability of a group of meshfree methods for the numerical simulation of forming processes is investigated. The common attribute of the used methods is that bodies are discretised by Lagrangian particles which carry all information. The differential equations in their weak form are solved on temporary meshes built of standard elements or without mesh. Like in the FEM the shape functions are defined locally. Material laws and the implicit time integration scheme are implemented in the usual way. Essential boundary conditions and contact conditions need a special treatment. In most meshfree methods the direct introduction of inhomogeneous displacement conditions is not possible due to the properties of the shape functions. Essential boundary conditions and contact conditions are enforced using a penalty method. The similarities between the group of meshfree methods and the FEM outbalance the differences so that all methods can be implemented efficiently in one programme system. Finally the programme system is proved on the basis of some numerical examples which deal with forming processes and a direct comparison between a completely meshfree method and the FEM is drawn.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Gliederung	2
1.3	Notation	3
1.3.1	Tensorschreibweise	3
1.3.2	Matrizenschreibweise	4
1.3.3	Formelzeichen	5
2	Netzfrie Methoden	7
2.1	Definition	7
2.2	Geschichtliche Entwicklung	8
2.2.1	GFDM, DEM, EFG und FPM	8
2.2.2	SPH, RKPM und MLSRK	10
2.2.3	Hp-Clouds und PUFEM	10
2.2.4	PIC, FLIP und MPM	11
2.3	Randbedingungen	12
3	Implementierte Methoden	15
3.1	Programmkonzept	15
3.2	Methodenüberblick	17
3.2.1	Methoden 1a (FEM) und 1b (FPM)	17

3.2.2	Methoden 2a (MPM) und 2b	18
3.2.3	Methoden 3a und 3b	19
3.3	Berechnungsablauf	20
3.4	Formelzusammenstellung für die Methoden 1a (FEM) und 1b (FPM)	21
4	Kontinuumsmechanische Grundgleichungen	25
4.1	Bilanzgleichungen	25
4.2	Kinematik	26
4.3	Materialgesetze	26
4.3.1	Rheologische Modelle	26
4.3.2	Viskoelastisches Materialverhalten	27
4.3.3	Elastoplastisches Materialverhalten	28
4.3.4	Allgemeine Form	31
4.4	Schwache Form der Bilanzgleichungen	31
4.5	Randbedingungen und Kontakt	32
5	Ansätze zur räumlichen Diskretisierung	39
5.1	Allgemeine Form	40
5.2	Elementansätze	41
5.2.1	Natürliche Elementkoordinaten	42
5.3	Punktzuordnung	45
5.3.1	Konvex polygonal berandete Elemente	45
5.3.2	Krummlinig berandete Elemente	46
5.4	Inverse Elementansätze	48
5.4.1	Herleitung über Pseudoinverse	48
5.4.2	Herleitung über <i>Partition of Unity</i> -Forderung	51
5.5	Ansätze für Punktnetze	52
5.5.1	Wichtungsfunktionen	52

5.5.2	Basisfunktionen	61
5.5.3	Approximation über Methode der bewegten kleinsten Fehlerquadrate	62
5.6	Vergleich der in der FPM und der FEM genutzten Ansätze	64
6	Diskretisierung, Linearisierung und Lösung	67
6.1	Übergang zur Matrizenschreibweise	67
6.2	Räumliche Diskretisierung der Bilanzgleichungen	70
6.3	Räumliche Diskretisierung der Kontaktoberflächen	73
6.3.1	Hermite-Interpolation	74
6.3.2	Kontaktlinien	78
6.3.3	Kontaktgeometrie	81
6.4	Zeitliche Diskretisierung	85
6.4.1	Zeitliche Diskretisierung der Entwicklungsgleichungen	86
6.4.2	Zeitliche Diskretisierung der Bilanzgleichungen	87
6.5	Linearisierung und Lösung	88
6.5.1	Linearisierung und Lösung der Materialgleichungen	88
6.5.2	Linearisierung und Lösung der Bilanzgleichungen	91
6.5.3	Erweiterung auf große Verformungen	94
7	Algorithmen	95
7.1	Lösung linearer Gleichungssysteme und linearer Ausgleichsprobleme	95
7.1.1	Lösung linearer Gleichungssysteme mit quadratischer und regulärer Koeffizientenmatrix	96
7.1.2	Lösung linearer Ausgleichsprobleme	96
7.1.3	Singulärwertzerlegung	98
7.1.4	Matrixkondition	99
7.2	Iterative Lösung nichtlinearer Gleichungssysteme	99
7.2.1	Newton-Raphson-Verfahren	100

7.2.2	Modifiziertes Newton-Raphson-Verfahren	101
7.2.3	Gauß-Newton-Verfahren	101
7.2.4	Diskretisiertes Newton-Verfahren	101
7.3	Zeitschrittverfahren	102
7.3.1	Zeitschrittverfahren erster Ordnung	102
7.3.2	Zeitschrittverfahren zweiter Ordnung	103
7.4	Zeitschrittsteuerung	104
8	Rechenregeln	107
8.1	Voigtsche Notation	107
8.2	Einheitensysteme	110
9	Beispiele	111
9.1	Übersicht	111
9.2	Zeitmessungen	112
9.3	Quadratscheibe unter Teillast	113
9.3.1	Wichtungsfunktionsausdehnungen	115
9.3.2	Konvergenzverhalten FPM / FEM	122
9.3.3	Zeitmessungen	124
9.4	Extrusion	129
9.5	Tiefziehen	136
9.6	Aufprall	142
10	Zusammenfassung	149
A	Eingabedatei	151
A.1	Dateiformat	151
A.2	Einteilung der Befehle	153
A.3	Syntax der Befehlsbeschreibungen	153

A.4	Befehlsbeschreibungen	154
A.4.1	Befehl <code>titel</code>	154
A.4.2	Befehl <code>aus_dat_gid</code>	154
A.4.3	Befehl <code>aus_dat_gnupl</code>	155
A.4.4	Befehl <code>obj_ausdat</code>	156
A.4.5	Befehl <code>stw_berechverf</code>	156
A.4.6	Befehl <code>stw_dimension</code>	157
A.4.7	Befehl <code>stw_zeitschrittverf</code>	158
A.4.8	Befehl <code>stw_m</code>	158
A.4.9	Befehl <code>stw_strafverf</code>	159
A.4.10	Befehl <code>stw_kontakt</code>	159
A.4.11	Befehl <code>stw_nummattangente</code>	160
A.4.12	Befehl <code>stw_wolken</code>	160
A.4.13	Befehl <code>stw_bneuberech</code>	161
A.4.14	Befehl <code>stw_uebtragrichtstart</code>	162
A.4.15	Befehl <code>stw_numplus</code>	162
A.4.16	Befehl <code>stw_t</code>	163
A.4.17	Befehl <code>stw_zstrg</code>	164
A.4.18	Befehl <code>stw_t_aus_dat</code>	164
A.4.19	Befehl <code>stw_t_aus_ktr</code>	165
A.4.20	Befehl <code>stw_autodtparam</code>	166
A.4.21	Befehl <code>stw_gleichloes_bil</code>	166
A.4.22	Befehl <code>stw_gleichloes_mat</code>	167
A.4.23	Befehl <code>stw_gleichloes_ans</code>	167
A.4.24	Befehl <code>stw_bilanzit</code>	168
A.4.25	Befehl <code>stw_matit</code>	169
A.4.26	Befehl <code>stw_schrankefliess</code>	169

A.4.27 Befehl <code>mat</code>	170
A.4.28 Befehl <code>mpu</code>	171
A.4.29 Befehl <code>kno</code>	172
A.4.30 Befehl <code>eletyp</code>	173
A.4.31 Befehl <code>ele</code>	175
A.4.32 Befehl <code>ele_c</code>	175
A.4.33 Befehl <code>pu_u</code>	175
A.4.34 Befehl <code>pu_ut</code>	176
A.4.35 Befehl <code>pu_utt</code>	176
A.4.36 Befehl <code>pu_z</code>	177
A.4.37 Befehl <code>pu_c</code>	177
A.4.38 Befehl <code>punktlast</code>	178
A.4.39 Befehl <code>konlin</code>	179
Abbildungsverzeichnis	181
Tabellenverzeichnis	185
Literaturverzeichnis	187

Kapitel 1

Einleitung

1.1 Motivation

Ziel ist die Simulation von Umformprozessen. Beispiele für kontinuierliche und diskontinuierliche Umformprozesse zeigt die Abbildung 1.1.

Insbesondere die Simulation von kontinuierlichen Umformprozessen ist aufwändig, da große Bereiche zu diskretisieren sind und gleichzeitig in kritischen Bereichen eine feine Diskretisierung notwendig ist. Außerdem sind teils längere Zeiträume zu berechnen, und die Prozesse können hochdynamisch sein.

Netzfrequenz Methoden bieten sich als mögliche Lösung an, da Vernetzungen und ggf. Neuvernetzungen weniger aufwändig sind als in der FEM, h-Adaptivität einfach implementiert werden kann, die Punktnetze wenig empfindlich hinsichtlich Netzdegeneration sind und schließlich Unstetigkeiten einfacher als in der FEM repräsentiert werden können.

Da kein Programmquellcode aus dem Bereich netzfreier Methoden zur Verfügung stand, war eine Eigenentwicklung notwendig. Ziel bei der Gestaltung des Programmsystems war es, eine möglichst hohe Flexibilität zu erhalten. Das Programmsystem soll als Testumgebung zur Überprüfung und Entwicklung von Methoden dienen.

Einige Abstraktionen machten es möglich, eine Gruppe netzfreier Methoden gemeinsam mit der Finite-Elemente-Methode zu implementieren. Die gemeinsame Implementierung bot den Vorteil, dass der Teil des Quellcodes, der von allen Methoden genutzt wird, mit der Finite-Elemente-Methode überprüft werden konnte, deren Verhalten bekannt ist. Außerdem ist ein direkter Vergleich der Methoden hinsichtlich Genauigkeit und Geschwindigkeit möglich.

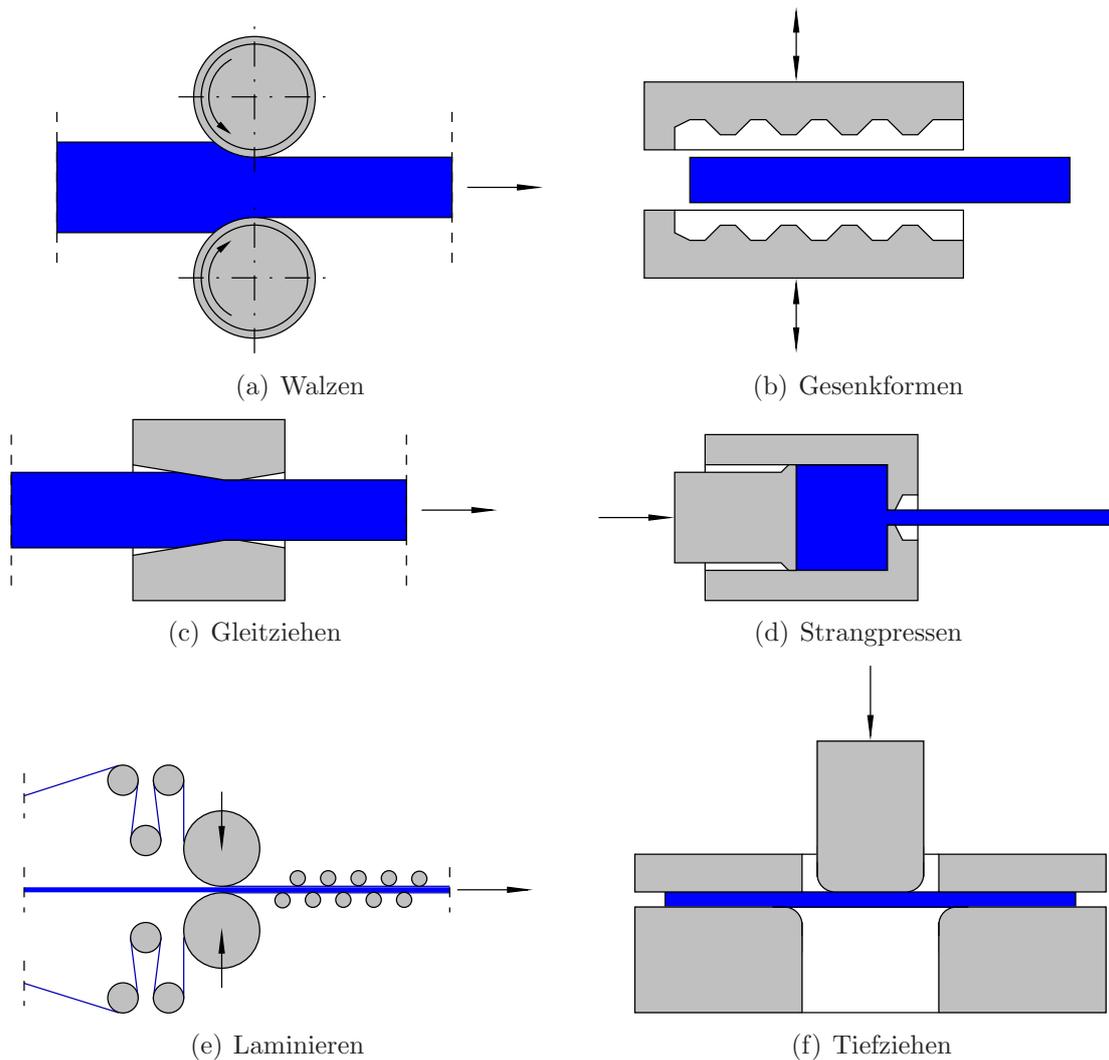


Abbildung 1.1: Beispiele für kontinuierliche (a), (c), (e) und diskontinuierliche (b), (d), (f) Umformprozesse

Die in Umformprozessen eingesetzten Werkzeuge sind in der Regel infolge der Materialwahl und ihrer konstruktiven Ausbildung wesentlich steifer als die umzuformenden Körper. Es ist daher in guter Näherung möglich, die Werkzeuge als starre Körper bzw. ihre Oberflächen als starre Kontaktflächen zu beschreiben.

Um auch gekrümmte Werkzeugoberflächen einfach und zutreffend diskretisieren zu können, werden C^1 -stetige, auf einer Hermite-Interpolation basierende, Kontaktelemente verwendet.

1.2 Gliederung

Die vorliegende Arbeit ist wie folgt gegliedert: Das 2. Kapitel liefert zunächst einen Überblick zu netzfreien Methoden. Eine Beschreibung der implementierten Metho-

den erfolgt im 3. Kapitel. Die benötigten kontinuumsmechanischen Grundgleichungen werden im 4. Kapitel zusammengestellt. Da die Ansätze zur räumlichen Diskretisierung wesentlich die Berechnungsmethoden charakterisieren, ist diesen Ansätzen ein eigenes, das 5. Kapitel, gewidmet. Die Grundgleichungen werden im 6. Kapitel räumlich und zeitlich diskretisiert, linearisiert und gelöst. In Kapitel 7 und 8 sind Algorithmen und Rechenregeln beschrieben, auf die im vorderen Teil der Arbeit Bezug genommen wird. Es folgen im 9. Kapitel die numerischen Beispiele zur Verifikation des Programmcodes. Anhand der in den Eingabedateien möglichen Befehle wird im Anhang A der Funktionsumfang des Programmsystems dargestellt.

1.3 Notation

Die kontinuumsmechanischen Grundgleichungen in Kapitel 4 werden in Tensor-schreibweise angegeben. Im Kapitel 5 wird zur Matrizenschreibweise übergegangen, die für die restliche Arbeit beibehalten wird.

Soweit wie möglich sind die Bestimmungsgleichungen so formuliert, dass sie unabhängig von der räumlichen Dimension ($N_D = 1, 2$ oder 3) gültig sind. Werden Matrizengleichungen ausgeschrieben, so erfolgt dies in der Regel für den dreidimensionalen Fall, da die Beziehungen für den ein- oder zweidimensionalen Fall einfach durch Streichen von Zeilen und Spalten erhalten werden können. Der C^1 -stetige Kontakt lässt sich im Zweidimensionalen einfacher formulieren und darstellen. Daher beschränkt sich das Programmsystem zur Untersuchung von Partikelmethoden zunächst auf zwei Dimensionen.

1.3.1 Tensorschreibweise

Skalare werden durch schräge Buchstaben und Vektoren und Tensoren zweiter Stufe durch gerade fette Buchstaben dargestellt. Bei Tensoren höherer Stufe wird die Stufe angegeben.

s	Skalar
\mathbf{u}	Vektor
\mathbf{A}	Tensor 2. Stufe
$\mathcal{C}^{(4)}$	Tensor 4. Stufe

Groß- und Kleinschreibung und die Wahl griechischer Buchstaben richten sich nicht nach der Stufe tensorieller Größen, sondern nach dem üblichen Gebrauch.

Bei skalaren Produkten gibt die Punktanzahl die Anzahl der Basisvektorüberschiebungen an. In der Literatur findet man unterschiedliche Definitionsvarianten der mehrfachen skalaren Produkte. Entweder werden die Basisvektoren der am Produkt beteiligten Tensoren „über Kreuz“ oder „von innen nach aussen“ überschoben.¹ In dieser Arbeit wird die erste Definitionsvariante (Überschiebung der Basisvektoren „über Kreuz“) verwendet, da sie weiter verbreitet ist

1.3.2 Matrizenschreibweise

Skalare werden durch schräge Buchstaben und Vektoren und Matrizen durch schräge fette Buchstaben dargestellt. Da sich in der Matrizenrechnung die Matrixdimensionen nicht zwangsläufig wie in der Tensorrechnung aus der Stufe und der räumlichen Dimension ergeben, werden die Matrixdimensionen meist zusätzlich angegeben.

s	Skalar
\mathbf{u} <small>(N)</small>	Vektor
\mathbf{A} <small>(N×M)</small>	Matrix

Es wird nur ein einziges Produkt, das *Matrizenprodukt*, verwendet, das sich über ein einfaches Rechenschema ausführen lässt. Als Operanden für das Matrizenprodukt kommen ausschließlich Vektoren und maximal zweidimensionale Matrizen in Frage. Zwischen Spalten- und Zeilenvektoren ist zu unterscheiden. Ist ein Vektor nicht gekennzeichnet, so handelt es sich um einen Spaltenvektor.²

Auf diese Art und Weise kann mit Hilfe des Matrizenproduktes sowohl ein „dyadisches“ Produkt

$$\mathbf{u} \mathbf{v}^T = \mathbf{A} \quad (1.1)$$

(N) (M) (N×M)

als auch ein „skalares“ Produkt

$$\mathbf{u}^T \mathbf{w} = a \quad (1.2)$$

(N) (N)

ausgedrückt werden.

¹Die erste Variante liefert im Vergleich zur zweiten unübersichtlichere Rechenregeln für gemischte Produkte aber einfachere Ableitungsregeln.

²Diese Art der Matrizenschreibweise erscheint etwas „altmodisch“, wird aber verwendet, da sie weit verbreitet ist.

Sobald in einem tensoriellen Ausdruck Tensoren von höherer als zweiter Stufe auftreten, muss bei dieser Variante der Matrixschreibweise in der dazugehörigen Matrixformulierung auf eine Hypermatrixdarstellung oder, wenn Symmetrieeigenschaften das erlauben, auf die Voigtsche Notation ausgewichen werden.

1.3.3 Formelzeichen

Gleiche Symbole kennzeichnen gleiche physikalische Größen und nicht gleiche Funktionen. Wenn bei der gleichen physikalischen Größe zwischen unterschiedlichen Funktionen unterschieden werden muss (z.B. bei Ableitungsprozeduren), werden zur Kennzeichnung alle Funktionsparameter explizit angegeben.

Alle verwendeten Formelzeichen werden an der Stelle ihres ersten Auftretens erklärt. Einen zusätzlichen Überblick gibt die folgende Aufstellung.

Dimensionen

Für Dimensionsangaben wird der Buchstabe N in serifenloser Schrift verwendet.

N_D	räumliche Dimension
N_V	Dimension der Voigtschen Vektoren
N_B	Basisfunktionsanzahl
N_G	Polynomgrad
N_E	Anzahl Elemente
N_K	Anzahl Knoten
N_P	Anzahl Punkte
N_{RP}	Anzahl Randpunkte
N_{MP}	Anzahl Materialpunkte
N_{IP}	Anzahl Integrationspunkte

Indizes

Für Laufindizes werden Kleinbuchstaben ebenfalls in serifenloser Schrift verwendet. Wenn möglich werden für bestimmte Zwecke einzelne Buchstaben bevorzugt verwendet.³

³Mathematisch gesehen besteht kein Unterschied zur Wahl beliebiger anderer Buchstaben. Die Darstellung gewinnt aber an Übersichtlichkeit.

$(\cdot)^k$	Materialiterationsindex (oben)
$(\cdot)^j$	Bilanziterationsindex (oben)
$(\cdot)_n$	Zeitschrittindex (unten)
$(\cdot)_p$	Punktindex
$(\cdot)_k$	Knotenindex
$(\cdot)_e$	Elementindex
$(\cdot)_d$	Dimensionsindex

Summation und Assemblierung

$\sum_{p=1}^{N_P}(\cdot)$	Summation (es wird keine Summenkonvention verwendet)
$A_{p=1}^{N_P}(\cdot)$	Assemblierung (im Gegensatz zur Summation sind Größen nicht nur zu addieren, sondern lokale Größen sind an den richtigen Stellen globalen Größen hinzuzufügen)

Symbole über Größen

$\tilde{(\cdot)}$	volumenbezogene Größe
$\hat{(\cdot)}$	deviatorischer Anteil
$\bar{(\cdot)}$	interpolierte oder approximierte Größe
$\overset{v}{(\cdot)}$	Voigtsche Notation

Kapitel 2

Netzfrie Methoden

Zusammenfassende Darstellungen und Klassifizierungen netzfreier Methoden finden sich u. a. in DUARTE & ODEN (1995a), BELYTSCHKO ET AL. (1996), GRIEBEL & SCHWEITZER (2002b) und FRIES & MATTHIES (2004).

In diesem Kapitel werden zunächst mögliche Definitionen des Begriffes der *Netzfrieheit* angegeben. Es folgt ein Überblick zu der geschichtlichen Entwicklung netzfreier Methoden. Abschließend werden Verfahren aufgeführt, mit denen sich wesentliche Randbedingungen in netzfreien Methoden berücksichtigen lassen.

2.1 Definition

DUARTE & ODEN (1995a) betrachten Methoden als netzfrei, wenn die diskreten Gleichungen des Randwertproblems nicht von einem eindeutig definierten (Element-) Netz abhängen. Eine schwache Abhängigkeit von einem Hintergrundnetz zur numerischen Integration wird zugelassen, sofern keine festen Knotenzusammenhänge genutzt werden.

Eine ähnliche Definition findet sich in ONATE ET AL. (1996a), wonach eine netzfreie Methode folgende Bedingungen erfüllen sollte: Die Ansatzfunktionen und Ansatzfunktionsableitungen dürfen nur durch Punktpositionen und an den Punkten gegebenen Parametern definiert sein, und die Wichtungsfunktionen und ihre Ableitungen dürfen nur von den Punktpositionen abhängen. Es wird keine (explizite) Volumen- oder Oberflächenintegration benötigt, oder eine Volumen- oder Oberflächenintegration ist unabhängig von der gewählten Interpolationsmethode. Die letzte Bedingung bedeutet, dass Hilfsnetze zur numerischen Integration zulässig sind.

Häufig werden in der Literatur abweichend von den obigen Definitionen Methoden auch dann noch als netzfrei bezeichnet, wenn temporäre Hintergrundnetze zur Bildung der Ansatzfunktionen genutzt werden.

2.2 Geschichtliche Entwicklung

Es werden im Folgenden auch Methoden beschrieben, die nach den obigen Definitionen nicht netzfrei sind, sofern ihre Entwicklung im unmittelbaren Zusammenhang mit netzfreien Methoden erfolgte.

2.2.1 GFDM, DEM, EFG und FPM

Eine der ersten netzfreien Methoden beruht auf einer Abwandlung der *Finite Difference Method* (FDM). Der Anwendungsbereich der ursprünglichen Finite-Differenzen-Methode ist beschränkt, da die Knoten ein orthogonales Raster bilden. Erweiterungen auf Diskretisierungen mit unregelmäßig angeordneten Knoten wurden von PERRONE & KAO (1975), GIRAULT (1974), PAVLIN & PERONNE (1975), LISZKA & ORKISZ (1977,1980) LISZKA (1984) und SNELL ET AL. (1981) unter der Bezeichnung *Generalized Finite Difference Method* (GFDM) vorgenommen. Wie in der FDM werden Sterne aus den Knoten gebildet und FD-Approximationen über Taylorreihenentwicklungen gebildet. Im Unterschied zur FDM stehen die einen Stern bildenden Knoten nicht von vorneherein fest, sondern es werden mit geeigneten Algorithmen Knoten aus der Umgebung des zentralen Knotens ausgewählt.

KROK (1989) und NAY & UTKU (1972) setzten davon abweichend *Least Square* (LS) Approximationen an den Knoten ein. Die LS Approximation ist unstetig.

Kontinuierliche Funktionen werden mit der *Moving Least Squares* (MLS) Approximation erhalten, die ursprünglich von SHEPARD (1968), McLAIN (1974), GORDON & WIXOM (1978) und LANCASTER & SALKAUSKAS (1981) zur Oberflächenerzeugung aus Datenpunkten verwendet wurde. CLEVELAND (1993) liefert zu dieser Problemklasse eine ausführliche Darstellung.

Im Rahmen netzfreier Methoden wurde die MLS Approximation erstmals durch NAYROLES ET AL. (1992) in der *Diffuse Element Method* (DEM) eingesetzt. Die Ansatzfunktionswerte an den Integrationspunkten werden bei diesem Verfahren über eine MLS-Approximation berechnet. Bei der Bildung der Ableitungen der Ansatzfunktionen werden einzelne Terme vernachlässigt, was die Genauigkeit der Methode

beeinträchtigt. Die numerische Integration erfolgt auf einem die Knoten verbindenden Elementnetz oder ohne Elementnetz.

In der von BELYTSCHKO ET AL. (1994) und LU ET AL. (1994), beschriebenen *Element Free Galerkin Method* (EFG) erfolgt hingegen die numerische Integration stets über ein Elementnetz. Dabei bilden die Elemente ein gleichmäßiges Hintergrundnetz und werden als *Zellen* bezeichnet. Außerdem sind die Ableitungen der Approximationsfunktionen im Gegensatz zur DEM vollständig. Wesentlichen Randbedingungen werden über *Lagrangesche Multiplikatoren* eingebracht.

Die EFG wurde besonders von BELYTSCHKO ET AL. (1995a,1995b,1998,2000) umfangreich zur Beschreibung von Rissen und Rissausbreitungen genutzt. In der EFG können Rissausbreitungen wesentlich einfacher beschrieben werden als in der FEM, da die Wichtungsfunktionen so gestaltet werden können, dass sie nicht über Unstetigkeitsstellen hinwegreichen.

Da die MLS-Ansatzfunktionen auch höhere Stetigkeitsanforderungen erfüllen, wurde die EFG zur Umsetzung von Materialmodellen eingesetzt, die höhere Ableitungen von Zustandsvariablen enthalten. ASKES ET AL. (2000) behandeln ein gradientenerweitertes Schädigungsmodell, ASKES & AIFANTIS (2002) Gradientenelastizität und PAMIN ET AL. (2003) Gradientenplastizität.

Im Bereich der Fluidmechanik wurde die EFG von HUERTA ET AL. (2004b) zur Beschreibung der Strömung eines inkompressiblen Fluides verwendet.

Eine vollständig netzfreies Verfahren ist die von ONATE ET AL. (1996a, 1996b) und MENDEZ & VELAZQUEZ (2004) im Zusammenhang mit fluidmechanischen Problemen beschriebene *Finite Point Method* (FPM). Es wird eine *Weighted Least Square* (WLS) Approximation über eine variable Anzahl von Punkten um jeden Auswertungspunkt durchgeführt. Die Approximation wird in den oben angegebenen Artikeln als *Multiple Fixed Least Square* (MFLS) bezeichnet. Die numerische Integration erfolgt direkt über die Punkte ohne Hilfsnetz.

Eine vom eingesetzten Berechnungsverfahren unabhängige Beschreibung der *Least Square* (LS), *Weighted Least Square* (WLS) und *Moving Least Square* (MLS) Approximationen findet sich z.B. in ZIENKIEWICZ & TAYLOR (2000a) oder ONATE ET AL. (1996a).

Die MLS Approximation ist auch geeignet um die Genauigkeit von FE-Berechnungen zu erhöhen (TABBARA ET AL. (1994)). FASSHAUER (2004) beschreibt eine vereinfachte Berechnung der MLS-Funktionen.

2.2.2 SPH, RKPM und MLSRK

Ein anderer Entwicklungszweig beruht auf *Kernschätzungen* (kernel estimates) zur Bestimmung der Ansatzfunktionen. Das ursprünglich aus dem Bereich der Signalanalyse stammende mathematische Verfahren wird auch als *Integrale Fenstertransformation* (Integral Window Transformation) bezeichnet.

Der älteste Vertreter dieser Verfahrensklasse ist die von GINGOLD & MONAGHAN (1977) und MONAGHAN (1982,1988,1995) beschriebene *Smooth Particle Hydrodynamics Method* (SPH), die auch als *Free Lagrange Method* bezeichnet wird. Es ist eine netzfremethode, die speziell im Bereich der Astrophysik zur Beschreibung von Problemen ohne ausgeprägte Ränder eingesetzt wurde. Ableitungen werden in der SPH auf Ableitungen der Fensterfunktion zurückgeführt. Die Ansatzfunktionen bilden besonders am Rand keine *Partition of Unity*. (Der Begriff wird im nächsten Abschnitt erklärt.) SWEGLE ET AL. (1995) wiesen in einer Stabilitätsanalyse nach, dass hierin die Ursache für die Zuginstabilität der Methode liegt. BONET & KULASEGARAM (2000) zeigen Möglichkeiten zur Korrektur und Stabilisierung.

Weiterentwickelt wurde die Methode von W. K. LIU (1995, 1996, 1997) unter der Bezeichnung *Reproducing Kernel Particle Method* (RKPM).

Den Zusammenhang zu Verfahren, die MLS-Approximationen verwenden, stellen LIU ET AL. (1997) her. Sie leiten MLS-Ansatzfunktionen über Kernschätzungen her und bezeichnen die resultierenden Verfahren als *Moving Least-Square Reproducing Kernel Methods* (MLSRK).

CHEN ET AL. (2003) stellen eine RKPM vor, deren Ansatzfunktionen die Knotenwerte interpolieren und nicht nur approximieren, sodass hier die wesentlichen Randbedingungen wie in der FEM direkt berücksichtigt werden können.

Die Konstruktion von höherwertigen Ansatzfunktionen für die FEM über Kernschätzungen wird in LIU ET AL. (2004) vorgestellt. Das Verfahren wird als *Reproducing Kernel Element Method* (RKEM) bezeichnet.

2.2.3 Hp-Clouds und PUFEM

Ein übergeordnetes Konzept ist die Forderung der *Partition of Unity* (PU), die besagt, dass die Summe der Ansatzfunktionen an allen Orten gleich 1 sein muss. Mit der *Partition of Unity* wird sichergestellt, dass konstante Funktionen exakt vom Ansatz wiedergegeben werden (siehe Abschnitt 5.1).

Laut BELYTSCHKO ET AL. (1998) ist es schwierig, die *Konsistenz*¹ eines Verfahrens nachzuweisen. Stattdessen wird vorgeschlagen, als notwendige Bedingung für die Konvergenz eines Verfahrens dessen *Vollständigkeit* zu überprüfen. Ein Verfahren heißt vollständig bis zu einer Ordnung, wenn Funktionen bis zu dieser Ordnung exakt reproduziert werden. Die Reproduktionsbedingung für konstante Funktionen und damit die Reproduktionsbedingung niedrigster Ordnung ist die *Partition of Unity*.

Basierend auf der PU-Forderung wurden sowohl netzfreie Methoden entwickelt als auch die FEM erweitert. Es entstanden Verfahren, die neben der h-Adaptivität auch die Möglichkeit einer p-Adaptivität beinhalten. Als vollständig netzfreie Methode wurden die *hp-Clouds* von DUARTE & ODEN (1995b,1996) und LISZKA ET AL. (1996) eingeführt. Als Erweiterung der FEM entwickelten MELENK & BABUSKA (1996,1997) und TAYLOR ET AL. (1998) die *Partition of Unity Finite Element Method* (PUFEM).

2.2.4 PIC, FLIP und MPM

Eine Stellung zwischen netzfreien Methoden und der FEM nehmen die von HARLOW (1964) mitbegründeten *Particle In Cell Methods* (PIC) ein. Mit dem Fluid oder Körper fest verbundene Lagrangesche Partikel tragen alle Informationen (Zustandsgrößen und ggfs. Geschichtsvariablen) mit sich. Die Berechnung wird in jedem Zeitschritt in eine Lagrangesche und eine konvektive Phase unterteilt. Während der Lagrangeschen Phase erfolgen die Berechnungen auf einem FE-Netz. Anschliessend werden in der konvektiven Phase die Partikel weiterbewegt. Das Netz wird durch ein unverformtes Netz ersetzt.

Für den Bereich der Fluidmechanik und der Magnetohydrodynamik entwickelten BRACKBILL ET AL. (1986,1987,1988,1991) *FLIP* (Fluid-Implicit-Particle).

SULSKY, SCHREYER, YORK ET AL. (1995,1996,1999,2000) begründeten die *Material Point Method* (MPM) zur Berechnung von Problemen der Festkörpermechanik. Das ursprünglich explizite Zeitintegrationsverfahren wurde von GUILKEY & WEISS (2003) und SULSKY & KAUL (2004) durch ein implizites ersetzt. BARDENHAGEN ET AL. (2000) wendeten die MPM auf granulare Materialien und MORESI ET AL. (2003) auf die Berechnung von Geomaterialien an.

¹In der Finite-Differenzen-Literatur wird die Konsistenz einer Approximation als ihre Fähigkeit definiert, eine Differentialgleichung exakt abzubilden, wenn die Anzahl der Netzknoten gegen Unendlich und der Abstand zwischen benachbarten Netzknoten gegen Null strebt.

2.3 Randbedingungen

Die Ansätze der meisten netzfreien Methoden interpolieren Knotenwerte nicht, sondern approximieren diese nur, da die Ansatzfunktionen an den Knoten im Allgemeinen nicht den Wert 0 oder 1 annehmen. Eine direkte Behandlung von wesentlichen Randbedingungen ist daher nicht möglich

Eine Übersicht über Verfahren zur Berücksichtigung der Dirichletschen Randbedingungen liefern GRIEBEL & SCHWEITZER (2002a) und FERNÁNDEZ-MÉNDEZ & HUERTA (2004).

FERNÁNDEZ-MÉNDEZ und HUERTA unterscheiden zwischen Methoden, die die schwache Form, und Methoden, die die Ansatzfunktionen verändern. In die erste Kategorie fallen die *Methode der Lagrangeschen Multiplikatoren*, das *Strafverfahren* und *Nitsches Methode*; in die zweite Kategorie die *Kopplung mit finiten Elementen* und die Verwendung spezieller *Randansatzfunktionen*.

BABUSKA (1973a) beschreibt die Methode der Lagrangeschen Multiplikatoren im Zusammenhang mit der FEM. Die Anwendung im Rahmen netzfreier Methoden unterscheidet sich hiervon nicht. Nachteilig an der Methode der Lagrangeschen Multiplikatoren ist, dass die Anzahl der Unbekannten vergrößert wird und dass das Gleichungssystem eine besondere Struktur hat und indefinit ist. Zur effizienten Berechnung sollten daher spezielle Löser eingesetzt werden. Außerdem ist die Wahl der zusätzlichen Freiheitsgrade nicht unproblematisch, da die Ansatzfunktionsräume im Inneren und auf dem Rand zusätzlich die *Babuska-Brezzi-Bedingung* (siehe BREZZI (1974)) erfüllen müssen.

BABUSKA (1973b) beschreibt das Strafverfahren (Penalty Method) im Zusammenhang mit der FEM; ZHU & ATLURI (1998) beschreiben es im Zusammenhang mit der EFG. Die Anzahl der Freiheitsgrade wird nicht erhöht, und das Gleichungssystem bleibt positiv definit, aber die Kondition des Gleichungssystem verschlechtert sich mit zunehmendem Strafparameter. Der Strafparameter ist so einzustellen, dass einerseits die Randbedingungen hinreichend genau erzwungen werden und andererseits die Qualität der Lösung nicht durch ein zu schlecht konditioniertes Gleichungssystem beeinträchtigt wird.

Die auf NITSCHKE (1970) zurückgehende Methode kann nach FERNÁNDEZ-MÉNDEZ & HUERTA (2004) als konsistente Verbesserung des Strafverfahrens angesehen werden. HANSBO & LARSON (2002) nutzen Nitsches Methode für eine diskontinuierliche FEM und GRIEBEL & SCHWEITZER (2002a) für eine netzfreie Partition of Unity Method. Verglichen mit den Lagrangeschen Multiplikatoren und dem Strafverfahren

ist Nitsches Methode komplizierter in der Anwendung.

Die Kopplung zwischen netzfreien Methoden im Bereichsinneren und netzbasierten Methoden (in der Regel der FEM) am Rand wird u. a. in BELYTSCHKO ET AL. (1995a), HUERTA & FERNÁNDEZ-MÉNDEZ (2000) und HUERTA ET AL. (2004a) beschrieben. Es erscheint allerdings wenig elegant, wenn im Rahmen einer netzfreien Methode zur Befriedigung von wesentlichen Randbedingungen auf finite Elemente zurückgegriffen werden muss.

Als letzte Methode bleibt die Veränderung der Ansatzfunktionen der netzfreien Methode. Diese können entweder nur an den Knoten, an denen wesentliche Randbedingungen vorgegeben werden sollen, oder im gesamten Gebiet dergestalt korrigiert werden, dass sie Knotenwerte interpolieren und nicht nur approximieren. CHEN ET AL. (2003) beschreiben eine *Reproducing Kernel Interpolation*.

Kapitel 3

Implementierte Methoden

3.1 Programmkonzept

Die Bestimmungsgleichungen der behandelten Methoden werden so aufbereitet, dass ein modularer Aufbau des Programmsystems möglich wird und die überwiegenden Teile des Quellcodes von allen Methoden gemeinsam genutzt werden können.

Die Arbeit beschränkt sich auf mechanische Probleme und kleine Verformungen.¹ Es wird das *Ritz-Galerkin-Verfahren* angewendet. Die Zeitintegration erfolgt implizit. Das Programmsystem ist für Lagrangesche und quasi-Eulersche Sichtweisen ausgelegt. Nichtlineare Materialgesetze (ein viskoelastisches und ein elastoplastisches Materialgesetz mit isotroper Verfestigung) sind implementiert. Die Eigenschaften der Ansatzfunktionen eines Teils der Methoden erlauben keine direkte Berücksichtigung der wesentlichen Randbedingungen und Kontaktbedingungen. Diese werden daher über ein Strafverfahren erzwungen. Da sich der C^1 -stetige Kontakt im Zweidimensionalen einfacher formulieren lässt, wurde das Programmsystem für ebene Probleme aufgestellt.

Unter <http://www.schilling-berlin.de/diss.htm> stehen das mit dem Compiler *gcc 4.02* unter dem Betriebssystem *Suse Linux V10.0* (Kernel 2.6.13-158) übersetzte Programm und die Eingabedateien zu den Berechnungsbeispielen aus Abschnitt 9 zum Herunterladen zur Verfügung.

¹ Die Änderungen, die vorgenommen werden müssen, um das Programmsystem für große Verformungen zu erweitern werden in Abschnitt 6.5.3 beschrieben. Für die Untersuchung der Unterschiede zwischen Methoden mit und ohne Elementnetzen ist die Linearisierung des Verzerrungsmaßes kein Nachteil, da die Unterschiede der Methoden nicht mit der Wahl des Verzerrungsmaßes zusammenhängen.

Netzbilder						
Methodennummer	1a (FEM)	1b (FPM)	2a (MPM)	2b	3a	3b
Punkte an Körper fixiert?	ja ja ja	ja ja ja	ja ja nein	ja ja nein	ja nein nein	ja nein nein
Ansatz- funktions- typ	E $(E^{-1}, W)^*$	W $(W)^*$	E E^{-1}, W	W W	E W E^{-1}, W	W W W

= Materialpunkt
 = Integrationspunkt
 = Knoten

E = Elementansätze
 E^{-1} = Invertierte Elementansätze
 W = MLS-Approximation
 $()^*$ = Nur für Postprocessing

Tabelle 3.1: Methoden und Ansatzfunktionen

	=	Materialpunkt	:	Lösung der Materialgleichungen
	=	Integrationspunkt	:	Auswertungspunkt für numerische Integration
	=	Knoten	:	Ort von globalen Unbekannten

Tabelle 3.2: Symbole

3.2 Methodenüberblick

Für eine übersichtliche Implementierung mehrerer Methoden in einem Programmsystem ist es sinnvoll einige Abstraktionen vorzunehmen. Die Materialgleichungen werden an *Materialpunkten* erfüllt. Hier werden auch die Geschichtsvariablen gespeichert. Die Auswertungspunkte für die numerische Integration sind die *Integrationspunkte*. In den meisten Fällen sind Materialpunkte und Integrationspunkte deckungsgleich. Die globalen Unbekannten sind an den *Knoten* beheimatet.

Die betrachteten Methoden unterscheiden sich in den verwendeten Ansätzen für die räumliche Diskretisierung und darin, welche der Punkte (Materialpunkte, Integrationspunkte und Knoten) permanent mit dem Material verbunden sind. Die Tabelle 3.1 gibt hierzu einen Überblick. Die obere Reihe zeigt Netzausschnitte. In der mittleren Reihe ist vermerkt, welche Punkte permanent mit dem Körper verbunden sind und welche nicht. Die untere Reihe zeigt die im Zusammenhang mit den verschiedenen Methoden genutzten Ansätze.

Im Rahmen dieser Arbeit werden von den sechs im Programmsystem vorgesehenen Methoden die *Methode 1a* (Finite-Elemente-Methode) und die *Methode 1b* (Finite-Punkte-Methode) anhand von Beispielen verifiziert und verglichen.

3.2.1 Methoden 1a (FEM) und 1b (FPM)

Die *Methode 1a* ist die Finite-Elemente-Methode (FEM), die *Methode 1b* eine Finite-Punkte-Methode (FPM). Beiden ist gemeinsam, dass alle Punkte (Materialpunkte, Integrationspunkte und Knoten) permanent mit dem diskretisierten Körper verbunden sind und die Material- und Integrationspunkte deckungsgleich sind.

Die Dehnungen und Dehnungsraten an den Materialpunkten werden bei der FEM über die üblichen Elementansätze und bei der FPM über netzfreie Ansätze berechnet. Im Rahmen dieser Arbeit wird für die netzfreien Ansätze die im Abschnitt 5.5.3 hergeleitete MLS-Approximation (MLS = Moving Least Squares, Methode der bewegten kleinsten Fehlerquadrate) verwendet.

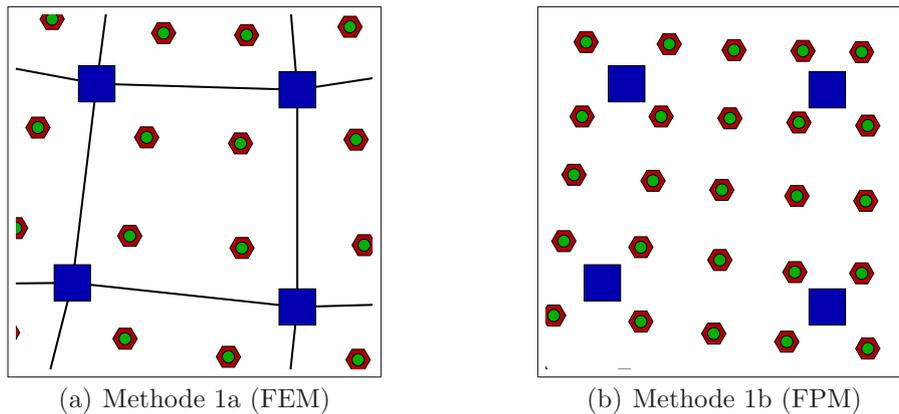


Abbildung 3.1: Netzausschnitte zu Methoden 1a und 1b

Die numerische Integration erfolgt bei der FEM durch eine Gauß-Integration und bei der FPM direkt punktweise, indem die Größen an den Integrationspunkten vor der Assemblierung mit den Integrationspunktvolumen multipliziert werden. Da diese direkte Integrationsmethode ungenauer ist, muss eine gegenüber der FEM höhere Integrationspunktanzahl gewählt werden.

3.2.2 Methoden 2a (MPM) und 2b

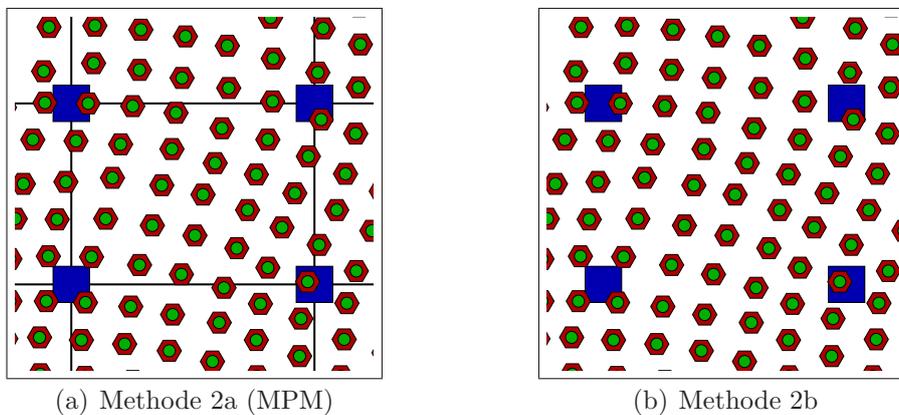


Abbildung 3.2: Netzausschnitte zu Methoden 2a und 2b

Die *Methode 2a* ist eine Abwandlung der Materialpunktmethode (MPM). Die *Methode 2b* verzichtet bei ansonsten gleichem Berechnungsablauf auf Elemente. Bei beiden Methoden (2a und 2b) sind die Material- und Integrationspunkte permanent mit dem diskretisierten Körper verbunden und deckungsgleich. Die Knoten werden nach jedem Zeitschritt in ihre Ausgangslage zurückgesetzt.

Die Berechnung wird in jedem Zeitschritt in eine Lagrangesche und eine konvektive Phase aufgeteilt. In der Lagrangeschen Phase werden aus den Verschiebungen und

Geschwindigkeiten der Knoten die Dehnungen und Dehnungsraten an den Materialpunkten berechnet. Hierzu werden bei der Methode 2a Elementansätze und bei der Methode 2b eine MLS-Approximation verwendet.

Die numerische Integration erfolgt direkt punktweise.

Nach Abschluss der globalen Iteration werden die Material- und Integrationspunkte in der konvektiven Phase entsprechend der im Zeitschritt erfolgten Verformung weiterbewegt. Anschließend werden die Knoten in ihre Ausgangslage zurückgesetzt und die Zustandsgrößen an den Knoten gelöscht.

Am Anfang des nächsten Zeitschrittes werden die Knoten mit den approximierten Materialpunktdaten der Verschiebungen und Geschwindigkeiten initialisiert. Auf diese Weise haben die Elementnetze der Methode 2a und die Knotennetze der Methode 2b in der verformten Konfiguration am Zeitschrittanfang eine regelmäßige Gestalt.

Problematisch ist bei der Methode 2a die direkte punktweise Integration. Werden die üblichen C^0 -stetigen finiten Elemente eingesetzt, so kommt es zu Sprüngen in Teilen der Steifigkeitsmatrix, wenn Materialpunkte in der konvektiven Phase Elementgrenzen überschreiten.

3.2.3 Methoden 3a und 3b

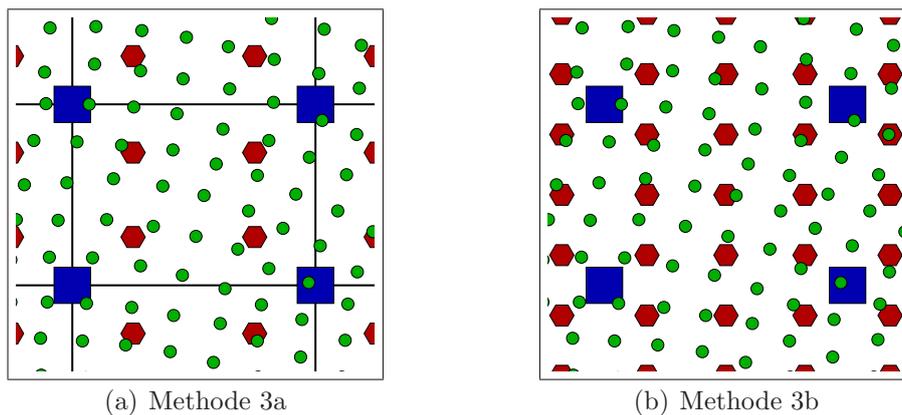


Abbildung 3.3: Netzausschnitte zu Methoden 3a und 2b

Um die bei der Methode 2a auftretenden Probleme zu beheben werden die Integrationspunkte von den Materialpunkten getrennt. Die Materialpunkte sind permanent mit dem diskretisierten Körper verbunden; die Knoten und Integrationspunkte werden nach jedem Zeitschritt auf ihre Ausgangslage zurückgesetzt.

Der Berechnungsablauf ist bis auf den Bereich der numerischen Integration identisch mit dem Berechnungsablauf der Methoden 2a und 2b. Vor der numerischen Integration werden die Integrationspunkte über eine Approximation von Materialpunktdaten initialisiert. Bei der Methode 3a wird eine Gauß-Integration und bei der Methode 3b eine direkte punktweise Integration ausgeführt.

3.3 Berechnungsablauf

Die Reihenfolge der Berechnungen ist für alle im Programmsystem vorgesehenen Methoden identisch. In einem globalen Iterationsschritt innerhalb eines Zeitschrittes werden folgende Berechnungen ausgeführt:

- → ● Die Dehnungen und Dehnungsraten an den Materialpunkten werden in Abhängigkeit von den Knotenverschiebungen und Knotengeschwindigkeiten berechnet.
- An den Materialpunkten werden die nichtlinearen Materialgleichungen iterativ gelöst und die Spannung und die Materialtangente berechnet, die an den Integrationspunkten benötigt werden.
- → ◆ Werden die Methoden 1a (FEM), 1b (FPM), 2a (MPM) oder 2b genutzt, so sind Integrationspunkte und Materialpunkte identisch. In den anderen beiden Fällen müssen durch eine Approximationsmethode Daten von den Materialpunkten zu den Integrationspunkten übertragen werden.
- ◆ An den Integrationspunkten werden volumenbezogene Größen berechnet. Bei der direkten numerischen Integration werden diese mit Integrationspunktvolumina multipliziert, bei einer Gauß-Integration mit den Wichtungsfaktoren der Gaußpunkte.
- ◆ → ■ Die Beiträge der Integrationspunkte werden zu globalen Größen assembliert.
- Ein Iterationsschritt zur Verbesserung des Vektors der globalen Unbekannten wird ausgeführt.

Die Berechnungsabfolge wird bis zur Konvergenz der globalen Iteration wiederholt. Sind die Knoten körperfest, wird mit dem nächsten Zeitschritt fortgefahren. Sind die Knoten nicht permanent, so werden sie am Ende des Zeitschrittes verworfen,

nachdem die Materialpunkte in einer konvektive Phase mit dem verformten Körper weiterbewegt wurden. Die Knoten des neuen Netzes müssen dann über eine geeignete Approximationsmethode mit von den Materialpunkten übertragenen Daten initialisiert werden. Die Berechnungen eines Zeitschrittes sind damit abgeschlossen.

3.4 Formelzusammenstellung für die Methoden 1a (FEM) und 1b (FPM)

Die Finite-Punkte-Methode und die Finite-Elemente-Methode wurden im Rahmen der vorliegenden Arbeit vollständig implementiert, anhand von Beispielen verifiziert und miteinander verglichen. Beide Methoden unterscheiden sich nur in der räumlichen Diskretisierung. Die Bestimmungsgleichungen lassen sich in einheitlicher Form angeben. Dies geschieht an dieser Stelle in zusammengefasster Form.

Es wird eine total Lagrangesche Betrachtungsweise verwendet. Die Kinematik kleiner Verformungen wird angenommen (siehe Fußnote auf S. 15).

$$\boldsymbol{\varepsilon} = \frac{1}{2} [\mathbf{u} \otimes \boldsymbol{\nabla} + \boldsymbol{\nabla} \otimes \mathbf{u}] \quad (3.1)$$

Die nichtlinearen Materialgleichungen und Entwicklungsgleichungen (siehe Abschnitt 4.3) können in der Form

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}, \mathbf{z}) \quad (3.2)$$

$$\dot{\mathbf{z}} = \begin{cases} \mathbf{f}(\boldsymbol{\varepsilon}, \mathbf{z}) & : \text{viskoelastisches Material} \\ \mathbf{f}(\boldsymbol{\varepsilon}, \dot{\boldsymbol{\varepsilon}}, \mathbf{z}) \quad (+ \text{Kuhn-Tucker}) & : \text{elastoplastisches Material} \end{cases} \quad (3.3)$$

angegeben werden. In \mathbf{z} sind die inneren Variablen zusammengefaßt. In der partiell integrierten schwachen Form des Impulssatzes

$$\int_{\mathcal{B}} \delta \boldsymbol{\varepsilon} : \boldsymbol{\sigma} dV + \int_{\mathcal{B}} \delta \mathbf{u} \cdot (\ddot{\mathbf{u}} - \mathbf{b}) \rho dV - \int_{\partial \mathcal{B}} \delta \mathbf{u} \cdot \mathbf{t} dA \quad (3.4)$$

$$- \int_{\partial \mathcal{B}_{\text{rnd}}} \delta \mathbf{u} \cdot \tilde{\mathbf{f}}_{\text{rnd}} dA - \int_{\partial \mathcal{B}_{\text{kon}}} \delta \mathbf{u} \cdot \tilde{\mathbf{f}}_{\text{kon}} dA = 0 \quad (3.5)$$

erscheinen zusätzliche Terme für die wesentlichen Randbedingungen und die Kontaktbedingungen, da diese durch ein Strafverfahren erzwungen werden (siehe Abschnitt 4.5). $\partial \mathcal{B}$, $\partial \mathcal{B}_{\text{rnd}}$ und $\partial \mathcal{B}_{\text{kon}}$ sind die Ränder, auf denen Spannungen, Verschie-

bungen und Kontaktbedingungen vorgegeben werden. Der Ansatz für die räumliche Diskretisierung hat die in Abschnitt 5 näher diskutierte Form:

$$\bar{f}(\mathbf{x}) = \sum_{k=1}^{N_K(\mathbf{x})} L_k(\mathbf{x}) f_k = \underset{\langle N_K \rangle}{\mathbf{L}^T(\mathbf{x})} \underset{\langle N_K \rangle}{\mathbf{f}} \quad (3.6)$$

$$\frac{\partial \bar{f}(\mathbf{x})}{\partial x_i} = \sum_{k=1}^{N_K(\mathbf{x})} \frac{\partial L_k(\mathbf{x})}{\partial x_i} f_k = \frac{\partial \underset{\langle N_K \rangle}{\mathbf{L}^T(\mathbf{x})}}{\partial x_i} \underset{\langle N_K \rangle}{\mathbf{f}} \quad (3.7)$$

$L_k(\mathbf{x})$ ist der Wert der Ansatzfunktion vom Knoten mit dem Index k an der Stelle \mathbf{x} , f_k ist der Stützwert am selben Knoten und $\bar{f}(\mathbf{x})$ ist der approximierte Wert an der Stelle \mathbf{x} . Die Anzahl der Ansatzfunktionen N_K hängt bei der FPM im Gegensatz zur FEM vom Ort \mathbf{x} ab. Die Interpolation mit Elementansatzfunktionen ist mit $N_K = \text{const}$ als Spezialfall enthalten. Die Ansätze für Verschiebungen und virtuelle Verschiebungen haben für FPM und FEM dieselbe Form.

$$\underset{\langle N_D \rangle}{\mathbf{u}(\mathbf{x}_p)} = \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\mathbf{U}_p} \quad (3.8)$$

$$\delta \underset{\langle N_D \rangle}{\mathbf{u}(\mathbf{x}_p)} = \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \delta \underset{\langle N_K N_D \rangle}{\mathbf{U}_p} \quad (3.9)$$

Auch die Verschiebungs-Verzerrungs- und Geschwindigkeits-Verzerrungsgeschwindigkeits-Beziehungen sehen für beide Methoden gleichartig aus.

$$\underset{\langle N_V \rangle}{\boldsymbol{\varepsilon}(\mathbf{x}_p)} = \underset{\langle N_V \times N_K N_D \rangle}{\mathbf{B}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\mathbf{U}_p} \quad (3.10)$$

$$\dot{\underset{\langle N_V \rangle}{\boldsymbol{\varepsilon}}(\mathbf{x}_p)} = \underset{\langle N_V \times N_K N_D \rangle}{\mathbf{B}(\mathbf{x}_p)} \dot{\underset{\langle N_K N_D \rangle}{\mathbf{U}}_p} \quad (3.11)$$

$$\delta \underset{\langle N_V \rangle}{\boldsymbol{\varepsilon}(\mathbf{x}_p)} = \underset{\langle N_K N_D \times N_V \rangle}{\mathbf{B}(\mathbf{x}_p)} \delta \underset{\langle N_K N_D \rangle}{\mathbf{U}_p} \quad (3.12)$$

Ein Unterschied besteht in der variablen Dimension der Matrizen \mathbf{L} und \mathbf{B} , die Werte der Formfunktionen und Werte der Formfunktionsableitungen enthalten. Das räumlich diskretisierte Gleichungssystem wird in der Zeit unter Verwendung der Newmark-Methode diskretisiert.

$$\ddot{\mathbf{U}}_{n+1} = \alpha_1 [\mathbf{U}_{n+1} - \mathbf{U}_n] - \alpha_2 \dot{\mathbf{U}}_n - \alpha_3 \ddot{\mathbf{U}}_n \quad (3.13)$$

$$\dot{\mathbf{U}}_{n+1} = \alpha_4 [\mathbf{U}_{n+1} - \mathbf{U}_n] + \alpha_5 \dot{\mathbf{U}}_n + \alpha_6 \ddot{\mathbf{U}}_n \quad (3.14)$$

Der Vektor \mathbf{U}_n beinhaltet die Knotenverschiebungen am Zeitschrittanfang, der Vektor \mathbf{U}_{n+1} die unbekanntenen Knotenverschiebungen am Zeitschrittende. α_1 bis α_6 sind skalare Vorfaktoren, die von den beiden Parametern des Newmark-

Zeitschrittverfahrens und von der Zeitschrittlänge anhängen. Das globale Gleichungssystem hat nach der Diskretisierung und der Assemblierung die bekannte Form:

$$\begin{aligned} \mathbf{G}(\mathbf{U}_{n+1}) &= \mathbf{F}_{\text{int}}(\mathbf{U}_{n+1}) + \mathbf{F}_{\text{m}}(\mathbf{U}_{n+1}) - \mathbf{F}_{\text{extV}} \\ &\quad - \mathbf{F}_{\text{extA}} - \mathbf{F}_{\text{rnd}}(\mathbf{U}_{n+1}) - \mathbf{F}_{\text{kon}}(\mathbf{U}_{n+1}) = \mathbf{0} \end{aligned} \quad (3.15)$$

mit

$$\mathbf{F}_{\text{m}}(\mathbf{U}_{n+1}) = \mathbf{M} \left[\alpha_1 [\mathbf{U}_{n+1} - \mathbf{U}_n] - \alpha_2 \dot{\mathbf{U}}_n - \alpha_3 \ddot{\mathbf{U}}_n \right] \quad (3.16)$$

Es handelt sich um ein nichtlineares Gleichungssystem für die Knotenverschiebungen am Ende des Zeitschrittes. Da wesentliche Randbedingungen und Kontaktbedingungen mit einem Strafverfahren erzwungen werden, sind zusätzliche von den Knotenverschiebungen abhängige externe Kräfte, \mathbf{F}_{rnd} für die Randbedingungen und \mathbf{F}_{kon} für den Kontakt, vorhanden. Das nichtlineare Gleichungssystem wird mit dem Newton-Raphson Algorithmus gelöst.

$$\mathbf{U}_{n+1}^0 = \mathbf{U}_n \quad (3.17)$$

$$\Delta \mathbf{U}_{n+1}^{j+1} = - \left. \frac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}} \right|_{\mathbf{U}_{n+1}^j} \mathbf{G}(\mathbf{U}_{n+1}^j) = - \mathbf{K}_{\text{T}}^{-1}(\mathbf{U}_{n+1}^j) \mathbf{G}(\mathbf{U}_{n+1}^j) \quad (3.18)$$

$$\mathbf{K}_{\text{T}}(\mathbf{U}_{n+1}^j) = \mathbf{K}_{\text{Tint}}(\mathbf{U}_{n+1}^j) - \mathbf{K}_{\text{Trnd}}(\mathbf{U}_{n+1}^j) - \mathbf{K}_{\text{Tkön}}(\mathbf{U}_{n+1}^j) + \alpha_1 \mathbf{M} \quad (3.19)$$

Der obere Index bezeichnet den Iterationsschritt, der untere den Zeitschritt. Im Linearisierungsprozess bringen die zusätzlichen externen Kräfte Beiträge zur tangentialen Steifigkeitsmatrix hervor.

Damit sind die Bestimmungsgleichungen für FPM und FEM zusammengestellt. Eine ausführliche Darstellung findet sich in den Kapiteln 4 bis 8.

Kapitel 4

Kontinuumsmechanische Grundgleichungen

Als Beispiel für ein Anfangsrandwertproblem wird das dynamische Verhalten deformierbarer Körper betrachtet. Dabei werden das klassische Kontinuumsmodell und die Theorie kleiner Verformungen angewendet. (Ausführliche Darstellungen finden sich in Lehrbüchern zur klassischen Mechanik und Kontinuumsmechanik wie z.B. in GUMMERT & RECKLING (1986), ALTENBACH & ALTENBACH (1994) oder CHADWICK (1999).) In diesem Kapitel werden die benötigten Grundgleichungen zusammengestellt. Es wird durchgehend Tensorschreibweise verwendet.

4.1 Bilanzgleichungen

Die beiden Axiome der klassischen Mechanik (Impulsatz und Drallsatz) haben in ihrer Feldgleichungsformulierung folgende Form:

$$\nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b} = \rho \ddot{\mathbf{u}} \quad (4.1)$$

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T \quad (4.2)$$

Für jeden materiellen Punkt eines Körpers ist die Summe aus der Divergenz des Spannungstensors und der Volumenlast gleich dem Produkt aus der Dichte und der Beschleunigung dieses Punktes. Im Rahmen der Theorie kleiner Verformungen fallen räumliche und materielle Betrachtungsweise zusammen. Es gibt nur ein, als Folgerung aus dem Drallsatz symmetrisches Spannungsmaß.

Die Massenbilanz ist bei materieller Betrachtungsweise automatisch erfüllt.

4.2 Kinematik

Die Linearisierung des Green-Lagrangeschen Verzerrungstensors für kleine Verformungen führt zum *Deformator*. Er ist invariant gegenüber Starrkörpertranslationen aber nicht invariant gegenüber Starrkörperrotationen (siehe Fußnote auf S. 15).

$$\boldsymbol{\varepsilon} = \frac{1}{2} [\mathbf{u} \otimes \boldsymbol{\nabla} + \boldsymbol{\nabla} \otimes \mathbf{u}] \quad (4.3)$$

4.3 Materialgesetze

Die Schnittstelle für Materialgesetze im Programmsystem ist vom Berechnungsverfahren unabhängig. Stellvertretend für andere ratenabhängige und ratenunabhängige Materialmodelle sind im Programmsystem bereits ein viskoelastisches und ein elastoplastisches Materialmodell eingebaut. Sie können als Vorlage bei der Implementierung weiterer Materialmodelle dienen. Das elastoplastische Materialmodell wird benötigt, da Partikelmethode im Zusammenhang mit Umformprozessen untersucht werden sollen.

4.3.1 Rheologische Modelle

Motiviert werden Materialgesetze häufig über eindimensionale rheologische Modelle. Tabelle 4.1 zeigt die gebräuchlichsten rheologischen Grundelemente und die dazuge-

Hookesche Feder	Newtonscher Dämpfungszylinder	St. Venantsches Trockenreibungsglied
		
$\sigma = E \varepsilon$	$\sigma = \eta \dot{\varepsilon}$	$\sigma \leq \sigma_y$
elastisch	viskos	starr-plastisch

Tabelle 4.1: Rheologische Grundelemente

hörigen Materialgesetze für den eindimensionalen Fall. Werden Elemente in Reihe angeordnet, so ergibt sich bei Annahme kleiner Verzerrungen die Gesamt*dehnung* aus der Summe der Einzeldehnungen; werden Elemente parallel angeordnet, so ergibt sich die Gesamt*spannung* aus der Summe der Einzelspannungen.

4.3.2 Viskoelastisches Materialverhalten

Rheologische Modelle für viskoelastisches Materialverhalten ergeben sich aus Kombinationen von Federn und Dämpfern. Eine Parallelschaltung von Feder und Dämpfer ergibt das Kelvin-Modell, eine Reihenschaltung das Maxwell-Modell. Mit dem Kelvin-Modell kann Kriechen (Zunahme der Verzerrungen bei konstanter Spannung), mit dem Maxwell-Modell Relaxation (Abnahme der Spannung bei konstanter Dehnung) beschrieben werden. Eine Vielzahl von Kombinationen aus den Grundele-

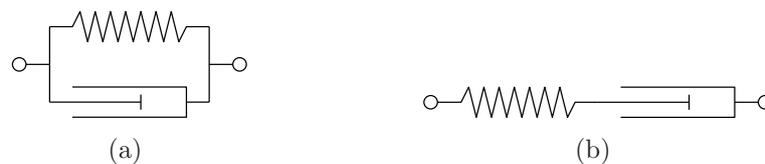


Abbildung 4.1: Viskoelastische 2-Element-Körper, Kelvin-Körper (a) und Maxwell-Körper (b)

menten sind denkbar.¹ Die Abbildung 4.2 zeigt gebräuchliche 3-Element-Körper und einen gebräuchlichen 4-Element-Körper. Die Maxwell-, Lethersich-, Jeffreys- und

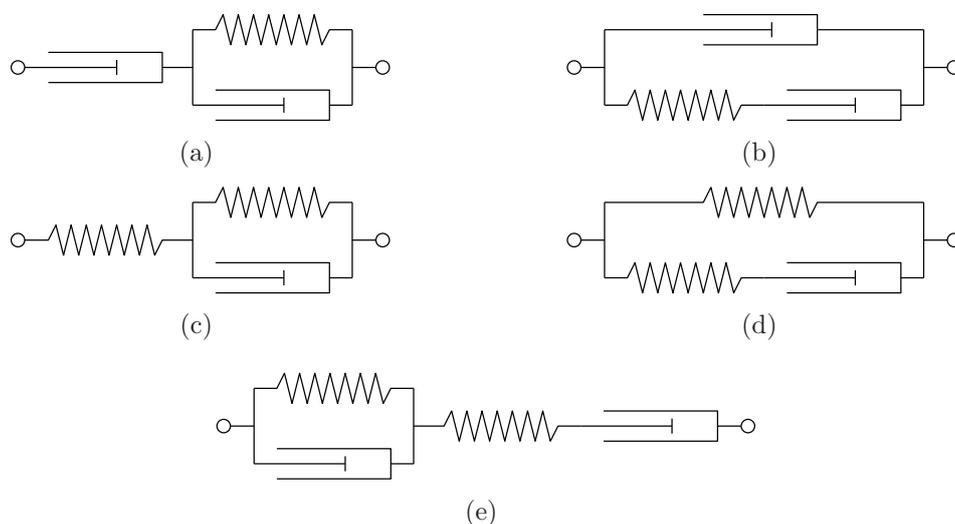


Abbildung 4.2: Viskoelastische 3- und 4-Element-Körper, Lethersich-Körper (a), Jeffreys-Körper (b), Zehner-K-Körper (c), Zehner-M-Körper (d), Burgers-Körper (e)

¹Die Dämpfungseigenschaften der gezeigten einfachen Modelle sind stark frequenzabhängig. Keine befriedigende Lösung bieten die verallgemeinerten Modelle die durch Parallelschaltung von Maxwell-Elementen (generalisierter Maxwell-Körper) oder Reihenschaltung von Kelvin-Elementen (Kelvin-Kette) erhalten werden, da diese Modelle eine hohe Parameteranzahl haben und außerdem keine „glatte“ Anpassung an gemessene Kurven möglich ist. Eine Lösungsmöglichkeit besteht in der Betrachtung eines verallgemeinerten rheologischen Grundkörpers, dessen Eigenschaften zwischen denen der Feder und des Dämpfungszylinders liegen. Dies führt auf das Konzept der fraktionalen Ableitung, welches hier nicht weiter verfolgt wird.

Burgers-Modelle beschreiben viskoelastische Fluide, da sie ein vollständiges Relaxationsvermögen aufweisen und unter endlicher Last unbegrenzte Kriechverformungen auftreten. Die übrigen Modelle beschreiben viskoelastische Festkörper.

Verwendet wird das Zener-M-Modell: Es hat drei Materialparameter (E , E_2 , η) und

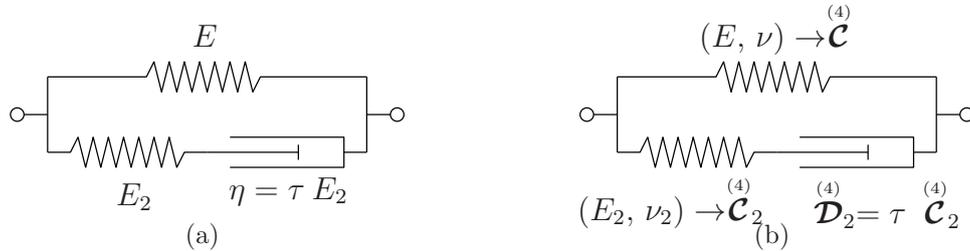


Abbildung 4.3: Zener-M-Körper mit zugeordneten Materialparametern, eindimensional (a), Verallgemeinerung (b)

eine innere Unbekannte (ε_d), die Dehnung im Dämpfer. Als Materialgesetz und als Entwicklungsgleichung für die innere Unbekannte ergeben sich mit τ als Relaxationszeit:

$$\sigma = E \varepsilon + E_2 [\varepsilon - \varepsilon_d] \quad (4.4)$$

$$\dot{\varepsilon}_d = \frac{E_2}{\eta} [\varepsilon - \varepsilon_d] = \frac{1}{\tau} [\varepsilon - \varepsilon_d] \quad (4.5)$$

Die vorstehenden Beziehungen werden auf den mehrdimensionalen Fall verallgemeinert:

$$\boldsymbol{\sigma} = \mathcal{C}^{(4)} : \boldsymbol{\varepsilon} + \mathcal{C}_2^{(4)} : [\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_d] \quad (4.6)$$

$$\dot{\boldsymbol{\varepsilon}}_d = \frac{1}{\tau} [\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_d] \quad (4.7)$$

Hierin sind $\mathcal{C}^{(4)}$ und $\mathcal{C}_2^{(4)}$ linear elastisches Materialverhalten beschreibende Materialtetraden. Es wurde vorausgesetzt, dass die die Dämpfung beschreibende Materialtetrade kollinear zu $\mathcal{C}_2^{(4)}$ ist. Das Materialverhalten wird dann durch vier die elastischen Komponenten des Modells charakterisierenden Materialparameter (z.B. die Elastizitätsmodule E und E_2 und die Querkontraktionszahlen ν und ν_2) und die Relaxationszeit τ beschrieben.

4.3.3 Elastoplastisches Materialverhalten

Als Beispiel für ratenunabhängiges nichtlineares Materialverhalten bei kleinen Verformungen wurde *von Mises* Plastizität mit isotroper Verfestigung innerhalb des

Programmsystems implementiert. Ausführliche Darstellungen finden sich in PRAGER (1955) und RECKLING (1967). An dieser Stelle werden nur die Materialgleichung, die Entwicklungsgleichungen für die inneren Variablen und die Fließbedingung angegeben.



Abbildung 4.4: Elastoplastischer Körper mit zugeordneten Materialparametern, ein-dimensional (a), Verallgemeinerung (b)

Die Gesamtdehnung setzt sich aus elastischer und plastischer Dehnung zusammen, wobei angenommen wird, dass die plastische Verformung volumenerhaltend ist.

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_{el} + \boldsymbol{\varepsilon}_{pl} \quad , \quad \boldsymbol{\varepsilon}_{pl} = \hat{\boldsymbol{\varepsilon}}_{pl} \quad (4.8)$$

Innere Variablen sind die plastische Dehnung $\boldsymbol{\varepsilon}_{pl}$ und die plastische Vergleichsdehnung α_{iso} . Zu der Materialgleichung

$$\boldsymbol{\sigma} = \mathcal{C}^{(4)} : \boldsymbol{\varepsilon}_{el} = \mathcal{C}^{(4)} : [\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_{pl}] \quad (4.9)$$

und den Entwicklungsgleichungen für die inneren Variablen

$$\begin{aligned} \dot{\hat{\boldsymbol{\varepsilon}}}_{pl} &= \lambda \mathbf{N} \quad \text{mit} \quad \mathbf{N} = \frac{\hat{\boldsymbol{\sigma}}}{\|\hat{\boldsymbol{\sigma}}\|} = \frac{\hat{\boldsymbol{\varepsilon}}_{el}}{\|\hat{\boldsymbol{\varepsilon}}_{el}\|} \\ \dot{\alpha}_{iso} &= \lambda \sqrt{\frac{2}{3}} \end{aligned} \quad (4.10)$$

kommt als weitere Beziehung die Fließbedingung

$$\Phi \leq 0 \quad (4.11)$$

mit der Fließfunktion

$$\Phi = \|2 \mu [\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_{pl}]\| - \sqrt{\frac{2}{3}} [\sigma_{y0} + H_{iso} \alpha_{iso}] \quad (4.12)$$

hinzu. In den vorstehenden Beziehungen sind deviatorische Anteile mit „ $\hat{\cdot}$ “ gekennzeichnet. $\mathcal{C}^{(4)}$ ist eine elastische Materialtetrad, \mathbf{N} ist die Richtung der plastischen Dehnung und λ ein skalarer Faktor, der die Größe der plastischen Dehnung bestimmt. Materialparameter sind der Elastizitätsmodul E , der Schubmodul μ , die Anfangsfließspannung σ_{y0} und der isotrope Verfestigungsparameter H_{iso} .

Die Fließfunktion Φ ist stets kleiner oder gleich Null. Wenn die Fließfunktion Φ verschwindet, können drei Fälle auftreten:

$$\begin{aligned} \dot{\Phi} < 0 \quad \text{und} \quad \lambda = 0 & : \text{ elastische Entlastung} \\ \dot{\Phi} = 0 \quad \text{und} \quad \lambda = 0 & : \text{ neutrale Spannungsänderung} \\ \dot{\Phi} = 0 \quad \text{und} \quad \lambda > 0 & : \text{ plastisches Fließen} \end{aligned} \quad (4.13)$$

Hierzu äquivalent sind die Kuhn-Tucker-Bedingungen (4.14) und die Konsistenzbedingung (4.15):

$$\lambda \geq 0 \quad , \quad \Phi \leq 0 \quad , \quad \lambda \Phi = 0 \quad (4.14)$$

$$\Phi = 0 \quad \rightarrow \quad \lambda \dot{\Phi} = 0 \quad (4.15)$$

Bei elastoplastischem Materialverhalten muss also in jedem Lastschritt bzw. Zeitschritt eine Fallunterscheidung vorgenommen werden. Die Fließfunktion entscheidet darüber, ob sich die inneren Variablen weiterentwickeln oder nicht. Tritt plastisches Fließen auf, so steht neben den Entwicklungsgleichungen (4.10) die Bedingung $\Phi = 0$ mit der Fließfunktion nach (4.12) zur Verfügung, um die inneren Variablen und den skalaren Faktor λ zu bestimmen.

Wird im rheologischen Modell parallel zur Feder und zum Trockenreibungselement ein Dämpfer angeordnet, so ergibt sich ein ratenabhängiges Materialverhalten. Die

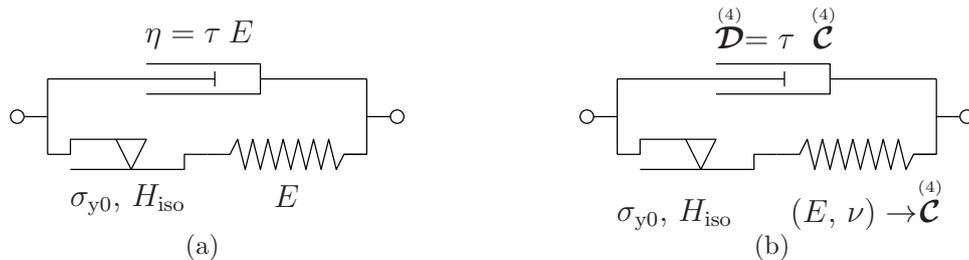


Abbildung 4.5: Elastoplastisch-viskoser Körper mit zugeordneten Materialparametern, eindimensional (a), Verallgemeinerung (b)

Spannungsbeziehung (4.9) ist um die Spannung im Dämpfer zu ergänzen:

$$\boldsymbol{\sigma} = \overset{(4)}{\mathcal{C}} : [\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}_{pl}] + \overset{(4)}{\mathcal{D}} : \dot{\boldsymbol{\varepsilon}} \quad (4.16)$$

Die viskose Materialtetrad $\overset{(4)}{\mathcal{D}}$ wird über die elastische Materialtetrad $\overset{(4)}{\mathcal{C}}$ und eine Relaxationszeit τ definiert:

$$\overset{(4)}{\mathcal{D}} = \tau \overset{(4)}{\mathcal{C}} \quad (4.17)$$

Mit $\tau = 0$ wird ratenunabhängiges Materialverhalten erhalten. Ein kleiner Wert für die Relaxationszeit τ kann zur Stabilisierung von Berechnungen dienen.

4.3.4 Allgemeine Form

Damit eine einheitliche Schnittstelle zu den Materialroutinen geschaffen werden kann, ist es notwendig, die Material- und Entwicklungsgleichungen in allgemeingültige Formen zu bringen, die die verwendeten Material- und Entwicklungsgleichungen als Spezialfälle enthalten. Hierzu werden alle inneren Variablen in Voigt-Notation in einem Vektor \mathbf{z} angeordnet. Im Falle des elastoplastischen Materialverhaltens wird außerdem der skalare Faktor λ in diesem Vektor untergebracht. Es ergibt sich die verallgemeinerte Materialgleichung

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}, \dot{\boldsymbol{\varepsilon}}, \mathbf{z}) \quad (4.18)$$

und die verallgemeinerte Entwicklungsgleichung

$$\dot{\mathbf{z}} = \mathbf{f}(\boldsymbol{\varepsilon}, \dot{\boldsymbol{\varepsilon}}, \mathbf{z}) \quad . \quad (4.19)$$

Im Falle elastoplastischen Materialverhaltens sind zusätzlich noch die Kuhn-Tucker-Bedingungen (4.14) und die Konsistenzbedingung (4.15) zu beachten.

4.4 Schwache Form der Bilanzgleichungen

Aus der starken Form des Impulssatzes wird die schwache Form gewonnen, indem Gleichung (4.1), in Residualform gebracht, mit einer Testfunktion $\delta \mathbf{u}$ multipliziert und über ein Kontrollvolumen \mathcal{B} integriert wird.

$$\begin{aligned} \hat{g} &= \int_{\mathcal{B}} [\nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b} - \rho \ddot{\mathbf{u}}] \cdot \delta \mathbf{u} dV \\ &= \int_{\mathcal{B}} [\nabla \cdot \boldsymbol{\sigma}] \cdot \delta \mathbf{u} dV + \int_{\mathcal{B}} [\mathbf{b} - \ddot{\mathbf{u}}] \cdot \delta \mathbf{u} \rho dV = 0 \end{aligned} \quad (4.20)$$

Zur räumlichen Diskretisierung werden aus einem Ansatzfunktionsraum stammende Ansatzfunktionen verwendet. Sind Ansatzfunktionsraum und Testfunktionsraum identisch, so handelt es sich um ein *Ritz-Galerkin-Verfahren*, andernfalls um ein *Petrov-Galerkin-Verfahren*.

Im Rahmen dieser Arbeit sollen ausschließlich Ritz-Galerkin-Verfahren zum Einsatz kommen. Damit bestimmt die höchste in Gleichung (4.20) auftretende räumliche Ableitungsordnung von Verschiebung \mathbf{u} oder Testfunktion $\delta\mathbf{u}$ die Glattheits- bzw. Stetigkeitsanforderung an den Funktionsraum.

Die Ansätze sind beim *Ritz-Galerkin-Verfahren* im Gegensatz zu den globalen Ansätzen des *Ritz-Verfahrens* lokal. Die Ansatzfunktionen sind wie beim Ritz-Verfahren unter Beachtung der wesentlichen Randbedingungen (beim vorliegenden physikalischen Problem sind das die Verschiebungsrandbedingungen) auszusuchen. Es ist also hinreichend wenn nach SZABÓ (1984b) *zulässige Funktionen* verwendet werden und nicht notwendig, dass *Vergleichsfunktionen* (die alle Randbedingungen erfüllen) verwendet werden.

Wird der Testfunktionsraum nicht eingeschränkt, so sind starke und schwache Form mathematisch gesehen gleichwertig. Soll die schwache Form des Impulssatzes numerisch behandelt werden, so muss der Testfunktionsraum eingeschränkt werden. Damit sorgt die Befriedigung der schwachen Form (4.20) nur noch näherungsweise für die Befriedigung der zugrunde liegenden Differentialgleichung (4.1).

Die Spannungen hängen von Verzerrungen (d.h. Verschiebungsableitungen) ab. Da im ersten Integral der schwachen Form (4.20) die Divergenz des Spannungstensors gebildet wird, treten Verschiebungsableitungen bis zur 2. Ordnung auf.

Um eine in den Ableitungsordnungen der Verschiebungen und der Testfunktionen symmetrische Formulierung zu erhalten, wird die schwache Form partiell integriert und unter Ausnutzung der Symmetrie des Spannungstensors und des Gaußschen Integralsatzes umgeformt. Damit folgt für die partiell integrierte schwache Form des Impulssatzes:

$$-\hat{g} = g = \int_{\mathcal{B}} \delta\boldsymbol{\varepsilon} : \boldsymbol{\sigma} dV + \int_{\mathcal{B}} \delta\mathbf{u} \cdot [\ddot{\mathbf{u}} - \mathbf{b}] \rho dV - \int_{\partial\mathcal{B}} \delta\mathbf{u} \cdot \mathbf{t} dA = 0 \quad (4.21)$$

4.5 Randbedingungen und Kontakt

Da bei einem Teil der implementierten Methoden die Eigenschaften der für die räumliche Diskretisierung verwendeten Ansatzfunktionen eine direkte Behandlung von wesentlichen Randbedingungen und Kontaktbedingungen nicht gestatten, wird ein Strafverfahren eingesetzt (siehe Abschnitte 2.3 und 5.1). Das Verfahren ermöglicht die Erzwingung beliebiger inhomogener und schiefwinkliger Verschiebungsrandbedingungen.

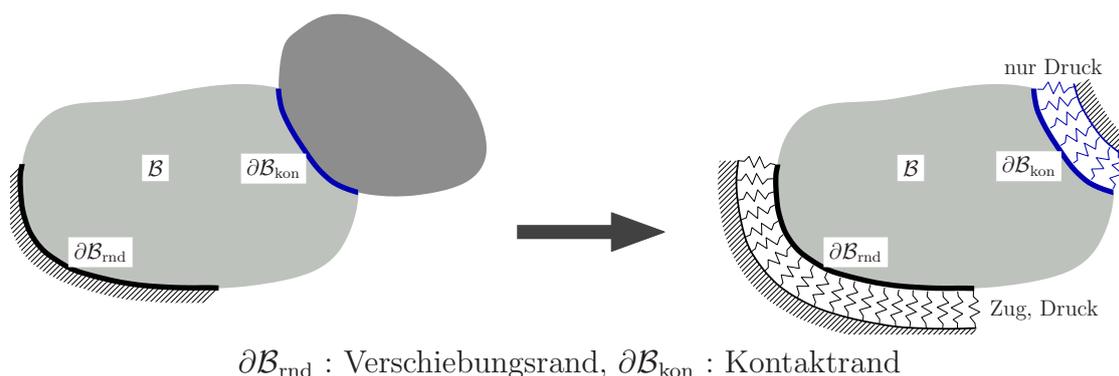


Abbildung 4.6: Wesentliche Randbedingungen und Kontaktbedingungen, Anwendung eines Strafverfahrens

Physikalisch betrachtet werden durch das Strafverfahren wesentliche Randbedingungen in gemischte Randbedingungen überführt. Die Spannung ist am nun nachgiebig gelagerten Rand abhängig von der Verschiebung. Die Lagerung wird so steif gewählt, dass die wesentlichen Randbedingungen hinreichend genau nachgebildet werden, und so weich, dass die Konditionszahl des globalen Gleichungssystems (siehe Abschnitt 7.1.4) hinreichend klein bleibt. Werden die Berechnungen mit doppelter Genauigkeit durchgeführt, bleibt für die Wahl des Strafparameters ein großer Spielraum.

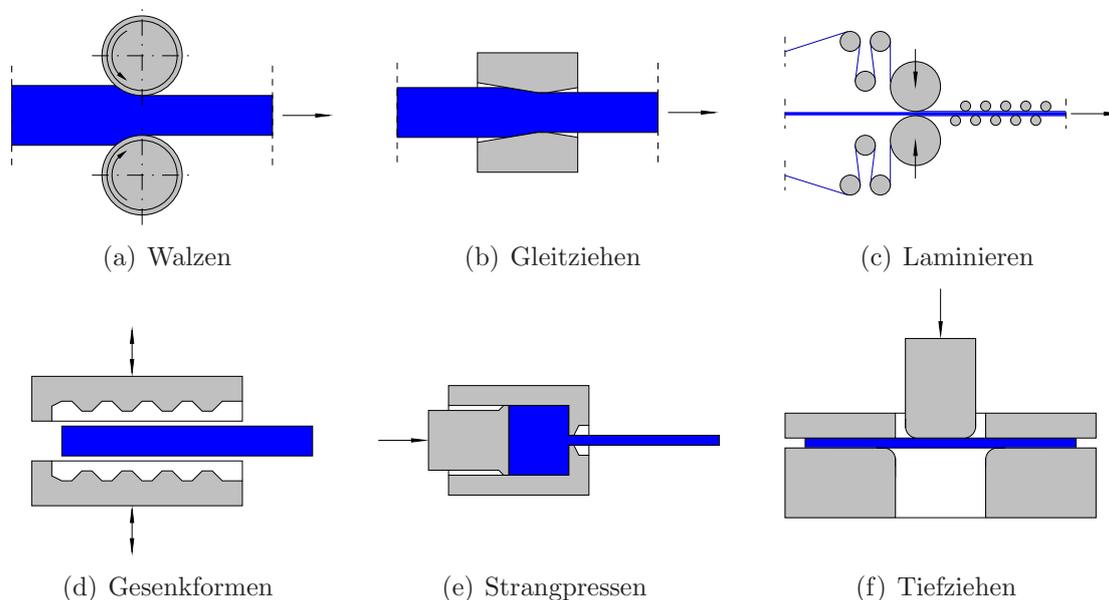


Abbildung 4.7: Beispiele für Umformprozesse

Der Kontakt wird beschränkt auf Normalkontakt zu starren Oberflächen. Eine Beschreibung der C^1 -stetigen Diskretisierung der Kontaktoberflächen und dazugehörige Literaturangaben finden sich in Abschnitt 6.3.

Die Randbedingungen und Kontaktbedingungen werden nicht kontinuierlich sondern

für diskrete Randpunkte aufgestellt. Hierzu werden punktweise die nachgiebigen Lagerungen zu Federn und die Randspannungen zu Randkräften zusammengefaßt.

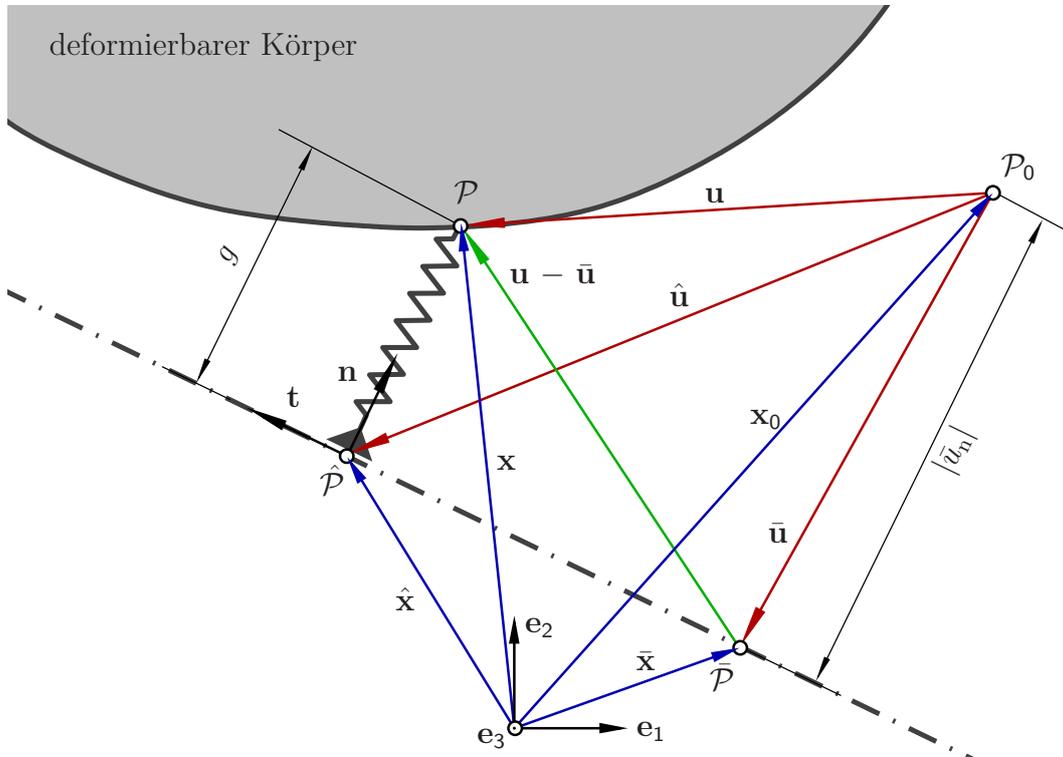


Abbildung 4.8: Strafverfahren, Geometrie einer Verschiebungsrandbedingung

Die Abbildungen 4.8 und 4.9 zeigen die Geometrien einer Verschiebungsrandbedingung und einer inaktiven und einer aktiven Kontaktbedingung für einen einzelnen Randpunkt.

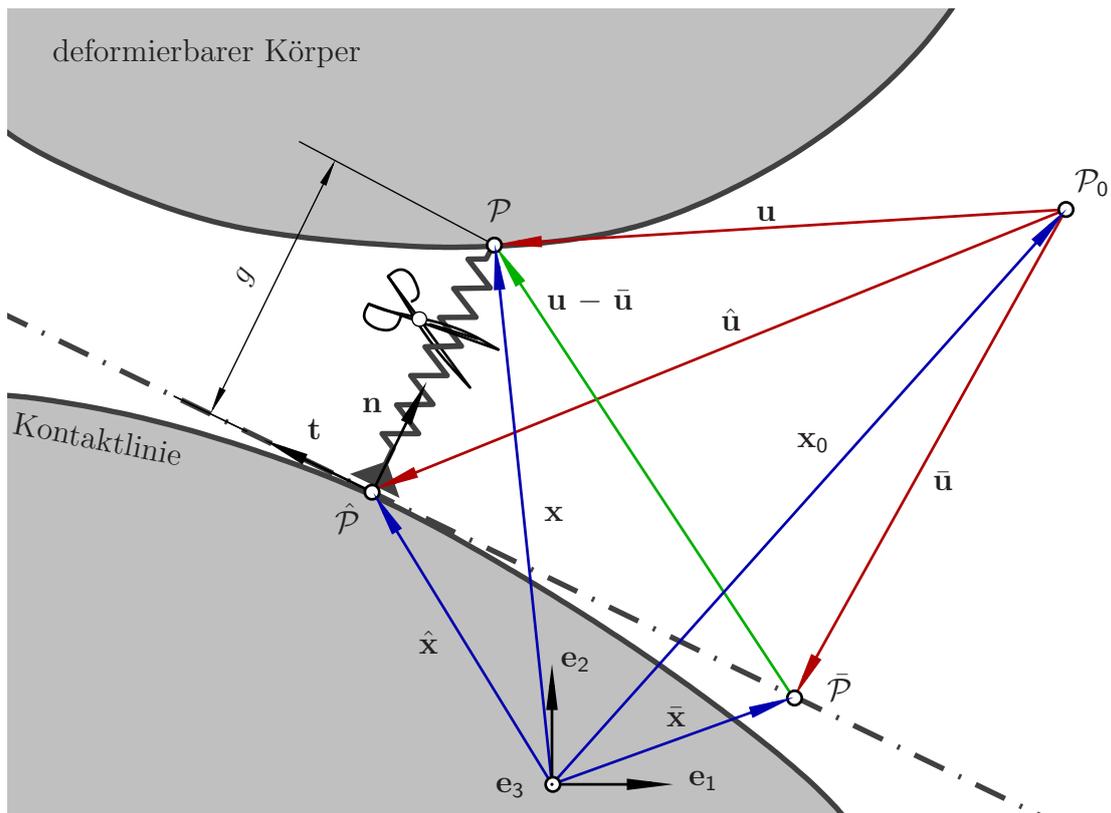
In beiden Abbildungen ist \mathcal{P}_0 die Lage des Randpunktes in der Referenzkonfiguration und \mathcal{P} die Lage in der Momentankonfiguration. Nimmt der Punkt eine Lage $\hat{\mathcal{P}}$ auf der strichpunkttierten Linie ein, so wirken auf ihn keine Kräfte infolge der Strafbedingung. $\hat{\mathcal{P}}$ ist der Fußpunkt des Lotes von \mathcal{P} auf die strichpunktiierte Linie.

Eine inhomogene und schiefe Verschiebungsrandbedingung wird über eine Richtung \mathbf{n} und eine Verschiebung in dieser Richtung

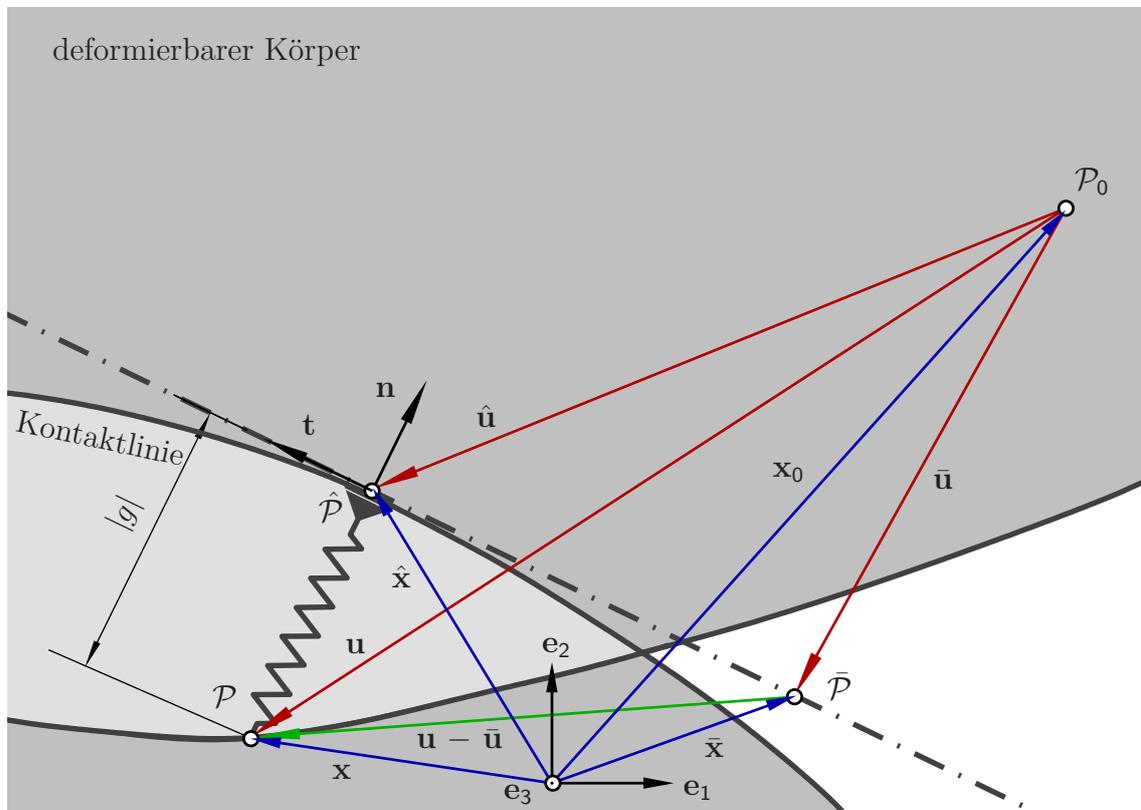
$$\bar{u}_n = \bar{\mathbf{u}} \cdot \mathbf{n} = \hat{\mathbf{u}} \cdot \mathbf{n} \quad (4.22)$$

definiert. (In der Abbildung 4.8 ist \bar{u}_n negativ.) Mit \mathbf{n} und \bar{u}_n sind die zulässigen Lagen des Randpunktes eindeutig definiert.

Im Falle einer Kontaktbedingung wird vom Randpunkt \mathcal{P} ein Lot auf die Kontaktfläche gefällt. Damit werden der Ortsvektor $\hat{\mathbf{x}}$ vom Fußpunkt $\hat{\mathcal{P}}$ des Lotes und



(a) Punkt P außerhalb der Kontaktoberfläche, $g > 0$, Kontaktbedingung inaktiv



(b) Punkt P innerhalb der Kontaktoberfläche, $g < 0$, Kontaktbedingung aktiv

Abbildung 4.9: Strafverfahren, Geometrie einer Kontaktbedingung

die Richtung \mathbf{n} erhalten. Die Berechnung hängt von der Art der Diskretisierung der Kontaktfläche ab und wird in Abschnitt 6.3.3 beschrieben. \bar{u}_n wird aus

$$\bar{u}_n = [\hat{\mathbf{x}} - \mathbf{x}_0] \cdot \mathbf{n} = \hat{\mathbf{u}} \cdot \mathbf{n} \quad (4.23)$$

gewonnen. (In den Abbildungen 4.9(a) und 4.9(b) ist \bar{u}_n negativ.)

Das Abstandsmaß g ergibt sich für wesentliche Randbedingungen und Kontaktbedingungen aus

$$\begin{aligned} g &= [\mathbf{x} - \bar{\mathbf{x}}] \cdot \mathbf{n} = [\mathbf{u} - \bar{\mathbf{u}}] \cdot \mathbf{n} \\ &= [\mathbf{x} - \hat{\mathbf{x}}] \cdot \mathbf{n} = [\mathbf{u} - \hat{\mathbf{u}}] \cdot \mathbf{n} = u_n - \bar{u}_n \quad . \end{aligned} \quad (4.24)$$

Die Ableitung des Abstandsmaßes g nach der Punktverschiebung \mathbf{u}

$$\frac{\partial g}{\partial \mathbf{u}} = \mathbf{n} \quad (4.25)$$

wird später bei der Linearisierung der Strafräfte benötigt.

Repräsentiert eine Feder eine wesentliche Randbedingung, so ist sie für positive und negative Werte von g aktiv. Steht eine Feder für eine Kontaktbedingung, so ist sie nur aktiv, wenn der Punkt eindringt ($g < 0$).

Für wesentliche Randbedingungen wird das lineare Federkraftgesetz

$$f_{\text{rnd}} = -\alpha g \quad (4.26)$$

$$\frac{\partial f_{\text{rnd}}}{\partial g} = -\alpha \quad (4.27)$$

verwendet; für Kontaktbedingungen ein lineares Federkraftgesetz, dass nur bei Eindringung ($g < 0$) aktiv ist

$$f_{\text{kon}} = \begin{cases} -\alpha g & \text{für } g < 0 \\ 0 & \text{für } g \geq 0 \end{cases} \quad (4.28)$$

$$\frac{\partial f_{\text{kon}}}{\partial g} = \begin{cases} -\alpha & \text{für } g < 0 \\ 0 & \text{für } g \geq 0 \end{cases} \quad (4.29)$$

Die Ableitungen der Kraftgesetze nach g sind angegeben, da diese bei der Linearisierung der Strafräfte in Abschnitt 6.5.2 benötigt werden. Die Kraftgesetze und deren Ableitungen zeigen die Abbildungen 4.10 und 4.11. Vorschläge für Kontaktfederkraftgesetze, die auch an der Stelle $g = 0$ glatt (C^1 -stetig) sind, finden sich in

ZAVARISE ET AL. (1998).

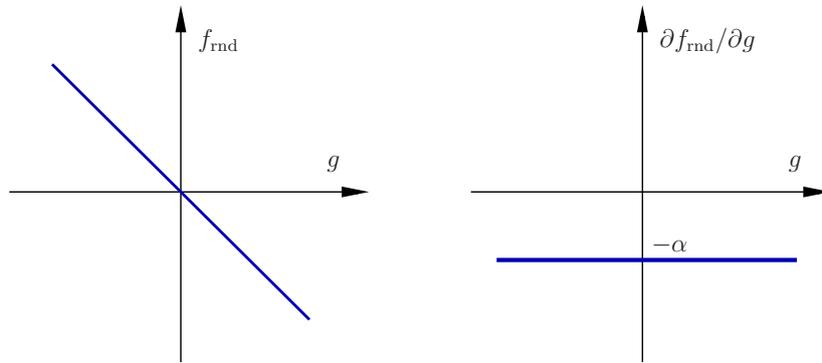


Abbildung 4.10: Strafverfahren, Federkraftgesetz für wesentliche Randbedingung

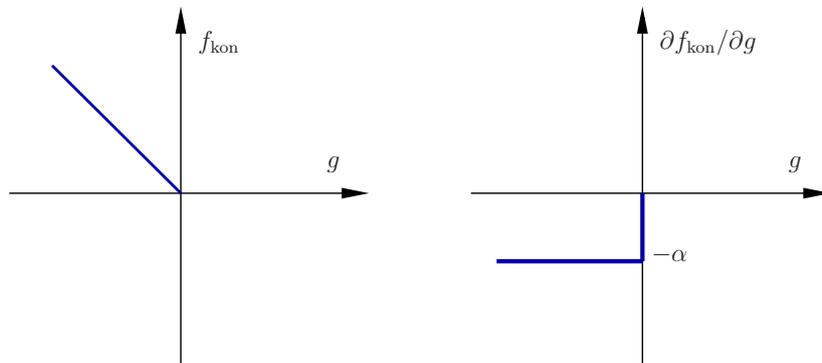


Abbildung 4.11: Strafverfahren, Federkraftgesetz für Kontaktbedingung

Als Kraftvektoren und deren Ableitungen ergeben sich für wesentliche Randbedingungen

$$\mathbf{f}_{\text{rnd}} = f_{\text{rnd}} \mathbf{n} = -\alpha g \mathbf{n} \quad (4.30)$$

$$\frac{\partial \mathbf{f}_{\text{rnd}}}{\partial g} = -\alpha \mathbf{n} \quad (4.31)$$

und für Normalkontakt

$$\mathbf{f}_{\text{kon}} = f_{\text{kon}} \mathbf{n} = \begin{cases} -\alpha g \mathbf{n} & \text{für } g < 0 \\ \mathbf{0} & \text{für } g \geq 0 \end{cases} \quad (4.32)$$

$$\frac{\partial \mathbf{f}_{\text{kon}}}{\partial g} = \begin{cases} -\alpha \mathbf{n} & \text{für } g < 0 \\ \mathbf{0} & \text{für } g \geq 0 \end{cases} \quad (4.33)$$

$\partial \mathcal{B}_{\text{rnd}}$ und $\partial \mathcal{B}_{\text{kon}}$ seien die Teile der Oberfläche des Integrationsgebietes \mathcal{B} auf denen wesentliche Randbedingungen und Kontaktbedingungen über das Strafverfahren erzwungen werden. Die partiell integrierte schwache Form des Impulssatzes (4.21) ist

dann um die zwei Terme

$$- \int_{\partial \mathcal{B}_{\text{rnd}}} \tilde{\mathbf{f}}_{\text{rnd}} \cdot \delta \mathbf{u} \, dA - \int_{\partial \mathcal{B}_{\text{kon}}} \tilde{\mathbf{f}}_{\text{kon}} \cdot \delta \mathbf{u} \, dA \quad (4.34)$$

zu ergänzen. Die Strafräfte $\tilde{\mathbf{f}}_{\text{rnd}}$ und $\tilde{\mathbf{f}}_{\text{kon}}$ sind wie in (4.30) und (4.32) definiert, aber im Unterschied zu diesen auf die Flächeneinheit bezogen. Damit nimmt die partiell integrierte schwache Form des Impulssatzes folgende Form an:

$$\begin{aligned} g = & \int_{\mathcal{B}} \delta \boldsymbol{\varepsilon} : \boldsymbol{\sigma} \, dV + \int_{\mathcal{B}} \delta \mathbf{u} \cdot \ddot{\mathbf{u}} \, \rho \, dV \\ & - \int_{\mathcal{B}} \delta \mathbf{u} \cdot \mathbf{b} \, \rho \, dV - \int_{\partial \mathcal{B}} \delta \mathbf{u} \cdot \mathbf{t} \, dA \\ & - \int_{\partial \mathcal{B}_{\text{rnd}}} \delta \mathbf{u} \cdot \tilde{\mathbf{f}}_{\text{rnd}} \, dA - \int_{\partial \mathcal{B}_{\text{kon}}} \delta \mathbf{u} \cdot \tilde{\mathbf{f}}_{\text{kon}} \, dA = 0 \end{aligned} \quad (4.35)$$

Kapitel 5

Ansätze zur räumlichen Diskretisierung

Im Rahmen numerischer Berechnungsverfahren werden kontinuierliche Zustandsgrößenverläufe diskretisiert. Die an einzelnen Stützpunkten bekannten oder zu berechnenden Zustandsgrößen werden zwischen den Stützpunkten über Ansätze interpoliert oder approximiert. Im Falle der Interpolation werden die Stützwerte durch den Ansatz exakt reproduziert; im allgemeineren Falle der Approximation werden die Abweichungen zwischen den Stützwerten und dem Ansatz minimiert.

Ein und derselbe Ansatztyp kann unterschiedlichen Zwecken dienen. Mit der elementnetzfreien MLS-Approximation können beispielsweise Knotenwerte an den Integrationspunkten oder umgekehrt Integrationspunktwerte an den Knoten approximiert werden. Um Ansätze unabhängig von ihrem Einsatzzweck beschreiben zu können ist es also nicht sinnvoll von Knoten und Integrationspunkten zu sprechen, da sich deren Rollen vertauschen können. Stattdessen werden die sich selbsterklärenden Begriffe *Stützpunkte* und *Auswertungspunkte* verwendet.

Alle in dieser Arbeit verwendeten Ansätze entsprechen einer allgemeinen Form, die im Abschnitt 5.1 angegeben ist.

Elementansätze beschreibt Abschnitt 5.2; Ansätze, die ohne Elemente auskommen, Abschnitt 5.5. Ein Vergleich beider erfolgt in Abschnitt 5.6.

Werden temporäre Elementnetze eingesetzt, so müssen die Knoten neu erzeugter Netze initialisiert werden. Hierzu müssen Zustandsgrößenwerte von Punkten, an denen diese Werte gespeichert sind, auf die Knoten des neuen Netzes übertragen werden. Ein ähnliches Problem stellt sich im Rahmen der FEM: Spannungen und

Verzerrungen werden an den Integrationspunkten der Elemente berechnet. Zur Erzeugung von grafischen Ausgaben mit stetigen Verläufen für diese Zustandsgrößen werden die Knotenwerte der Zustandsgrößen benötigt. Es bieten sich drei unterschiedliche Strategien an:

1. Die Punkte werden durch ein weiteres FE-Netz verbunden. Zumindest für den zweidimensionalen Fall stehen Netzgeneratoren zur Verfügung, die hochwertige Dreieckselementnetze erzeugen, die keine manuelle Nachbearbeitung mehr erfordern. Dieser Weg wird nicht verfolgt, da bei Punktmethoden der Vernetzungsaufwand gerade reduziert werden soll.
2. Es werden die im Abschnitt 5.5 beschriebenen elementfreien Ansätze verwendet.
3. Aus den ohnehin vorhandenen Elementansatzfunktionen des (neuen) FE-Netzes werden Ansätze konstruiert, die die Knotenwerte approximieren. Da im Gegensatz zu den Elementansätzen nicht Elementknotenwerte im Element interpoliert werden, sondern Punktwerte approximiert werden, um Elementknotenwerte zu erhalten, werden diese Ansätze hier als *Inverse Elementansätze* bezeichnet. Sie werden im Abschnitt 5.4 hergeleitet.

5.1 Allgemeine Form

Eine allgemeine Form, der alle in dieser Arbeit verwendeten Ansätze entsprechen, kann wie folgt angegeben werden:

$$\bar{f}(\mathbf{x}) = \sum_{k=1}^{\mathbf{N}_K(\mathbf{x})} L_k(\mathbf{x}) f_k = \underset{\langle \mathbf{N}_K \rangle}{\mathbf{L}^T(\mathbf{x})} \underset{\langle \mathbf{N}_K \rangle}{\mathbf{f}} \quad (5.1)$$

$$\frac{\partial \bar{f}(\mathbf{x})}{\partial x_i} = \sum_{k=1}^{\mathbf{N}_K(\mathbf{x})} \frac{\partial L_k(\mathbf{x})}{\partial x_i} f_k = \frac{\partial \underset{\langle \mathbf{N}_K \rangle}{\mathbf{L}^T(\mathbf{x})}}{\partial x_i} \underset{\langle \mathbf{N}_K \rangle}{\mathbf{f}} \quad (5.2)$$

Hierin sind $L_k(\mathbf{x})$ die Ansatzfunktionen, $\partial L_k(\mathbf{x})/\partial x_i$ die räumlichen Ansatzfunktionsableitungen, f_k die Ansatzfunktionskoeffizienten und $\mathbf{N}_K(\mathbf{x})$ die vom Ort \mathbf{x} abhängige Anzahl der Ansatzfunktionen und Stützpunkte. Die interpolierten oder approximierten Werte sind mit einem Querstrich gekennzeichnet.

Die Arbeit beschränkt sich auf Ansätze, bei denen alle Ansatzfunktionskoeffizienten

f_k die physikalische Bedeutung von Werten an den Stützstellen haben.¹ Weiterhin wird gefordert, dass die Summe der Ansatzfunktionen an jedem Ort 1 ergibt:

$$\sum_{k=1}^{N_K(\mathbf{x})} L_k(\mathbf{x}) = 1 \quad (5.3)$$

Diese Forderung wird auch als *Partition of Unity* bezeichnet. Sie stellt sicher, dass konstante Funktionen exakt wiedergegeben werden. Letzteres wird nachgewiesen, indem der Ansatz gemäß Gleichung 5.1 für eine konstante Funktion aufgestellt, der (konstante) Stützwert f_k ausgeklammert und Gleichung 5.3 berücksichtigt wird. Berechnungsverfahren, deren Ansätze Gleichung (5.3) erfüllen gehören zur Familie der *Partition of Unity Methods*.

Als unmittelbare Folgerung aus Gleichung (5.3) ergibt sich:

$$\sum_{k=1}^{N_K(\mathbf{x})} \frac{\partial L_k(\mathbf{x})}{\partial x_i} = 0 \quad (5.4)$$

Die vorstehende Gleichung kann genutzt werden um Ansatzfunktionsableitungen zu kontrollieren.

Enthält \mathbf{x}_i die globalen Koordinaten des Punktes \mathcal{P}_k , so ist im Allgemeinen

$$L_k(\mathbf{x}_i) \neq \delta_{ik} \quad . \quad (5.5)$$

Die Ansatzfunktionen sind also im Allgemeinen an den Stützpunkten nicht gleich 0 oder 1. Dies erschwert die Berücksichtigung von wesentlichen Randbedingungen und Kontaktbedingungen.

5.2 Elementansätze

Elementansätze haben die Form

$$\bar{f}(\mathbf{x}) = \sum_{k=1}^{N_K} L_k(\mathbf{x}) f_k = \mathbf{L}^T(\mathbf{x}) \mathbf{f} \quad (5.6)$$

$\langle N_K \rangle$ $\langle N_K \rangle$

¹Bei Verwendung *hierarchischer Ansatzfunktionen* wäre dies nicht mehr der Fall. Diese erhöhen die Rechengenauigkeit, indem sie die „konventionellen“ Ansatzfunktionen ergänzen. Das Netz muss dabei nicht verändert werden. Zu den zusätzlichen Ansatzfunktionen gehören zusätzliche Freiwerte, die sich größtenteils einer direkten physikalischen Interpretation entziehen.

worin $L_k(\mathbf{x})$ die Elementansatzfunktionen, f_k die Knotenwerte und N_K die Knotenanzahl im Element sind. Die Ansatzfunktionen nehmen jeweils genau an einem Knoten den Wert 1 und an den übrigen Knoten den Wert 0 an. Enthält \mathbf{x}_i die globalen Koordinaten des Knotens \mathcal{K}_i , so gilt:

$$L_k(\mathbf{x}_i) = \delta_{ik} \quad (5.7)$$

Hieraus ergibt sich, dass die Knotenwerte exakt abgebildet werden:

$$\bar{f}(\mathbf{x}_i) = \sum_{k=1}^{N_K} L_k(\mathbf{x}_i) f_k = \sum_{k=1}^{N_K} \delta_{ik} f_k = f_i \quad (5.8)$$

Wird die Beziehung (5.6) für N_P Punkte im Element (z.B. für die Integrationspunkte) aufgestellt, so lassen sich die N_P Gleichungen

$$\bar{f}(\mathbf{x}_p) = \sum_{k=1}^{N_K} L_k(\mathbf{x}_p) f_k = \mathbf{L}^T(\mathbf{x}_p) \mathbf{f} \quad , \quad p = 1, \dots, N_P \quad (5.9)$$

$\langle N_P \rangle$ $\langle N_K \rangle$ $\langle N_K \rangle$

zu einer Matrixgleichung zusammenfassen:

$$\underbrace{\begin{Bmatrix} \bar{f}(\mathbf{x}_1) \\ \bar{f}(\mathbf{x}_2) \\ \vdots \\ \bar{f}(\mathbf{x}_{N_P}) \end{Bmatrix}}_{\mathbf{\bar{f}} \langle N_P \rangle} = \underbrace{\begin{bmatrix} L_1(\mathbf{x}_1) & L_2(\mathbf{x}_1) & \cdots & L_{N_K}(\mathbf{x}_1) \\ L_1(\mathbf{x}_2) & L_2(\mathbf{x}_2) & \cdots & L_{N_K}(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ L_1(\mathbf{x}_{N_P}) & L_2(\mathbf{x}_{N_P}) & \cdots & L_{N_K}(\mathbf{x}_{N_P}) \end{bmatrix}}_{\mathbf{S} \langle N_P \times N_K \rangle} \underbrace{\begin{Bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N_K} \end{Bmatrix}}_{\mathbf{f} \langle N_K \rangle} \quad (5.10)$$

Die Matrix \mathbf{S} vermittelt eine Abbildung zwischen Elementknotenwerten und Werten an Punkten im Element. Für die Komponenten von \mathbf{S} gilt:

$$S_{pk} = L_k(\mathbf{x}_p) \quad (5.11)$$

Die Beziehung (5.10) wird in Abschnitt 5.4 genutzt, um eine inverse Abbildung herzuleiten.

5.2.1 Natürliche Elementkoordinaten

Die Elementansatzfunktionen lassen sich nur, wenn die Elementknoten auf den Kreuzungspunkten und die Elementränder auf den Linien eines regelmäßigen Netzes aus Koordinatenlinien liegen, effektiv und systematisch konstruieren. Daher lassen sich

die Elementansatzfunktionen nur dann in direkter Abhängigkeit von den globalen Koordinaten darstellen, wenn die Elementnetze orthogonal sind. Um auch Elementnetze verwenden zu können, bei denen die Elementränder schiefwinklig und/oder krummlinig sind, werden *natürliche Elementkoordinaten* eingeführt. Sie sind elementweise definiert und normiert. Die Abbildungen 5.1 und 5.2 zeigen an den

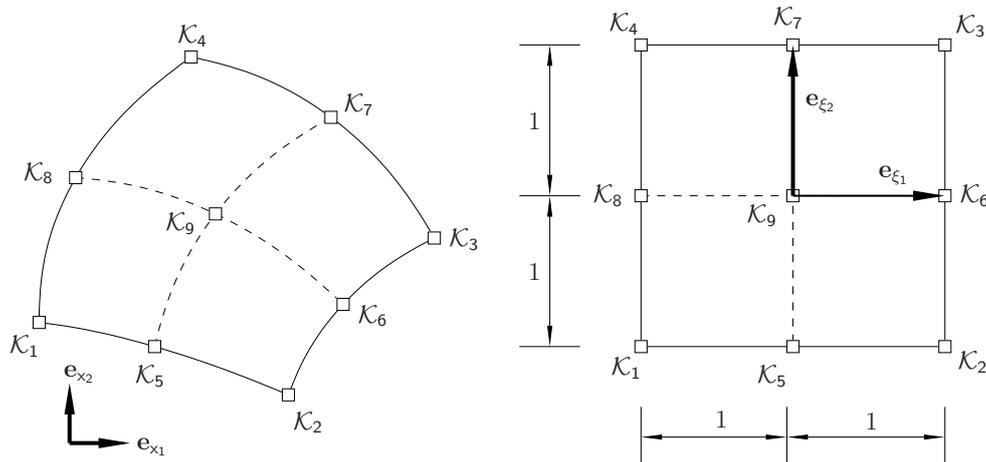


Abbildung 5.1: Krummlinig berandetes Viereckselement

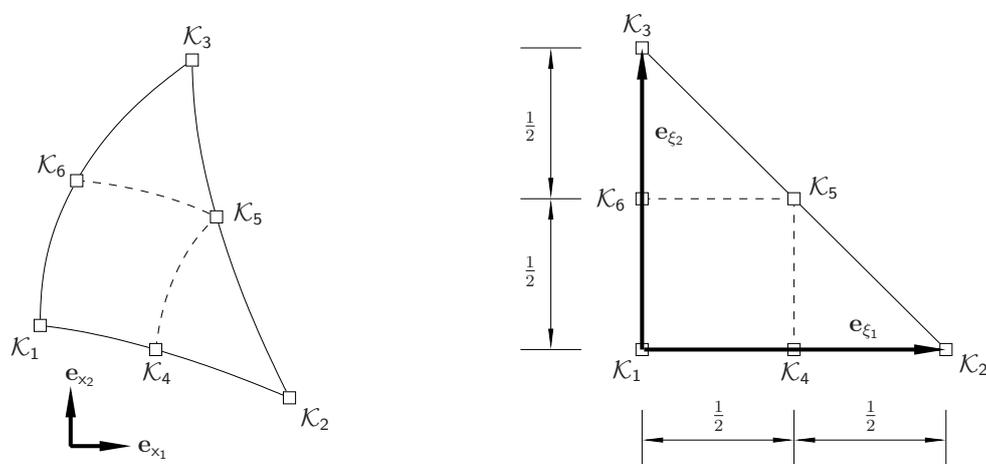


Abbildung 5.2: Krummlinig berandetes Dreieckselement

Beispielen des 9-Knoten-Viereckselementes und des 6-Knoten-Dreieckselementes die Beziehung zwischen dem globalen und dem lokalen normierten Koordinatensystem. Im durch die globale kartesische Basis aufgespannten Raum sind die Linien der globalen Koordinaten orthogonal und die Linien der natürlichen Elementkoordinaten im Allgemeinen schiefwinklig und krummlinig. Umgekehrt verhält es sich im durch die lokale Basis aufgespannten Raum. Hier sind die natürlichen Koordinaten orthogonal und die globalen Koordinaten im Allgemeinen schiefwinklig und krummlinig. Die Beziehung zwischen globalen und lokalen Koordinaten wird gewonnen, indem

die globalen Koordinaten im Element mittels der in Abhängigkeit von lokalen Koordinaten definierten Ansatzfunktionen interpoliert werden:

$$\underset{\langle \mathbf{N}_D \rangle}{\mathbf{x}(\boldsymbol{\xi})} = \underset{\langle \mathbf{N}_D \times \mathbf{N}_K \rangle}{\mathbf{X}_K} \underset{\langle \mathbf{N}_K \rangle}{\mathbf{L}(\boldsymbol{\xi})} \quad (5.12)$$

Die Matrix \mathbf{X}_K enthält die globalen Koordinaten der Elementknoten, \mathbf{N}_D ist die räumliche Dimension und \mathbf{N}_K die Anzahl der Elementknoten. Die Beziehung ist im Allgemeinen nichtlinear, sodass sie sich nicht direkt invertieren lässt. Im Abschnitt 5.3.2 wird beschrieben, wie sich die lokalen Elementkoordinaten in Abhängigkeit von den globalen Koordinaten bestimmen lassen. Die Schreibweise $\mathbf{L}(\mathbf{x})$ ist also im Falle der Elementansätze so zu verstehen, dass zunächst die Elementkoordinaten in Abhängigkeit von den globalen Koordinaten und anschließend die Elementansatzfunktionen in Abhängigkeit von den Elementkoordinaten berechnet werden.

Als Definitionsbereich der Ansatzfunktionen gilt für Linienelemente, Viereckselemente und Oktaederelemente

$$-1 \leq \xi_i \leq 1 \quad \text{mit } i = 1, \dots, \mathbf{N}_D \quad (5.13)$$

und für Dreieckselemente und Tetraederelemente

$$\begin{aligned} 0 \leq \xi_i \leq 1 \quad \text{mit } i = 1, \dots, \mathbf{N}_D \\ \text{und } \sum_{i=1}^{\mathbf{N}_D} \xi_i \leq 1 \quad . \end{aligned} \quad (5.14)$$

Die Elementkoordinaten der Mittelpunkte von Linienelementen, Viereckselementen und Oktaederelementen sind

$$\xi_{\mathcal{M}i} = 0 \quad \text{mit } i = 1, \dots, \mathbf{N}_D \quad , \quad (5.15)$$

die von Dreieckselementen sind

$$\xi_{\mathcal{M}i} = \frac{1}{3} \quad \text{mit } i = 1, \dots, 2 \quad (5.16)$$

und die von Tetraederelementen sind

$$\xi_{\mathcal{M}i} = \frac{1}{4} \quad \text{mit } i = 1, \dots, 3 \quad . \quad (5.17)$$

Werden zur Interpolation der globalen Koordinaten die gleichen Ansatzfunktionen verwendet wie zur Interpolation der Unbekannten, so wird das Element als isopara-

metrisch bezeichnet. Wird zur Beschreibung der Elementgeometrie eine niedrigere Ansatzfunktionsordnung verwendet als zur Beschreibung der Zustandsgrößenverläufe, so ist das Element subparametrisch; im umgekehrten Fall ist es superparametrisch.

Im Rahmen dieser Arbeit wurden Dreiecks- und Viereckselemente der *Lagrange*- und der *Serendipity*-Familie implementiert. Die Ansätze dieser Elementfamilien sind an den Elementgrenzen C^0 -stetig. Es handelt sich um Standardelementansätze, die z.B. in KNOTHE & WESSELS (1991), ZIENKIEWICZ & TAYLOR (2000a) oder BATHE (2002) beschrieben werden.

5.3 Punktzuordnung

Im Rahmen der klassischen FEM bewegen sich die Materialpunkte, die gleichzeitig auch Integrationspunkte sind, mit den Elementen mit, d.h. die Elementeinheitskoordinaten der Materialpunkte ändern sich nicht und die Punkte sind immer den selben Elementen zugeordnet.

Finden im Rahmen anderer Methoden Relativbewegungen zwischen Punkten und FE-Netz statt oder wird das FE-Netz neu generiert, so sind die Punkte nach jeder Relativbewegung oder Netzneugenerierung den Elementen neu zuzuordnen und die Elementkoordinaten zu bestimmen.

Für das Problem der Punktzuordnung wird der Spezialfall konvex polygonal begrenzter 2D-Elemente und der allgemeinere Fall beliebig begrenzter 2D- und 3D-Elemente untersucht. In die erste Kategorie fallen 3-Knoten-Dreieckselemente und nicht entartete 4-Knoten-Viereckselemente; in die zweite alle 2-D-Elemente mit höherer Ansatzfunktionsordnung für die Elementgeometrie und alle 3D-Elemente.

5.3.1 Konvex polygonal berandete Elemente

Die Abbildung 5.3 zeigt einen Punkt innerhalb eines konvexen Polygons. Ein Punkt \mathcal{P} befindet sich genau dann innerhalb eines Polygons mit den Eckpunkten \mathcal{K}_i und den Kantenvektoren \mathbf{k}_i , wenn alle Teilflächen A_i positiv sind ($i = 1, \dots, N_K$).

$$2 A_i \mathbf{e}_3 = \mathbf{k}_i \times \mathbf{a}_i = \begin{vmatrix} \mathbf{e}_3 & \mathbf{e}_3 & \mathbf{e}_3 \\ k_{i1} & k_{i2} & 0 \\ a_{i1} & a_{i2} & 0 \end{vmatrix} = [k_{i1} a_{i2} - k_{i2} a_{i1}] \mathbf{e}_3 \quad (5.18)$$

$$A_i = \frac{1}{2} [k_{i1} a_{i2} - k_{i2} a_{i1}] \quad (5.19)$$

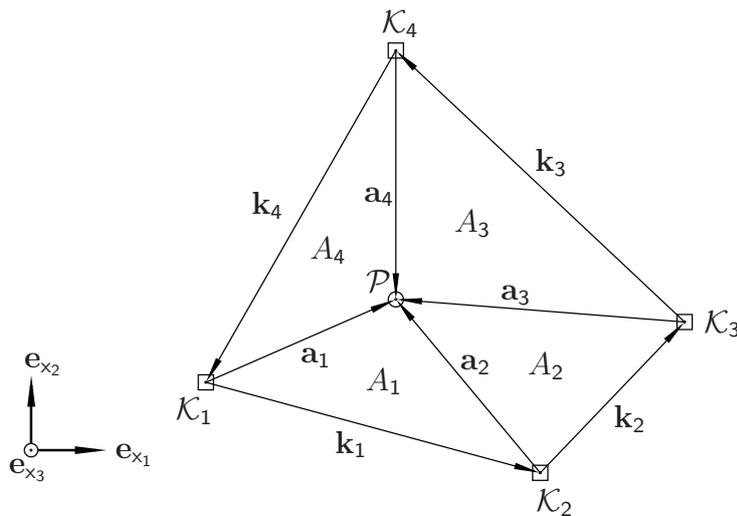


Abbildung 5.3: Punkt in konvex polygonal begrenztem Element

Der Algorithmus kann bei Netzen aus konvex polygonal begrenzten 2D-Elementen dazu dienen, numerisch effizient Punkte den Elementen zuzuordnen. In der Regel werden zusätzlich die Elementkoordinaten der Punkte benötigt. Zur Berechnung der Elementkoordinaten kann das im folgenden Abschnitt beschriebene Verfahren verwendet werden.

5.3.2 Krummlinig berandete Elemente

Im allgemeineren Fall von krummlinig berandeten Elementen ist es nicht mehr möglich die Elementzugehörigkeit von Punkten über eine einfache geometrische Bedingung zu entscheiden. Stattdessen wird die nichtlineare Beziehung (5.12) zwischen den normierten Elementkoordinaten ξ und den globalen Koordinaten \mathbf{x} eines Punktes ausgenutzt:

$$\mathbf{x}(\xi) = \mathbf{X}_K \mathbf{L}(\xi) \quad (5.20)$$

$\langle N_D \rangle \quad \langle N_D \times N_K \rangle \quad \langle N_K \rangle$

Die globalen Koordinaten des Punktes sind bekannt; die Elementkoordinaten des Punktes sind gesucht. Die Gleichung (5.20) lässt sich nur für Sonderfälle (z.B. 3-Knoten-Dreieckselement) invertieren. Stattdessen wird die Gleichung in Residualform gebracht

$$\mathbf{r}(\xi) = \mathbf{X}_K \mathbf{L}(\xi) - \mathbf{x} = 0 \quad (5.21)$$

und numerisch mit dem im Abschnitt 7.2.1 beschriebenen Newton-Raphson-Verfahren gelöst:

$$\begin{aligned} \mathbf{r}(\boldsymbol{\xi}^{i+1}) &= \mathbf{r}(\boldsymbol{\xi}^i + \Delta \boldsymbol{\xi}^{i+1}) \\ &\approx \mathbf{r}(\boldsymbol{\xi}^i) + \left. \frac{\partial \mathbf{r}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} \right|_{\boldsymbol{\xi}^i} \Delta \boldsymbol{\xi}^{i+1} \\ &\stackrel{!}{=} 0 \end{aligned} \quad (5.22)$$

Die Tangente wird als Jakobimatrix identifiziert:

$$\frac{\partial \mathbf{r}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \mathbf{X}_K \frac{\partial \mathbf{L}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \frac{\partial \mathbf{x}(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}} = \mathbf{J}(\boldsymbol{\xi}) \quad (5.23)$$

Mit den Gleichungen (5.22) und (5.23) ergibt sich als Verbesserung der lokalen Koordinaten im Iterationsschritt $i + 1$

$$\Delta \boldsymbol{\xi}^{i+1} = -\mathbf{J}(\boldsymbol{\xi}^i)^{-1} \mathbf{r}(\boldsymbol{\xi}^i) \quad (5.24)$$

und abschließend für die verbesserten lokalen Koordinaten

$$\boldsymbol{\xi}^{i+1} = \boldsymbol{\xi}^i + \Delta \boldsymbol{\xi}^{i+1} \quad (5.25)$$

Als Anfangswerte für das Iterationsverfahren werden die Einheitskoordinaten des Elementmittelpunktes entsprechend der Gleichungen (5.15) bis (5.17) verwendet. Die Iteration wird abgebrochen wenn

$$\|\mathbf{r}(\boldsymbol{\xi}^{i+1})\| < \varepsilon \quad (5.26)$$

erfüllt ist. ε ist eine kleine Zahl, die in Abhängigkeit von der Rechnergenauigkeit und der gewünschten Genauigkeit der Lösung festgelegt wird. Sind die Ungleichungen (5.13) bzw. (5.14) für die Lösung erfüllt, so liegt der Punkt im Element. Die Iteration wird vorzeitig abgebrochen, wenn entweder die Jakobimatrix nicht invertierbar ist oder in zwei aufeinander folgenden Iterationsschritten für den Punkt Elementkoordinaten außerhalb des durch die Ungleichungen (5.13) bzw. (5.14) definierten Bereiches erhalten werden. In beiden Fällen liegt der Punkt außerhalb des Elementes.

5.4 Inverse Elementansätze

Die Approximation von Elementknotenwerten aus an Punkten im Element vorliegenden Werten kann in folgender Weise erfolgen:

$$\bar{f}(\mathbf{x}_k) = \bar{f}_k = \sum_{p=1}^{N_P(\mathbf{x}_k)} L_p^*(\mathbf{x}_k) f_p = \mathbf{L}^{*\mathbf{T}}(\mathbf{x}_k) \mathbf{f} \quad (5.27)$$

Hierin sind f_p die Werte an den Stützpunkten und N_P ist die Stützpunktanzahl. Die Größen $L_p^*(\mathbf{x}_k)$ sind keine kontinuierlichen Ansatzfunktionen. Sie sind nur an den Knoten definiert.

Inverse Elementansätze können entweder elementweise über eine Invertierung oder gegebenenfalls Pseudoinvertierung der Beziehung (5.10) gewonnen werden oder knotenweise über die *Partition of Unity*-Forderung.

5.4.1 Herleitung über Pseudoinverse

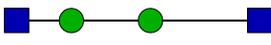
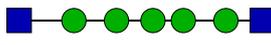
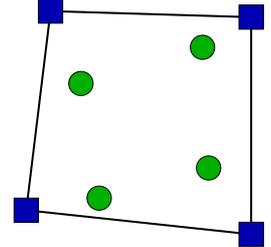
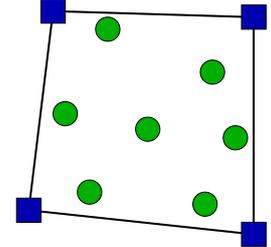
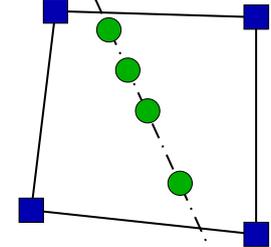
Fall 1	Fall 2	Fall 3
		
		
$N_P = N_K$ und $\text{Rang}(\mathbf{S}) = N_K$	$N_P > N_K$ und $\text{Rang}(\mathbf{S}) = N_K$	$N_P < N_K$ oder $\text{Rang}(\mathbf{S}) < N_K$
$\mathbf{S}^* = \mathbf{S}^{-1}$	$\mathbf{S}^* = \mathbf{S}^\dagger$	$\mathbf{S}^* = \mathbf{S}^{\dagger(\text{eff})}$

Tabelle 5.1: Fallunterscheidung bei der Bestimmung der (pseudo)inversen Elementansatzfunktionsmatrix am Beispiel des linearen 2-Knoten-Elementes und des bilinearen 4-Knoten-Elementes

In der Beziehung (5.10) in Abschnitt 5.2 enthält der Vektor \mathbf{f} Knotenwerte und der Vektor $\bar{\mathbf{f}}$ interpolierte Werte an Punkten im Element. Nun soll die umgekehrte Beziehung aufgestellt werden. Die Werte an den Punkten im Element sind Stützwerte und die Werte an den Elementknoten approximierte Werte. Daher wird nun in (5.10) der Vektor der approximierten Knotenwerte mit dem Querstrich gekennzeichnet:

$$\underset{\langle \mathbf{N}_P \rangle}{\mathbf{f}} = \underset{\langle \mathbf{N}_P \times \mathbf{N}_K \rangle}{\mathbf{S}} \underset{\langle \mathbf{N}_K \rangle}{\bar{\mathbf{f}}} \quad (5.28)$$

Die Matrix \mathbf{S} , die Werte von Ansatzfunktionen an den Orten der Punkte im Element enthält, bildet Werte an den Elementknoten auf Werte an den Punkten im Element ab. Die umgekehrte Abbildung wird über die Beziehung

$$\underset{\langle \mathbf{N}_K \rangle}{\bar{\mathbf{f}}} = \underset{\langle \mathbf{N}_K \times \mathbf{N}_P \rangle}{\mathbf{S}^*} \underset{\langle \mathbf{N}_P \rangle}{\mathbf{f}} \quad (5.29)$$

definiert, wobei noch geklärt werden muss, wie die Matrix \mathbf{S}^* berechnet werden kann. Es sind drei Fälle zu unterscheiden:

1. $\mathbf{N}_P = \mathbf{N}_K$ und $\text{Rang}(\mathbf{S}) = \mathbf{N}_K$:

Das Gleichungssystem (5.28) ist bestimmt. \mathbf{S}^* wird durch Invertierung von \mathbf{S} gewonnen:

$$\underset{\langle \mathbf{N}_K \times \mathbf{N}_K \rangle}{\mathbf{S}^*} = \underset{\langle \mathbf{N}_K \times \mathbf{N}_K \rangle}{\mathbf{S}^{-1}} \quad (5.30)$$

2. $\mathbf{N}_P > \mathbf{N}_K$ und $\text{Rang}(\mathbf{S}) = \mathbf{N}_K$:

Das Gleichungssystem (5.28) ist überbestimmt. \mathbf{S}^* wird durch Pseudoinvertierung von \mathbf{S} gemäß Gleichung (7.14) aus Abschnitt 7.1.2 gewonnen:

$$\underset{\langle \mathbf{N}_K \times \mathbf{N}_P \rangle}{\mathbf{S}^*} = \left[\underset{\langle \mathbf{N}_P \times \mathbf{N}_K \rangle}{\mathbf{S}} \right]^\dagger \quad (5.31)$$

3. $\mathbf{N}_P < \mathbf{N}_K$ oder $\text{Rang}(\mathbf{S}) < \mathbf{N}_K$:

Das Gleichungssystem (5.28) ist unterbestimmt. \mathbf{S}^* wird durch eine effektive Pseudoinvertierung von \mathbf{S} gemäß Gleichung (7.18) aus Abschnitt 7.1.3 gewonnen:

$$\underset{\langle \mathbf{N}_K \times \mathbf{N}_P \rangle}{\mathbf{S}^*} = \left[\underset{\langle \mathbf{N}_P \times \mathbf{N}_K \rangle}{\mathbf{S}} \right]^{\dagger(\text{eff})} \quad (5.32)$$

Die drei Fälle werden anhand der Beispiele des bilinearen 4-Knoten-Elementes und des linearen 2-Knoten-Elementes in Tabelle 5.1 illustriert.

Tabelle 5.2 zeigt am Beispiel des linearen 2-Knoten-Elementes wie Knotenwerte bei unterschiedlicher Stützpunktanzahl und Stützpunktposition approximiert werden,

Fall 1	Fall 2	Fall 3
$\mathbf{S}^* = \mathbf{S}^{-1}$	$\mathbf{S}^* = \mathbf{S}^\dagger$	$\mathbf{S}^* = \mathbf{S}^{\dagger(\text{eff})}$
Fall 3a	Fall 3b	Fall 3c
$\mathbf{S}^* = \mathbf{S}^{\dagger(\text{eff})}$	$\mathbf{S}^* = \mathbf{S}^{\dagger(\text{eff})}$	$\mathbf{S}^* = \mathbf{S}^{\dagger(\text{eff})}$

Tabelle 5.2: Eigenschaften der Approximation von Elementknotenwerten aus Werten an Punkten im Element über die (pseudo)inverse Elementansatzfunktionsmatrix am Beispiel des linearen 2-Knoten-Elementes

wenn hierzu Beziehung (5.29) mit (5.30), (5.31) oder (5.32) genutzt wird.

Um die nur an den Elementknoten definierten *inversen Elementansatzfunktionen* $L_p^*(\mathbf{x}_k)$ zu identifizieren, wird die Matrixgleichung (5.29) ausgeschrieben

$$\underbrace{\begin{Bmatrix} \bar{f}(\mathbf{x}_1) \\ \bar{f}(\mathbf{x}_2) \\ \vdots \\ \bar{f}(\mathbf{x}_{N_K}) \end{Bmatrix}}_{\mathbf{\bar{f}}_{\langle N_K \rangle}} = \underbrace{\begin{bmatrix} S_{11}^* & S_{12}^* & \cdots & S_{1N_P}^* \\ S_{21}^* & S_{22}^* & \cdots & S_{2N_P}^* \\ \vdots & \vdots & \ddots & \vdots \\ S_{N_K 1}^* & S_{N_K 2}^* & \cdots & S_{N_K N_P}^* \end{bmatrix}}_{\mathbf{S}^*_{\langle N_K \times N_P \rangle}} \underbrace{\begin{Bmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N_P} \end{Bmatrix}}_{\mathbf{f}_{\langle N_P \rangle}} \quad (5.33)$$

und mit der für alle Elementknoten aufgestellten Beziehung (5.27)

$$\bar{f}(\mathbf{x}_k) = \bar{f}_k = \sum_{p=1}^{N_P(\mathbf{x}_k)} L_p^*(\mathbf{x}_k) f_p = \mathbf{L}^{*\mathbf{T}}(\mathbf{x}_k) \mathbf{f} \quad , \quad k = 1, \dots, N_K \quad (5.34)$$

verglichen. Hieraus ergibt sich

$$\mathbf{S}^*_{\langle N_K \times N_P \rangle} = \begin{bmatrix} L_1^*(\mathbf{x}_1) & L_2^*(\mathbf{x}_1) & \cdots & L_{N_P}^*(\mathbf{x}_1) \\ L_1^*(\mathbf{x}_2) & L_2^*(\mathbf{x}_2) & \cdots & L_{N_P}^*(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ L_1^*(\mathbf{x}_{N_K}) & L_2^*(\mathbf{x}_{N_K}) & \cdots & L_{N_P}^*(\mathbf{x}_{N_K}) \end{bmatrix} \quad (5.35)$$

oder kürzer

$$L_p^*(\mathbf{x}_k) = S_{kp}^* \quad (5.36)$$

Grenzen an einen Knoten mehrere Elemente, so werden im Allgemeinen für diesen Knoten in jedem Element andere Knotenwerte durch die Approximation erhalten.

² Wird eine kontinuierliche Approximation der Knotenwerte benötigt, so sind die Knotenwerte noch zu mitteln. Dies lässt sich vermeiden, wenn die inversen Elementansatzfunktionen wie im nächsten Abschnitt beschrieben bestimmt werden.

5.4.2 Herleitung über *Partition of Unity*-Forderung

Als Bausteine für die inversen Elementansatzfunktionen werden aus Elementansatzfunktionen knotenweise zusammengefaßte Funktionen konstruiert. Alle Elementansatzfunktionen, die am Knoten mit dem Index k den Wert 1 haben, bilden die zusammengesetzte Funktion \hat{L}_k . Die zusammengesetzte Funktion nimmt nur in Elementen, die an den Knoten mit dem Index k angrenzen, von 0 verschiedene Werte an.

Für die inversen Elementansatzfunktionen wird gefordert, dass diese die *Partition of Unity*-Forderung (5.3) erfüllen:

$$\sum_{p=1}^{N_P(\mathbf{x}_k)} L_p^*(\mathbf{x}_k) = 1 \quad (5.37)$$

²Bei der Visualisierung über Farbverläufe von im Rahmen der Finite-Elemente-Methode an Integrationspunkten gewonnenen Berechnungsergebnissen (wie z.B. Spannungen oder Dehnungen) können die approximierten Knotenwerte direkt zur elementweisen Darstellung der Ergebnisse genutzt werden. Die Farbverläufe sind dann im Allgemeinen an den Elementrändern diskontinuierlich. Starke Farbsprünge an den Elementrändern zeigen größere Sprünge in den Berechnungsergebnissen an, was auf eine zu grobe Diskretisierung in den entsprechenden Bereichen hindeutet.

Hierin ist $N_P(\mathbf{x}_k)$ die Punktzahl in den an den Knoten mit dem Index k angrenzenden Elementen.

Ausgegangen wird von einem Quotienten mit identischem Zähler und Nenner. Der Bruch wird in eine Summe von $N_P(\mathbf{x}_k)$ Brüchen zerlegt und mit (5.37) verglichen.

$$1 = \frac{\sum_{p=1}^{N_P(\mathbf{x}_k)} V_p \hat{L}_k(\mathbf{x}_p)}{\sum_{q=1}^{N_P(\mathbf{x}_k)} V_q \hat{L}_k(\mathbf{x}_q)} = \sum_{p=1}^{N_P(\mathbf{x}_k)} \left[\frac{V_p \hat{L}_k(\mathbf{x}_p)}{\sum_{q=1}^{N_P(\mathbf{x}_k)} V_q \hat{L}_k(\mathbf{x}_q)} \right] \stackrel{!}{=} \sum_{p=1}^{N_P(\mathbf{x}_k)} L_p^*(\mathbf{x}_k) \quad (5.38)$$

$$\rightarrow L_p^*(\mathbf{x}_k) = \frac{V_p \hat{L}_k(\mathbf{x}_p)}{\sum_{q=1}^{N_P(\mathbf{x}_k)} V_q \hat{L}_k(\mathbf{x}_q)} \quad (5.39)$$

Die zusammengesetzten Funktionen \hat{L}_k werden mit den Volumina, die den Punkten zugeordnet sind, gewichtet. Werden Knotenwerte über den Ansatz (5.27) mit $L_p^*(\mathbf{x}_k)$ nach (5.39) approximiert, so werden eindeutige Knotenwerte erhalten. Eine Mittelung von mehreren Knotenwerten wie beim Vorgehen nach Abschnitt 5.4.1 entfällt.

5.5 Ansätze für Punktnetze

Als elementfreie räumliche Diskretisierung wird in dieser Arbeit die in Abschnitt 5.5.3 beschriebene MLS-Approximation verwendet (MLS = Moving Least Squares, Methode der bewegten kleinsten Fehlerquadrate). Zunächst werden deren Bestandteile angegeben. Dies sind die im Abschnitt 5.5.1 behandelten Wichtungsfunktionen und die im Abschnitt 5.5.2 behandelten Basisfunktionen.

5.5.1 Wichtungsfunktionen

Die meisten netzfreien Methoden nutzen als Bausteine für Ansatzfunktionen Wichtungsfunktionen. Wichtungsfunktionen sorgen dafür, dass der Einfluss eines Punktes mit zunehmendem Abstand abnimmt. Die Ausdehnung der Wichtungsfunktionen bestimmt die Anzahl der Stützpunkte $N_K(\mathbf{x})$ der Approximation.³

³Werden die Einflussbereiche der Wichtungsfunktionen nicht begrenzt, so ergeben sich globale Ansätze. Diese Ansätze führen zu einem dicht besetzten Gleichungssystem, wenn sie im Rahmen einer numerische Berechnungsmethode zur räumlichen Diskretisierung eingesetzt werden.

	Exponential- funktion	Polynom 2n-ten Grades
$\tilde{w}(s)$	$e^{-c^2 \left[\frac{s}{h}\right]^2}$	$\left[1 - \left[\frac{s}{h}\right]^2\right]^n$
β_{1D}	$\left[\frac{\operatorname{erf}(c) \sqrt{\pi} h}{c}\right]^{-1}$	$\left[\frac{\Gamma(n+1) h \sqrt{\pi}}{\Gamma(n+\frac{3}{2})}\right]^{-1}$
β_{2D}	$\left[\left[1 - e^{-c^2}\right] \frac{h^2}{c^2} \pi\right]^{-1}$	$\left[\frac{h^2 \pi}{n+1}\right]^{-1}$

Tabelle 5.3: Wichtungsfunktionen und Faktoren zur Normierung

Werden die Wichtungsfunktionen in Abhängigkeit vom Abstand vom Basispunkt \mathcal{P}_k

$$s = \|x - x_k\| \quad (5.40)$$

dargestellt, so ergeben sich im eindimensionalen Fall achsensymmetrische, im zweidimensionalen Fall rotationssymmetrische und im dreidimensionalen Fall kugelsymmetrische Wichtungsfunktionen bzgl. \mathcal{P}_k .

Wichtungsfunktionen sollten nur im Einflussbereich des Basispunktes von Null verschiedene Werte annehmen

$$w(s) \begin{cases} \neq 0 & \text{für } s < h \\ = 0 & \text{für } s \geq h \end{cases}, \quad (5.41)$$

mit zunehmendem Abstand abnehmen

$$w(s_2) \leq w(s_1) \quad \text{für } s_2 > s_1, \quad (5.42)$$

möglichst glatt (mindestens C^1 -stetig) und normiert sein

$$\int_{\mathcal{B}} w(s) dV = 1 \quad (5.43)$$

(\mathcal{B} symbolisiert den Definitionsbereich).

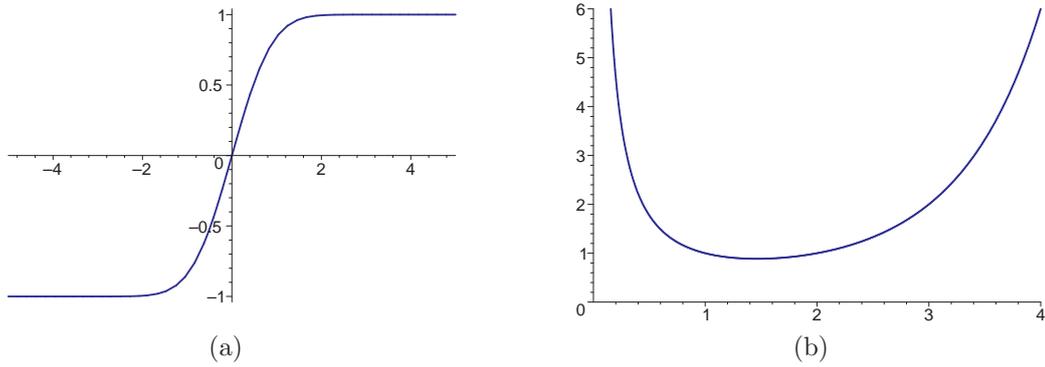


Abbildung 5.4: Fehlerfunktion (a) und Gammafunktion (b)

Die Normierungsbedingung für den eindimensionalen Fall lautet

$$2 \int_{s=0}^h w_{1D}(s) ds = 1 \quad , \quad (5.44)$$

woraus sich als Normierungsvorschrift

$$w_{1D}(s) = \beta_{1D} \tilde{w}(s) \quad (5.45)$$

$$\text{mit } \beta_{1D} = \left[2 \int_{s=0}^h \tilde{w}(s) ds \right]^{-1} \quad (5.46)$$

ergibt. $\tilde{w}(s)$ ist hierin eine nicht normierte Funktion. Die Normierungsbedingung für den zweidimensionalen Fall ist

$$\int_{s=0}^h \int_{\varphi=0}^{2\pi} w_{2D}(s) s d\varphi ds = 2\pi \int_{s=0}^h w_{2D}(s) s ds = 1 \quad . \quad (5.47)$$

Als Normierungsvorschrift ergibt sich

$$w_{2D}(s) = \beta_{2D} \tilde{w}(s) \quad (5.48)$$

$$\text{mit } \beta_{2D} = \left[2\pi \int_{s=0}^h \tilde{w}(s) s ds \right]^{-1} \quad . \quad (5.49)$$

Häufig werden als Wichtungsfunktionen Exponentialfunktionen oder Polynome $2n$ -ten Grades verwendet. In Tabelle (5.3) sind die nichtnormierten Funktionen und die Faktoren β_{1D} und β_{2D} zur Normierung für den ein- und zweidimensionalen Fall angegeben. $\operatorname{erf}(\cdot)$ ist die Fehlerfunktion und $\Gamma(\cdot)$ die Gammafunktion (siehe Abbil-

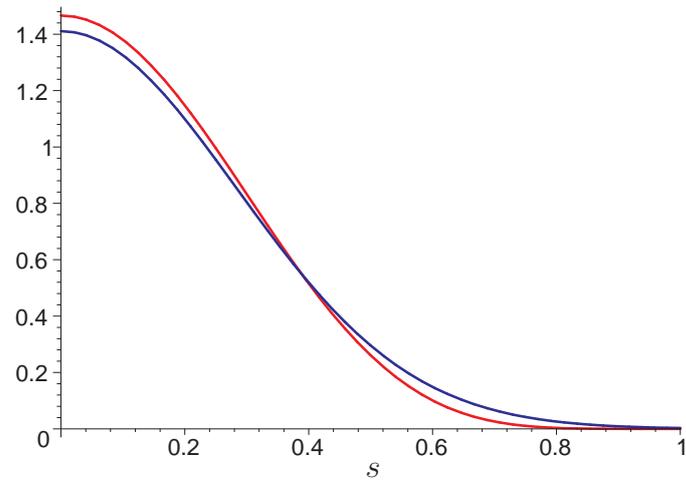
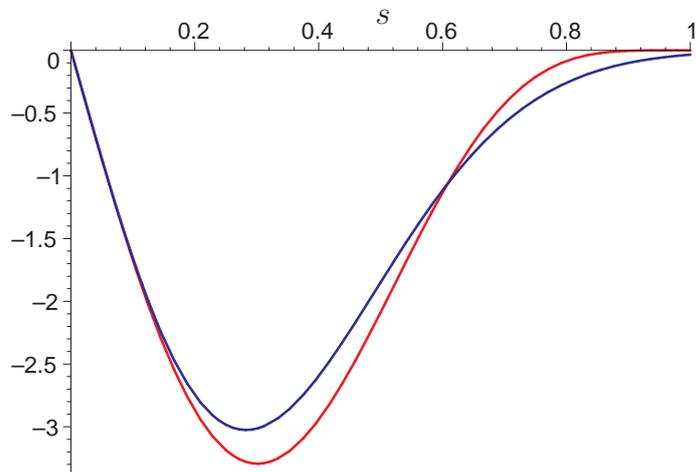
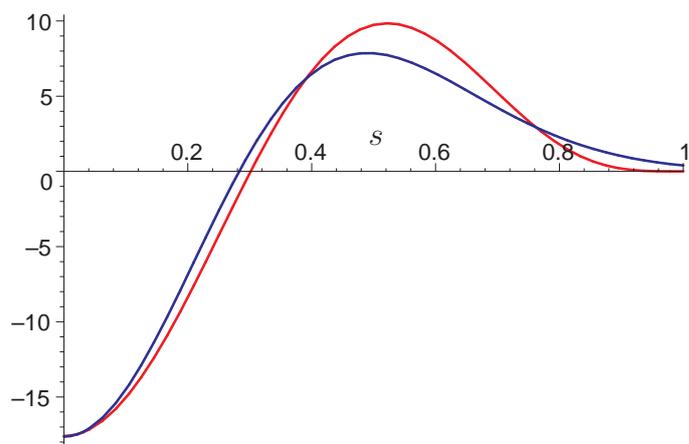
(a) w_{exp} (blau) und w_{p2n} (rot)(b) $\partial w_{\text{exp}} / \partial s$ (blau) und $\partial w_{\text{p2n}} / \partial s$ (rot)(c) $\partial^2 w_{\text{exp}} / \partial s^2$ (blau) und $\partial^2 w_{\text{p2n}} / \partial s^2$ (rot)

Abbildung 5.5: Normierte Wichtungsfunktionen (Exponentialfunktion und Polynom $2n$ -ten Grades) und Ableitungen, 1D, Funktionsparameter $c = \frac{5}{2}$ und $n = 6$, Einflussbereich $h = 1$

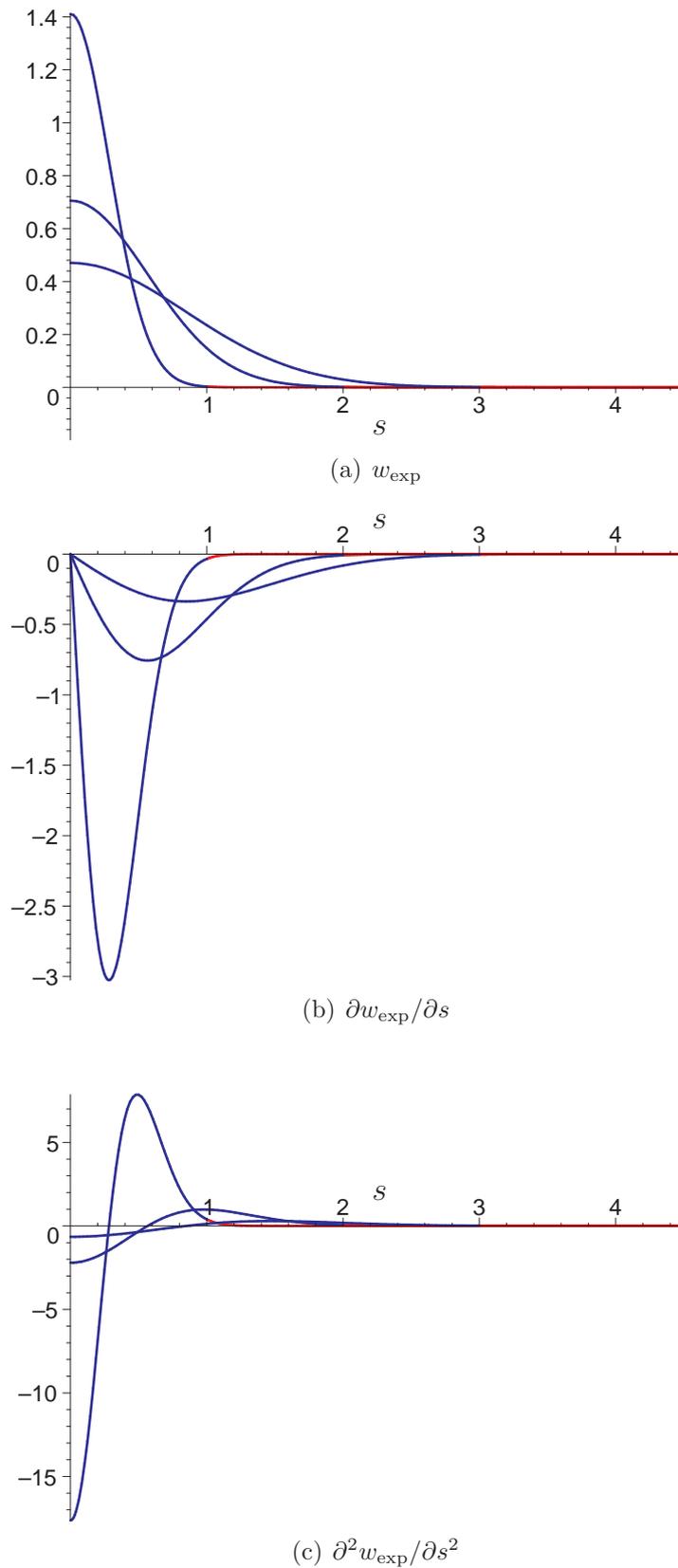


Abbildung 5.6: Normierte Wichtungsfunktion (Exponentialfunktion) und Ableitungen, 1D, Funktionsparameter $c = \frac{5}{2}$, Einflussbereich $h \in \{1, 2, 3\}$

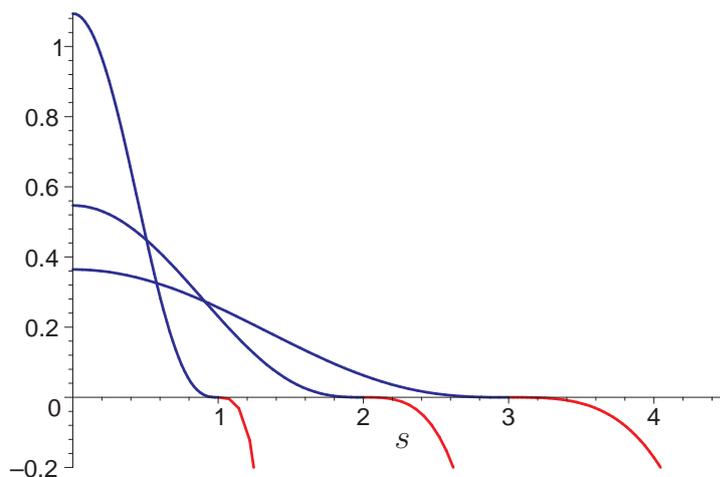
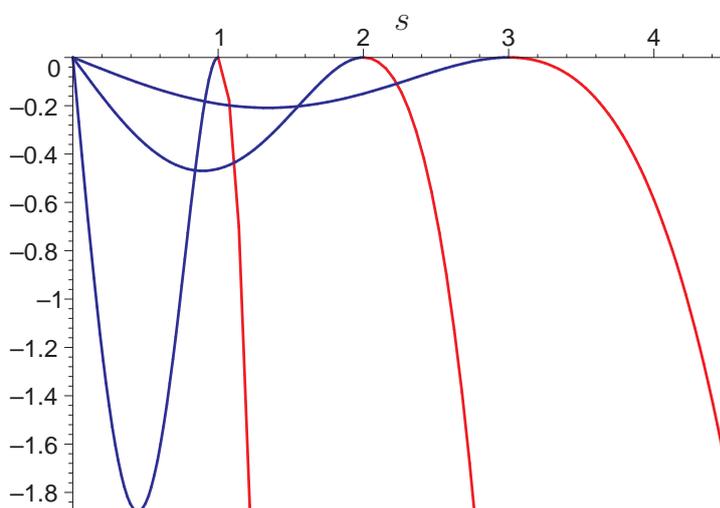
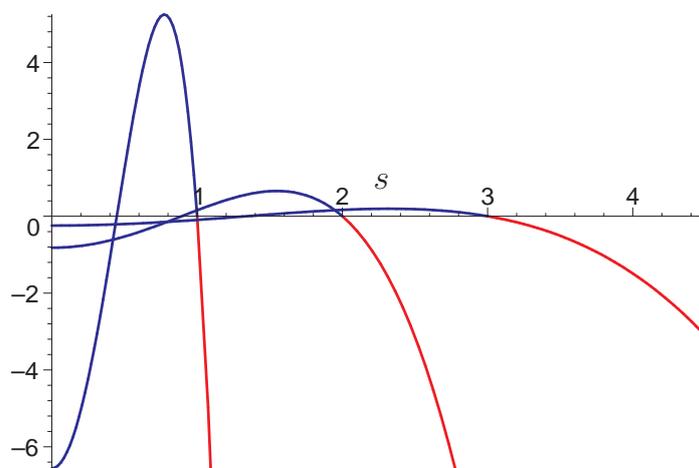
(a) w_{p2n} (b) $\partial w_{p2n} / \partial s$ (c) $\partial^2 w_{p2n} / \partial s^2$

Abbildung 5.7: Normierte Wichtungsfunktion (Polynom $2n$ -ten Grades) und Ableitungen, 1D, Funktionsparameter $n = 6$, Einflussbereich $h \in \{1, 2, 3\}$

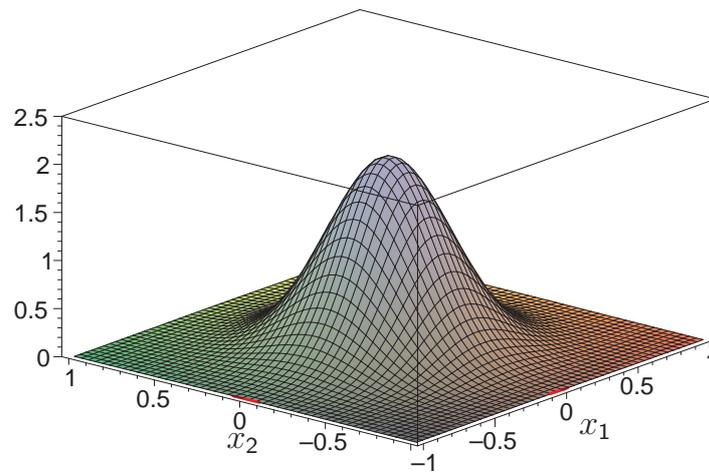
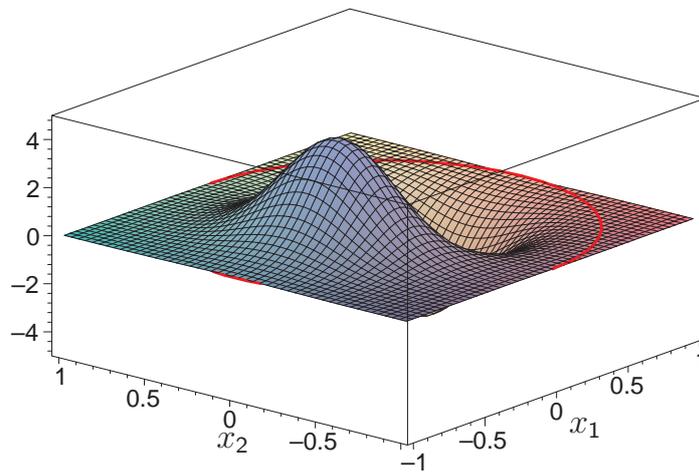
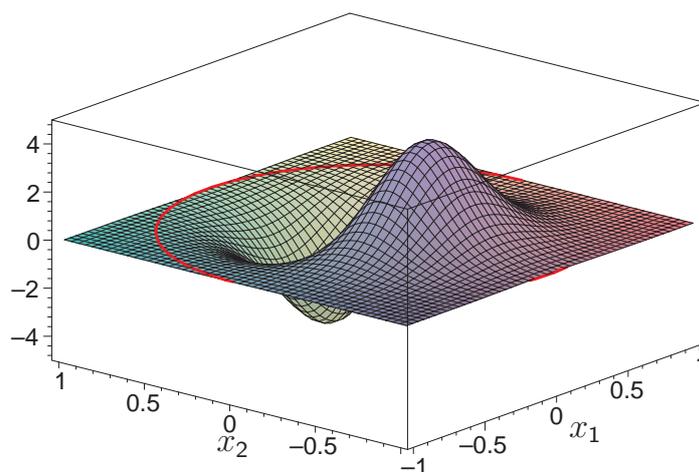
(a) w_{exp} (b) $\partial w_{\text{exp}}/\partial x_1$ (c) $\partial w_{\text{exp}}/\partial x_2$

Abbildung 5.8: Normierte Wichtungsfunktion (Exponentialfunktion) und Ableitungen, 2D, Funktionsparameter $c = \frac{5}{2}$, Einflussbereich $h = 1$

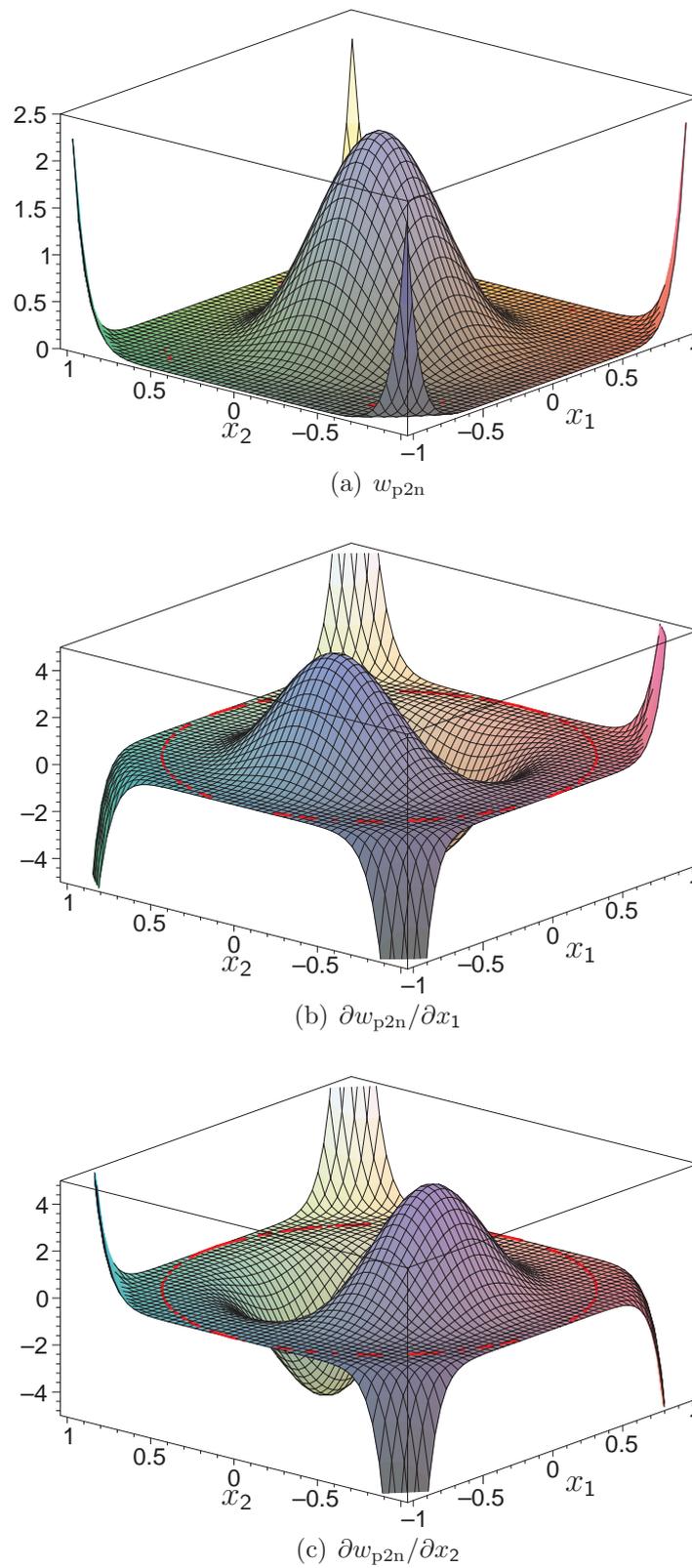


Abbildung 5.9: Normierte Wichtungsfunktion (Polynom $2n$ -ten Grades) und Ableitungen, 2D, Funktionsparameter $n = 6$, Einflussbereich $h = 1$

Dimension N_D	1 (Fläche A)	2 (Dicke d)	3
Gitter	äquidistant	quadratisch, äquidistant	kubisch, äquidistant
V_k	$a A$ $= h_0 A$	$a^2 d$ $= \pi h_0^2 d$	a^3 $= \frac{4}{3}\pi h_0^3$
h_0	a	$\frac{1}{\sqrt{\pi}} a = 0.564190a$	$\frac{1}{\sqrt[3]{\frac{4}{3}\pi}} a = 0.620350a$

Tabelle 5.4: Grundwerte der Wichtungsfunktionsbereiche h_0 für regelmäßige Punktgitter mit Punktabstand a

dung 5.4).

Die normierten eindimensionalen Wichtungsfunktionen zeigen die Abbildungen 5.5 bis 5.7; die normierten zweidimensionalen Wichtungsfunktionen die Abbildungen 5.8 und 5.9.

Bei der Parameterwahl $c = \frac{5}{2}$ und $n = 6$ haben die normierte Exponentialfunktion und das normierte Polynom $2n$ -ten Grades für $s < h$ eine sehr ähnliche Gestalt, wie in Abbildung 5.5 zu erkennen ist. Ein wesentlicher Unterschied besteht darin, dass das Polynom $2n$ -ten Grades für Argumente $s = h$ exakt verschwindet und außerhalb des Einflussbereiches h große Werte annimmt. Diese Funktion ist daher immer für $s > h$ abzuschneiden. Die Exponentialfunktion verschwindet für $s = h$ nur näherungsweise aber geht für $s \rightarrow \infty$ streng monoton fallend gegen Null, so dass es nicht zwingend erforderlich ist, diese Funktion für $s > h$ abzuschneiden. (Die Exponentialfunktion wird lediglich abgeschnitten, um lokale Ansätze zu erhalten.) Das Polynom $2n$ -ten Grades ist n -fach stetig differenzierbar, die Exponentialfunktion ∞ -fach, sofern sie nicht abgeschnitten wird.

Mit dem Parameter h wird festgelegt wie stark die Wichtungsfunktionen benachbarter Punkte überlappen. h wird über einen dimensionslosen Parameter α angepaßt:

$$h = \alpha h_0 \quad (5.50)$$

h_0 ist der Grundwert für den Wichtungsfunktionseinflussbereich. Liegen regelmäßige Punktraster vor, so kann h_0 nach Tabelle 5.4 berechnet werden. Andernfalls müssen zunächst, z.B durch die Bildung eines *Voronoi-Diagrammes*, die den Punkten

Dimension N_D	1 (Fläche: A)	2 (Dicke: d)	3
V_k	$h_0 A$	$\pi h_0^2 d$	$\frac{4}{3}\pi h_0^3$
h_0	$\frac{V_k}{A}$	$\sqrt{\frac{V_k}{\pi d}} = 0.564190\sqrt{\frac{V_k}{d}}$	$\sqrt[3]{\frac{V_k}{\frac{4}{3}\pi}} = 0.620350\sqrt[3]{V_k}$

Tabelle 5.5: Grundwerte der Wichtungsfunktionsbereichsparameter h_0 für unregelmäßige Punktgitter

zugeordneten Volumina berechnet werden. Aus diesen ergibt sich h_0 nach Tabelle 5.5.

Bei $\alpha = 1$ überlappen die Wichtungsfunktionen gerade nicht (im 2- und 3-dimensionalen Fall nur näherungsweise, siehe Abbildung 9.3).

Sinnvolle Überlappungsparameter α werden im Abschnitt 9.3.1 numerisch bestimmt und in der Tabelle 9.2 für unterschiedliche MLS-Ansatzfunktionsordnungen angegeben.

Um eine Wichtungsfunktion für einen Punkt \mathcal{P}_k zu erhalten, wird die normierte Wichtungsfunktion mit dem Volumen V_k , das dem Punkt zugeordnet ist, multipliziert. Da eine Funktion in Abhängigkeit von den globalen Koordinaten benötigt wird, ist außerdem Beziehung (5.40) zu berücksichtigen.

$$w_k(\mathbf{x}) = V_k w(s) \quad \text{mit} \quad s = \|\mathbf{x} - \mathbf{x}_k\| \quad (5.51)$$

5.5.2 Basisfunktionen

Neben den Wichtungsfunktionen wird für die MLS-Approximation eine Funktionsbasis benötigt. In der Funktionsbasis enthaltene Funktionen werden durch die MLS-Approximation exakt wiedergegeben, sofern die Stützpunktanzahl hinreichend groß ist. Durch die Wahl der Funktionsbasis können die Ansatzfunktionen an unterschiedliche Erfordernisse angepasst werden. Geeignete Funktionsbasen für zweidimensionale Berechnungen sind in Tabelle 5.6 angegeben. Ist der Polynomgrad N_G , so sind in der Funktionsbasis alle Monome der ersten $N_G + 1$ Zeilen des Pascalschen Dreiecks enthalten.

Tabelle 5.7 zeigt, dass die Anzahl der Basisfunktionen N_B im zwei- und dreidimensionalen Fall rasch mit dem Polynomgrad N_G ansteigt.

Polynomgrad N_G	Anzahl Basis- funktionen N_B	Funktionsbasis $\{b_i\}, i = 1, \dots, N_B$
0	1	1
1	3	$1, x_1, x_2$
2	6	$1, x_1, x_2, x_1^2, x_1x_2, x_2^2$
3	10	$1, x_1, x_2, x_1^2, x_1x_2, x_2^2, x_1^3, x_1^2x_2, x_1x_2^2, x_2^3$
\vdots	\vdots	\vdots

Tabelle 5.6: Funktionsbasen im Zweidimensionalen bis zum Polynomgrad $N_G = 3$

Polynomgrad N_G	Dimension N_D		
	1	2	3
0	1	1	1
1	2	3	4
2	3	6	10
3	4	10	20
4	5	15	35
n	$\binom{n+1}{1}$	$\binom{n+2}{2}$	$\binom{n+3}{3}$

Tabelle 5.7: Anzahl Basisfunktionen N_B in Abhängigkeit von Polynomgrad N_G und der räumlichen Dimension N_D

5.5.3 Approximation über Methode der bewegten kleinsten Fehlerquadrate

Die Ansatzfunktionen werden wie folgt konstruiert:

$$L_k(\mathbf{x}) = \sum_{i=1}^{N_B} \alpha_i(\mathbf{x}) b_i(\mathbf{x}_k) w_k(\mathbf{x}) = \underset{\langle N_B \rangle}{\boldsymbol{\alpha}^T(\mathbf{x})} \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x}_k)} w_k(\mathbf{x}) \quad (5.52)$$

Hierin sind die Wichtungsfunktion w_k des Knoten \mathcal{K} und die Koeffizienten α_i am Ort \mathbf{x} und die Basisfunktionen b_i am Ort \mathbf{x}_k des Knoten \mathcal{K} auszuwerten ($i = 1, \dots, N_B$).

Es wird gefordert, dass alle Funktionen der Funktionsbasis selbst exakt approximiert werden.

$$b_i(\mathbf{x}) = \sum_{k=1}^{N_K(\mathbf{x})} L_k(\mathbf{x}) b_i(\mathbf{x}_k) \quad , \quad i = 1, \dots, N_B \quad (5.53)$$

Einsetzen von (5.52) in (5.53) ergibt:

$$b_i(\mathbf{x}) = \sum_{k=1}^{N_K(\mathbf{x})} \boldsymbol{\alpha}^T(\mathbf{x}) \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x}_k)} w_k(\mathbf{x}) \underset{\langle N_B \rangle}{b_i(\mathbf{x}_k)} \quad (5.54)$$

Zusammenfassen von N_B Gleichungen und anschließendes Umordnen liefert:

$$\begin{aligned} \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x})} &= \sum_{k=1}^{N_K(\mathbf{x})} \boldsymbol{\alpha}^T(\mathbf{x}) \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x}_k)} w_k(\mathbf{x}) \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x}_k)} \\ &= \sum_{k=1}^{N_K(\mathbf{x})} \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x}_k)} w_k(\mathbf{x}) \underset{\langle N_B \rangle}{\mathbf{b}^T(\mathbf{x}_k)} \underset{\langle N_B \rangle}{\boldsymbol{\alpha}(\mathbf{x})} \\ &= \underset{\langle N_B \times N_K \rangle}{\mathbf{B}} \underset{\langle N_K \times N_K \rangle}{\mathbf{W}(\mathbf{x})} \underset{\langle N_K \times N_B \rangle}{\mathbf{B}^T} \underset{\langle N_B \rangle}{\boldsymbol{\alpha}(\mathbf{x})} \end{aligned} \quad (5.55)$$

Für die symmetrische Größe $\mathbf{B} \mathbf{W} \mathbf{B}^T$ wird eine Abkürzung eingeführt:

$$\underset{\langle N_B \times N_B \rangle}{\mathbf{A}(\mathbf{x})} = \underset{\langle N_B \times N_K \rangle}{\mathbf{B}} \underset{\langle N_K \times N_K \rangle}{\mathbf{W}(\mathbf{x})} \underset{\langle N_K \times N_B \rangle}{\mathbf{B}^T} = \underset{\langle N_B \times N_B \rangle}{\mathbf{A}^T(\mathbf{x})} \quad (5.56)$$

Gleichung (5.55) wird nach dem Vektor der Koeffizienten der Basisfunktionen $\boldsymbol{\alpha}$ umgestellt. Unter Berücksichtigung von (5.56) ergibt sich:

$$\underset{\langle N_B \rangle}{\boldsymbol{\alpha}(\mathbf{x})} = \underset{\langle N_B \times N_B \rangle}{\mathbf{A}^{-1}(\mathbf{x})} \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x})} = \underset{\langle N_B \rangle}{\mathbf{b}^T(\mathbf{x})} \underset{\langle N_B \times N_B \rangle}{\mathbf{A}^{-1}(\mathbf{x})} \quad (5.57)$$

Mit (5.52) und (5.57) wird schließlich eine Bestimmungsgleichung für die Ansatzfunktionen erhalten.

$$L_k(\mathbf{x}) = \underset{\langle N_B \rangle}{\mathbf{b}^T(\mathbf{x})} \underset{\langle N_B \times N_B \rangle}{\mathbf{A}^{-1}(\mathbf{x})} \underset{\langle N_B \rangle}{\mathbf{b}(\mathbf{x})} w_k(\mathbf{x}) \quad (5.58)$$

Der Rechenaufwand für die Inversion der Matrix \mathbf{A} steigt mit zunehmender Ansatzfunktionsordnung und damit zunehmender Größe der Funktionsbasis. Ebenso wächst die Anzahl der benötigten Stützpunkte. Die Lage der Stützpunkte beeinflusst die Kondition der Matrix \mathbf{A} und damit die Güte des Rechenergebnisses.

Die Ansatzfunktionsordnung wird bei zu geringer Stützpunktanzahl reduziert. Außerdem wird die Genauigkeit der Inversion von \mathbf{A} nach (7.19) und (7.20) überprüft

und die Berechnung der Ansatzfunktionen gegebenenfalls mit niedrigerem Polynomgrad wiederholt.

Bei der Bildung der Ansatzfunktionsableitungen ist die Produktregel zu beachten.

$$\begin{aligned} \frac{\partial L_k(\mathbf{x})}{\partial x_i} &= \frac{\partial \mathbf{b}^T(\mathbf{x})}{\partial x_i} \mathbf{A}^{-1}(\mathbf{x}) \mathbf{b}(\mathbf{x}_k) w_k(\mathbf{x}) \\ &+ \mathbf{b}^T(\mathbf{x}) \frac{\partial \mathbf{A}^{-1}(\mathbf{x})}{\partial x_i} \mathbf{b}(\mathbf{x}_k) w_k(\mathbf{x}) \\ &+ \mathbf{b}^T(\mathbf{x}) \mathbf{A}^{-1}(\mathbf{x}) \mathbf{b}(\mathbf{x}_k) \frac{\partial w_k(\mathbf{x})}{\partial x_i} \end{aligned} \quad (5.59)$$

Die Berechnung der Ansatzfunktionen wird numerisch verbessert, wenn die Basisfunktionen b_i nicht in Abhängigkeit von globalen Koordinaten sondern in Abhängigkeit von lokalen, skalierten Koordinaten berechnet werden. Das lokale Koordinatensystem hat seinen Ursprung im Punkt \mathbf{x} , für den die MLS-Approximation berechnet wird. Formal sind in den Gleichungen (5.52 bis 5.59) die Basisfunktionswerte $b_i(\mathbf{x}_k)$ durch $\tilde{b}_i(\mathbf{x}_k, \mathbf{x})$ und $b_i(\mathbf{x})$ durch $\tilde{b}_i(\mathbf{x}, \mathbf{x})$ zu ersetzen.

$$b_i(\mathbf{x}_k) \longrightarrow \tilde{b}_i(\mathbf{x}_k, \mathbf{x}) = b_i\left(\frac{\mathbf{x}_k - \mathbf{x}}{\bar{h}_0}\right) = b_i(\boldsymbol{\xi}_k) \quad (5.60)$$

$$b_i(\mathbf{x}) \longrightarrow \tilde{b}_i(\mathbf{x}, \mathbf{x}) = b_i\left(\frac{\mathbf{x} - \mathbf{x}}{\bar{h}_0}\right) = b_i(\mathbf{0}) \quad (5.61)$$

\bar{h}_0 ist hierbei der durchschnittliche Grundwert der Einflussbereiche (siehe Abschnitt 5.5.1) der beteiligten Stützpunkte. Damit ergibt sich für den Wertebereich der skalierten lokalen Koordinaten näherungsweise:

$$-\alpha \lesssim \boldsymbol{\xi}_k = \frac{\mathbf{x}_k - \mathbf{x}}{\bar{h}_0} \lesssim \alpha \quad (5.62)$$

α ist der Wichtungsfunktionsüberlappungsparameter nach Gleichung (5.50).

5.6 Vergleich der in der FPM und der FEM genutzten Ansätze

Zur räumlichen Diskretisierung wird für die Finite-Elemente-Methode die in Abschnitt 5.2 beschriebene Interpolation mit Elementansatzfunktionen und für die Finite-Punkte-Methode die in Abschnitt 5.5 beschriebene MLS-Approximation verwendet.

	FPM	FEM
Ansatz	$\bar{f}(\mathbf{x}) = \sum_{k=1}^{N_K(\mathbf{x})} L_k(\mathbf{x}) f_k$ $= \mathbf{L}^T(\mathbf{x}) \mathbf{f}$ <p style="text-align: center;">$\langle N_K \rangle \quad \langle N_K \rangle$</p>	$\bar{f} = \sum_{k=1}^{N_K} L_k(\mathbf{x}) f_k$ $= \mathbf{L}^T(\mathbf{x}) \mathbf{f}$ <p style="text-align: center;">$\langle N_K \rangle \quad \langle N_K \rangle$</p>
Partiton of Unity	$\sum_{k=1}^{N_K(\mathbf{x})} L_k(\mathbf{x}) = 1$	$\sum_{k=1}^{N_K} L_k(\mathbf{x}) = 1$
Ansatzfunktionswerte an Knoten	$L_k(\mathbf{x}_i) \neq \delta_{ik}$ <p>(im Allgem.)</p>	$L_k(\mathbf{x}_i) = \delta_{ik}$
Art der räumlichen Diskretisierung	$\bar{f}(\mathbf{x}_k) \neq f_k$ <p>Approximation</p>	$\bar{f}(\mathbf{x}_k) = f_k$ <p>Interpolation</p>

Tabelle 5.8: Vergleich der Ansätze zur räumlichen Diskretisierung in FPM und FEM

Die Anzahl der Ansatzfunktionen N_K ist bei der FPM im Gegensatz zur FEM abhängig vom Ort \mathbf{x} . Die Summe der Ansatzfunktionen an jedem Ort ist in beiden Fällen gleich 1. Da die MLS-Ansatzfunktionen an den Knoten im Allgemeinen nicht die Werte 0 oder 1 annehmen, werden bei der FPM die Stützwerte an den Knoten f_k nicht „durchgereicht“. Dies hat zur Folge, dass zur Befriedigung der wesentlichen Randbedingungen in der FPM besondere Maßnahmen getroffen werden müssen (siehe Abschnitt 4.5).

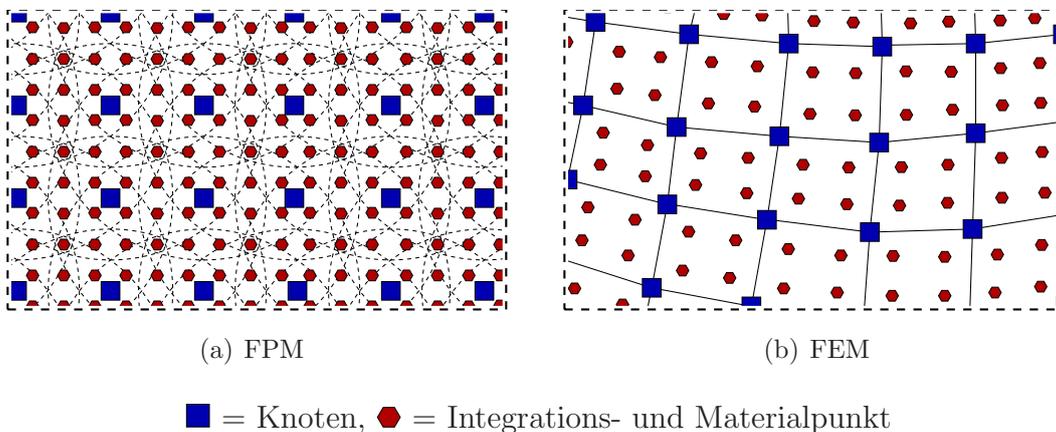


Abbildung 5.10: Netzausschnitte zur elementnetzfreien FPM und zur FEM

Die Interaktion zwischen Knotenfreiheitsgraden erfolgt über Einträge in der Systemsteifigkeitsmatrix bzw. der tangentialen Systemsteifigkeitsmatrix. Im Folgenden wird beschrieben, welche Knoten in der FPM und der FEM über solche Einträge gekoppelt sind.

In der FPM liefert ein Integrationspunkt Beiträge zu den Knotengrößen derjenigen Knoten, deren Wichtungsfunktionen den Integrationspunkt überdecken. Damit hat ein Knoten mit allen Knoten, mit denen er gemeinsame Integrationspunkte hat, gemeinsame Steifigkeitsmatrizeneinträge.

In der FEM liefern Integrationspunkte eines Elementes Beiträge zu den Knotengrößen aller Knoten des Elementes. Damit hat ein Knoten mit allen Knoten der angrenzenden Elemente gemeinsame Steifigkeitsmatrizeneinträge.

Kapitel 6

Diskretisierung, Linearisierung und Lösung

In diesem Kapitel werden die im vorangegangenen Kapitel zusammengestellten Grundgleichungen räumlich und zeitlich diskretisiert, linearisiert und gelöst.

6.1 Übergang zur Matrizenschreibweise

Es wird auf ein globales kartesisches Koordinatensystem Bezug genommen (①), die Voigtsche Notation nach Abschnitt 8.1 verwendet (②) und zur Matrizenschreibweise übergegangen (③). Diese wird dann durchgängig verwendet. Die Beziehungen werden für den dreidimensionalen Fall aufgestellt, da die Beziehungen für den zweidimensionalen Fall durch Streichen von Zeilen und Spalten einfach erhalten werden können.

Für den Spannungstensor ergibt sich

$$\boldsymbol{\sigma} \stackrel{\textcircled{1}}{=} \sum_{i,j=1}^3 \sigma_{ij} \mathbf{e}_i \otimes \mathbf{e}_j \stackrel{\textcircled{2}}{=} \sum_{i=1}^6 \overset{\text{v}}{\sigma}_i \overset{\text{v}}{\mathbf{e}}_i \stackrel{\textcircled{3}}{=} \overset{\text{v}}{\sigma} \overset{\text{v}}{\underset{\langle N_V \rangle}{\wedge}} = \left\{ \begin{array}{c} \overset{\text{v}}{\sigma}_1 \\ \overset{\text{v}}{\sigma}_2 \\ \overset{\text{v}}{\sigma}_3 \\ \overset{\text{v}}{\sigma}_4 \\ \overset{\text{v}}{\sigma}_5 \\ \overset{\text{v}}{\sigma}_6 \end{array} \right\} = \left\{ \begin{array}{c} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sqrt{2} \sigma_{12} \\ \sqrt{2} \sigma_{23} \\ \sqrt{2} \sigma_{13} \end{array} \right\} . \quad (6.1)$$

Um die Darstellung nicht zu überladen, wird im Folgenden mit Ausnahme der Komponenten der Voigtschen Vektoren auf die Kennzeichnung der Voigtschen Notation verzichtet.

Die Dehnungen werden entsprechend übersetzt. Bei der Umwandlung von Materialtetraden oder tangentialen Materialtetraden ist (8.5) zu beachten.

Für die Verschiebungs-Verzerrungs-Relation in Matrixschreibweise folgt durch Vergleich mit (4.3)

$$\underset{\langle N_V \rangle}{\boldsymbol{\varepsilon}} = \underset{\langle N_V \times N_D \rangle}{\mathbf{D}_{u\varepsilon}} \underset{\langle N_D \rangle}{\mathbf{u}} = \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{Bmatrix} = \begin{Bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \sqrt{2} \varepsilon_{12} \\ \sqrt{2} \varepsilon_{23} \\ \sqrt{2} \varepsilon_{13} \end{Bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial x_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial x_3} \\ \frac{1}{\sqrt{2}} \frac{\partial}{\partial x_2} & \frac{1}{\sqrt{2}} \frac{\partial}{\partial x_1} & 0 \\ 0 & \frac{1}{\sqrt{2}} \frac{\partial}{\partial x_3} & \frac{1}{\sqrt{2}} \frac{\partial}{\partial x_2} \\ \frac{1}{\sqrt{2}} \frac{\partial}{\partial x_3} & 0 & \frac{1}{\sqrt{2}} \frac{\partial}{\partial x_1} \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \\ u_3 \end{Bmatrix} . \quad (6.2)$$

Unter Ausnutzung von

$$\delta \boldsymbol{\varepsilon}^T = \delta \mathbf{u}^T \mathbf{D}_{u\varepsilon}^T \quad (6.3)$$

wird die schwache Form des Impulsatzes (4.35) in Matrixschreibweise überführt

$$\begin{aligned} g &= \underbrace{\int_{\mathcal{B}} [\delta \mathbf{u}^T \mathbf{D}_{u\varepsilon}^T] \boldsymbol{\sigma} dV}_{g_{\text{int}}} + \underbrace{\int_{\mathcal{B}} \delta \mathbf{u}^T \ddot{\mathbf{u}} \rho dV}_{g_{\text{m}}} \\ &\quad - \left[\underbrace{\int_{\mathcal{B}} \delta \mathbf{u}^T \mathbf{b} \rho dV}_{g_{\text{extV}}} + \underbrace{\int_{\delta \mathcal{B}} \delta \mathbf{u}^T \mathbf{t} dA}_{g_{\text{extA}}} \right] \\ &\quad - \underbrace{\int_{\delta \mathcal{B}_{\text{rnd}}} \delta \mathbf{u}^T \tilde{\mathbf{f}}_{\text{rnd}} dA}_{g_{\text{rnd}}} - \underbrace{\int_{\delta \mathcal{B}_{\text{kon}}} \delta \mathbf{u}^T \tilde{\mathbf{f}}_{\text{kon}} dA}_{g_{\text{kon}}} \\ &= g_{\text{int}} + g_{\text{m}} - g_{\text{extV}} - g_{\text{extA}} - g_{\text{rnd}} - g_{\text{kon}} = 0 . \end{aligned} \quad (6.4)$$

Für die im nächsten Abschnitt (6.2) folgende räumliche Diskretisierung werden die Ansätze für Verschiebungen, Testfunktionen, Dehnungen und virtuelle Dehnungen in Form von Matrixgleichungen bereitgestellt.

Ausgegangen wird von der in Abschnitt 5.1 näher beschriebenen allgemeinen Form

eines interpolierenden oder approximierenden Ansatzes.

$$\bar{f}(\mathbf{x}) = \sum_{k=1}^{N_K(\mathbf{x})} L_k(\mathbf{x}) f_k = \underset{\langle N_K \rangle}{\mathbf{L}^T(\mathbf{x})} \underset{\langle N_K \rangle}{\mathbf{f}} \quad (6.5)$$

$$\frac{\partial \bar{f}(\mathbf{x})}{\partial x_i} = \sum_{k=1}^{N_K(\mathbf{x})} \frac{\partial L_k(\mathbf{x})}{\partial x_i} f_k = \frac{\partial \underset{\langle N_K \rangle}{\mathbf{L}^T(\mathbf{x})}}{\partial x_i} \underset{\langle N_K \rangle}{\mathbf{f}} \quad (6.6)$$

Die Anzahl der Knoten (Stützpunkte) N_K hängt im allgemeinen Fall vom Ort \mathbf{x} der Interpolation oder Approximation ab. Die Interpolation mit Elementansatzfunktionen ist als Spezialfall enthalten. Auf die Kennzeichnung der diskretisierten Größe mit einem Querstrich wird im Folgenden verzichtet.

Die Ansätze für Verschiebungen \mathbf{u} , Geschwindigkeiten $\dot{\mathbf{u}}$, Beschleunigungen $\ddot{\mathbf{u}}$, Testfunktionen $\delta \mathbf{u}$, Dehnungen $\boldsymbol{\varepsilon}$, Dehnungsgeschwindigkeiten $\dot{\boldsymbol{\varepsilon}}$ und virtuelle Dehnungen $\delta \boldsymbol{\varepsilon}$ ähneln denen der FEM. Ein Unterschied besteht in der vom Ort abhängigen Dimension der \mathbf{L} - und \mathbf{B} -Matrizen, die die Werte der Ansatzfunktionen und der Ansatzfunktionsableitungen am Ort \mathbf{x}_p enthalten.

$$\underset{\langle N_D \rangle}{\mathbf{u}(\mathbf{x}_p)} = \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\mathbf{U}_p} \quad (6.7)$$

$$\underset{\langle N_D \rangle}{\dot{\mathbf{u}}_p(\mathbf{x}_p)} = \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\dot{\mathbf{U}}_p} \quad (6.8)$$

$$\underset{\langle N_D \rangle}{\ddot{\mathbf{u}}_p(\mathbf{x}_p)} = \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\ddot{\mathbf{U}}_p} \quad (6.9)$$

$$\underset{\langle N_D \rangle}{\delta \mathbf{u}(\mathbf{x}_p)} = \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\delta \mathbf{U}_p} = \underset{\langle N_K N_D \rangle}{\delta \mathbf{U}_p^T} \underset{\langle N_K N_D \times N_D \rangle}{\mathbf{L}^T(\mathbf{x}_p)} \quad (6.10)$$

$$\underset{\langle N_V \rangle}{\boldsymbol{\varepsilon}(\mathbf{x}_p)} = \underset{\langle N_V \times N_D \rangle}{\mathbf{D}_{u\varepsilon}} \underset{\langle N_D \rangle}{\mathbf{u}(\mathbf{x}_p)} = \underset{\langle N_V \times N_D \rangle}{\mathbf{D}_{u\varepsilon}} \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\mathbf{U}_p} = \underset{\langle N_V \times N_K N_D \rangle}{\mathbf{B}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\mathbf{U}_p} \quad (6.11)$$

$$\underset{\langle N_V \rangle}{\dot{\boldsymbol{\varepsilon}}(\mathbf{x}_p)} = \underset{\langle N_V \times N_D \rangle}{\mathbf{D}_{u\varepsilon}} \underset{\langle N_D \rangle}{\dot{\mathbf{u}}_p(\mathbf{x}_p)} = \underset{\langle N_V \times N_D \rangle}{\mathbf{D}_{u\varepsilon}} \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\dot{\mathbf{U}}_p} = \underset{\langle N_V \times N_K N_D \rangle}{\mathbf{B}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\dot{\mathbf{U}}_p} \quad (6.12)$$

$$\underset{\langle N_V \rangle}{\delta \boldsymbol{\varepsilon}(\mathbf{x}_p)} = \underset{\langle N_V \times N_D \rangle}{\mathbf{D}_{u\varepsilon}} \underset{\langle N_D \rangle}{\delta \mathbf{u}(\mathbf{x}_p)} = \underset{\langle N_V \times N_D \rangle}{\mathbf{D}_{u\varepsilon}} \underset{\langle N_D \times N_K N_D \rangle}{\mathbf{L}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\delta \mathbf{U}_p} = \underset{\langle N_V \times N_K N_D \rangle}{\mathbf{B}(\mathbf{x}_p)} \underset{\langle N_K N_D \rangle}{\delta \mathbf{U}_p} = \underset{\langle N_K N_D \rangle}{\delta \mathbf{U}_p^T} \underset{\langle N_K N_D \times N_V \rangle}{\mathbf{B}(\mathbf{x}_p)^T} \quad (6.13)$$

N_K ist hier nicht die globale Knotenanzahl, sondern die Anzahl der Knoten, deren Einflussbereiche den Ort \mathbf{x}_p umfassen. Die Vektoren \mathbf{U}_p , $\dot{\mathbf{U}}_p$, $\ddot{\mathbf{U}}_p$ und $\delta \mathbf{U}_p$ enthalten die Verschiebungs-, Geschwindigkeits-, Beschleunigungs- und Testfunktionswerte an diesen Knoten. N_D ist die räumliche Dimension und N_V die Dimension der Voigtschen Vektoren.

Für spätere Linearisierungen werden noch die Ableitungen

$$\frac{\partial \mathbf{u}(\mathbf{x}_p)}{\partial \mathbf{U}_p} = \mathbf{L}(\mathbf{x}_p) \quad (6.14)$$

$$\frac{\partial \boldsymbol{\varepsilon}(\mathbf{x}_p)}{\partial \mathbf{U}_p} = \mathbf{B}(\mathbf{x}_p) \quad (6.15)$$

$$\frac{\partial \dot{\boldsymbol{\varepsilon}}(\mathbf{x}_p)}{\partial \dot{\mathbf{U}}_p} = \mathbf{B}(\mathbf{x}_p) \quad (6.16)$$

benötigt, die sich unmittelbar aus (6.7), (6.11) und (6.12) ergeben.

6.2 Räumliche Diskretisierung der Bilanzgleichungen

Zur räumlichen Diskretisierung der schwachen Form des Impulssatzes (6.4) wird zunächst das Integrationsgebiet \mathcal{B} in Teilgebiete zerlegt. Bei einer Diskretisierung mit finiten Elementen werden die Teilgebiete \mathcal{B}_e (mit $e = 1, \dots, N_E$) durch die Elemente gebildet. Bei einer elementfreien Diskretisierung sind die Teilgebiete \mathcal{B}_p (mit $p = 1, \dots, N_P$) Integrationspunkten zugeordnete Bereiche. Die Integration über die Teilgebiete erfolgt in beiden Fällen numerisch.

Anhand eines der Volumenintegrale von (6.4) wird gezeigt, wie die Gebietszerlegung und numerische Integration für Methoden mit Elementnetzen und elementfreie Methoden vereinheitlicht werden kann. g_\diamond steht im Folgenden für g_{int} , g_m oder $g_{\text{ext}V}$.

Für eine Methode mit Elementen ergibt sich

$$g_\diamond = \int_{\mathcal{B}} \tilde{g}_\diamond dV = \sum_{e=1}^{N_E} \int_{\mathcal{B}_e} \tilde{g}_\diamond dV = \sum_{e=1}^{N_E} \sum_{p=1}^{N_{P_e}} \tilde{g}_{\diamond p} V_p \quad (6.17)$$

und für eine elementfreie Methode

$$g_\diamond = \int_{\mathcal{B}} \tilde{g}_\diamond dV = \sum_{p=1}^{N_P} \int_{\mathcal{B}_p} \tilde{g}_\diamond dV = \sum_{p=1}^{N_P} \tilde{g}_{\diamond p} V_p \quad (6.18)$$

Volumenbezogene Größen sind mit $(\tilde{\cdot})$ gekennzeichnet. $\tilde{g}_{\diamond p}$ ist eine abkürzende Schreibweise für $\tilde{g}_\diamond(\mathbf{x}_p)$ und V_p ist das einem Integrationspunkt zugeordnete Volumen.

Eine einheitliche Vorgehensweise wird erzielt, wenn bei Methoden, die Elementnetze

nutzen, die verschachtelte Summation über die Elemente und die Integrationspunkte der Elemente ersetzt wird durch eine Summation über die Integrationspunkte aller Elemente:

$$g_{\diamond} = \sum_{e=1}^{N_E} \sum_{p=1}^{N_{P_e}} \tilde{g}_{\diamond p} V_p = \sum_{p=1}^{N_P} \tilde{g}_{\diamond p} V_p \quad (6.19)$$

Im nächsten Schritt werden Verschiebungen und Testfunktionen durch lokale Ansätze entsprechend der Gleichungen (6.7) und (6.10) räumlich diskretisiert.

$$\mathbf{u}_p = \mathbf{u}(\mathbf{x}_p) = \mathbf{L}(\mathbf{x}_p) \mathbf{U}_p \quad (6.20)$$

$$\delta \mathbf{u}_p = \delta \mathbf{u}(\mathbf{x}_p) = \mathbf{L}(\mathbf{x}_p) \delta \mathbf{U}_p \quad (6.21)$$

Die Vektoren \mathbf{U}_p und $\delta \mathbf{U}_p$ enthalten die Verschiebungs- und Testfunktionswerte an den Knoten, deren Einflussbereiche den Ort \mathbf{x}_p umfassen.

Da lokale Ansätze verwendet werden, muss die Summation der Beiträge der Integrationspunkte durch eine Assemblierungsprozedur ersetzt werden. Die lokalen Größen sind an den richtigen Stellen den globalen Größen hinzuzufügen. Die Assemblierung wird mit dem Zeichen „ \mathbb{A} “ symbolisiert.¹

Der vollständige Ablauf der räumlichen Diskretisierung wird wieder anhand des Teilintegrals g_{\diamond} verdeutlicht.

$$\begin{aligned} g_{\diamond} &= \int_{\mathcal{B}} \tilde{g}_{\diamond} dV \stackrel{\textcircled{1}}{=} \sum_{p=1}^{N_P} \tilde{g}_{\diamond p} V_p && \stackrel{\textcircled{2}}{=} \sum_{p=1}^{N_P} \delta \mathbf{U}_p^T \mathbf{L}_p^T \tilde{\mathbf{f}}_{\diamond p} V_p \\ &\stackrel{\textcircled{3}}{=} \delta \mathbf{U}^T \mathbb{A}_{p=1}^{N_P} \mathbf{L}_p^T \tilde{\mathbf{f}}_{\diamond p} V_p && \stackrel{\textcircled{4}}{=} \delta \mathbf{U}^T \mathbb{A}_{p=1}^{N_P} \mathbf{F}_{\diamond p} \\ &\stackrel{\textcircled{5}}{=} \delta \mathbf{U}^T \mathbf{F}_{\diamond} \quad . \end{aligned} \quad (6.22)$$

Im Einzelnen werden folgende Schritte durchgeführt: ① Zerlegung in Teilbereiche und numerische Integration. ② Räumliche Diskretisierung der Verschiebungen und Testfunktionen über lokale Ansätze und Abspalten der Testfunktionsvektoren. ③ „Ausklammern“ der Testfunktionsvektoren; die Summation muss hier durch eine

¹Um auf eine gesonderte Schreibweise für die Assemblierungsprozedur verzichten zu können, werden formal häufig Zuordnungsmatrizen eingeführt. Die lokalen Größen werden durch Multiplikation mit den dünn besetzten Zuordnungsmatrizen auf die Dimension der globalen Größen „aufgebläht“ und können dann einfach aufsummiert werden. Diese Vorgehensweise ist jedoch numerisch ineffizient und daher für die praktische Umsetzung ungeeignet. Stattdessen wird zu jeder lokalen Freiheitsgradnummer die dazugehörige globale Freiheitsgradnummer abgespeichert, so dass eine Assemblierungsroutine die Einordnung der Komponenten der lokalen Größen in die globalen Größen vornehmen kann.

Assemblierung ersetzt werden. ④ Bestimmung der lokalen Kraftgrößen an den Integrationspunkten. ⑤ Assemblierung der Beiträge der Integrationspunkte zu globalen Kraftgrößen.

Entsprechend werden alle Volumenintegrale von (6.4) behandelt. Bei den Oberflächenintegralen wird ähnlich vorgegangen. Hier werden die flächenbezogenen Kräfte an den Orten der Randintegrationspunkte mit den den Randintegrationspunkten zugeordneten Flächen multipliziert. Assembliert wird über die Anzahl der Randpunkte N_{RP} .

Für die räumlich diskretisierte schwache Form der Impulsbilanz ergibt sich damit

$$\begin{aligned}
 g &= g_{\text{int}} + g_{\text{m}} - g_{\text{ext}} - g_{\text{rnd}} - g_{\text{kon}} \\
 &= \delta \mathbf{U}^T \left[\mathbf{F}_{\text{int}}(\mathbf{U}, \dot{\mathbf{U}}) + \mathbf{F}_{\text{m}}(\ddot{\mathbf{U}}) - \mathbf{F}_{\text{extV}} \right. \\
 &\quad \left. - \mathbf{F}_{\text{extA}} - \mathbf{F}_{\text{rnd}}(\mathbf{U}) - \mathbf{F}_{\text{kon}}(\mathbf{U}) \right] \\
 &= \delta \mathbf{U}^T \mathbf{G}(\mathbf{U}, \dot{\mathbf{U}}, \ddot{\mathbf{U}}) = 0 \quad .
 \end{aligned} \tag{6.23}$$

Da g für beliebige Testfunktionsvektoren $\delta \mathbf{U}$ verschwinden muss, kann gefolgert werden, dass die Kraftsumme \mathbf{G} gleich dem Nullvektor sein muss, womit abschließend als räumlich diskretisiertes System

$$\begin{aligned}
 \mathbf{G}(\mathbf{U}, \dot{\mathbf{U}}, \ddot{\mathbf{U}}) &= \mathbf{F}_{\text{int}}(\mathbf{U}, \dot{\mathbf{U}}) + \mathbf{F}_{\text{m}}(\ddot{\mathbf{U}}) - \mathbf{F}_{\text{extV}} \\
 &\quad - \mathbf{F}_{\text{extA}} - \mathbf{F}_{\text{rnd}}(\mathbf{U}) - \mathbf{F}_{\text{kon}}(\mathbf{U}) = \mathbf{0}
 \end{aligned} \tag{6.24}$$

mit den aus Volumenintegrationen stammenden Kräften

$$\mathbf{F}_{\text{int}}(\mathbf{U}, \dot{\mathbf{U}}) = \sum_{p=1}^{N_p} \tilde{\mathbf{F}}_{\text{int}p}(\mathbf{U}_p, \dot{\mathbf{U}}_p) V_p = \sum_{p=1}^{N_p} \mathbf{B}_p^T \boldsymbol{\sigma}_p(\mathbf{U}_p, \dot{\mathbf{U}}_p) V_p \tag{6.25}$$

$$\mathbf{F}_{\text{m}}(\ddot{\mathbf{U}}) = \mathbf{M} \ddot{\mathbf{U}} = \sum_{p=1}^{N_p} \tilde{\mathbf{F}}_{\text{m}p}(\ddot{\mathbf{U}}_p) V_p = \sum_{p=1}^{N_p} \mathbf{L}_p^T \mathbf{L}_p \rho_p V_p \ddot{\mathbf{U}}_p = \sum_{p=1}^{N_p} \mathbf{M}_p \ddot{\mathbf{U}}_p \tag{6.26}$$

$$\mathbf{F}_{\text{extV}} = \sum_{p=1}^{N_p} \tilde{\mathbf{F}}_{\text{extV}p} V_p = \sum_{p=1}^{N_p} \mathbf{L}_p^T \tilde{\mathbf{f}}_{\text{extV}p} V_p \tag{6.27}$$

und den aus Oberflächenintegrationen stammenden Kräften

$$\mathbf{F}_{\text{ext}A} = \sum_{p=1}^{N_{\text{RP}}} \tilde{\mathbf{F}}_{\text{ext}V_p} A_p \quad (6.28)$$

$$\mathbf{F}_{\text{rnd}}(\mathbf{U}) = \sum_{p=1}^{N_{\text{RP}}} \tilde{\mathbf{F}}_{\text{rnd}p}(\mathbf{U}_p) A_p = \sum_{p=1}^{N_{\text{RP}}} \mathbf{L}_p^T \tilde{\mathbf{f}}_{\text{rnd}p}(\mathbf{U}_p) A_p \quad (6.29)$$

$$\mathbf{F}_{\text{kon}}(\mathbf{U}) = \sum_{p=1}^{N_{\text{RP}}} \tilde{\mathbf{F}}_{\text{kon}p}(\mathbf{U}_p) A_p = \sum_{p=1}^{N_{\text{RP}}} \mathbf{L}_p^T \tilde{\mathbf{f}}_{\text{kon}p}(\mathbf{U}_p) A_p \quad (6.30)$$

erhalten wird. Im Abschnitt 6.4.2 wird das semidiskrete System (6.24) zeitlich diskretisiert.

6.3 Räumliche Diskretisierung der Kontaktflächen

Zur glatten(mindestens C^1 -stetigen) Beschreibung von Kontaktflächen können unterschiedliche Wege begangen werden:

EL-ABBASI ET AL. (2001) glätten Oberflächen von Starrkörpern über C^2 -kontinuierliche und Oberflächen von deformierbaren Körpern über C^1 -kontinuierliche Splines. Die Splines stützen sich auf die Oberflächenknoten und eine zusätzliche Angabe von Normalen oder Tangenten ist nicht erforderlich. Ähnlich gehen WRIGGERS ET AL. (2001) vor. An Stelle von Splines nutzen sie Bézier-Kurven zur Interpolation der Master-Oberflächen.

BELYTSCHKO ET AL. (2002) verwenden eine glatte Abstandsfunktion. Die vorzeichenbehaftete Abstandsfunktion basiert auf einer MLS-Approximation über die Oberflächenknoten. PUSO ET AL. (2002) glätten Oberflächen mit *Gregory Patches*; STADLER ET AL. (2003) verwenden hierzu *NURBS*.

Im Rahmen dieser Arbeit werden die Kontaktflächen im zweidimensionalen Raum (d.h. Kontaktlinien) mit C^1 -stetigen Hermite-Linienelementen (d.h. Splines) diskretisiert. Im Folgenden werden die hierzu verwendete Hermite-Interpolation (siehe z.B. DHATT ET AL. (1984)), die Definition der aus Hermite-Elementen zusammengesetzten Kontaktlinien und die Kontaktgeometrie beschrieben.

6.3.1 Hermite-Interpolation

Ein C^1 -stetiges Hermite-Element (Abbildung 6.1) wird über die Lage seiner Knoten und die Tangentenrichtungen in den Knoten definiert. Wie bei Lagrangeschen Linienelementen wird eine normierte Elementkoordinate ξ eingeführt, die an den Knoten \mathcal{A} und \mathcal{B} die Werte $\xi_{\mathcal{A}} = -1$ und $\xi_{\mathcal{B}} = 1$ annimmt. Die globalen Koordinaten des Elementes ergeben sich in Abhängigkeit von der normierten Koordinate aus

$$\hat{\mathbf{x}}(\xi) = H_1(\xi) \mathbf{x}_{\mathcal{A}} + H_2(\xi) \mathbf{x}_{\mathcal{B}} + \beta_{\mathcal{A}} l_{\mathcal{A}} H_3(\xi) \mathbf{t}_{\mathcal{A}} + \beta_{\mathcal{B}} l_{\mathcal{B}} H_4(\xi) \mathbf{t}_{\mathcal{B}} \quad . \quad (6.31)$$

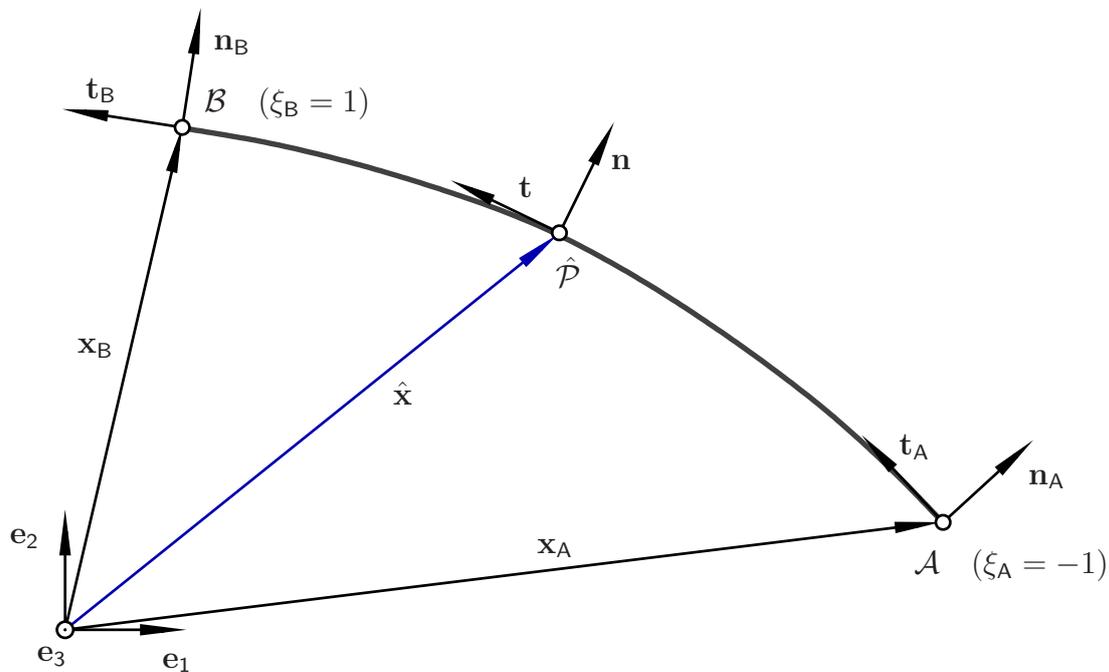


Abbildung 6.1: Hermite-Element

Die Hermite-Ansatzfunktionen

$$\begin{aligned} H_1(\xi) &= \frac{1}{4} [2 - 3\xi + \xi^3] & H_3(\xi) &= \frac{1}{4} [1 - \xi - \xi^2 + \xi^3] \\ H_2(\xi) &= \frac{1}{4} [2 + 3\xi - \xi^3] & H_4(\xi) &= \frac{1}{4} [-1 - \xi + \xi^2 + \xi^3] \end{aligned} \quad (6.32)$$

und ihre Ableitungen zeigt die Abbildung 6.2.

$\beta_{\mathcal{A}} l_{\mathcal{A}}$ und $\beta_{\mathcal{B}} l_{\mathcal{B}}$ sind Tangentenwichtungsfaktoren, wobei $\beta_{\mathcal{A}}$ und $\beta_{\mathcal{B}}$ positive dimen-

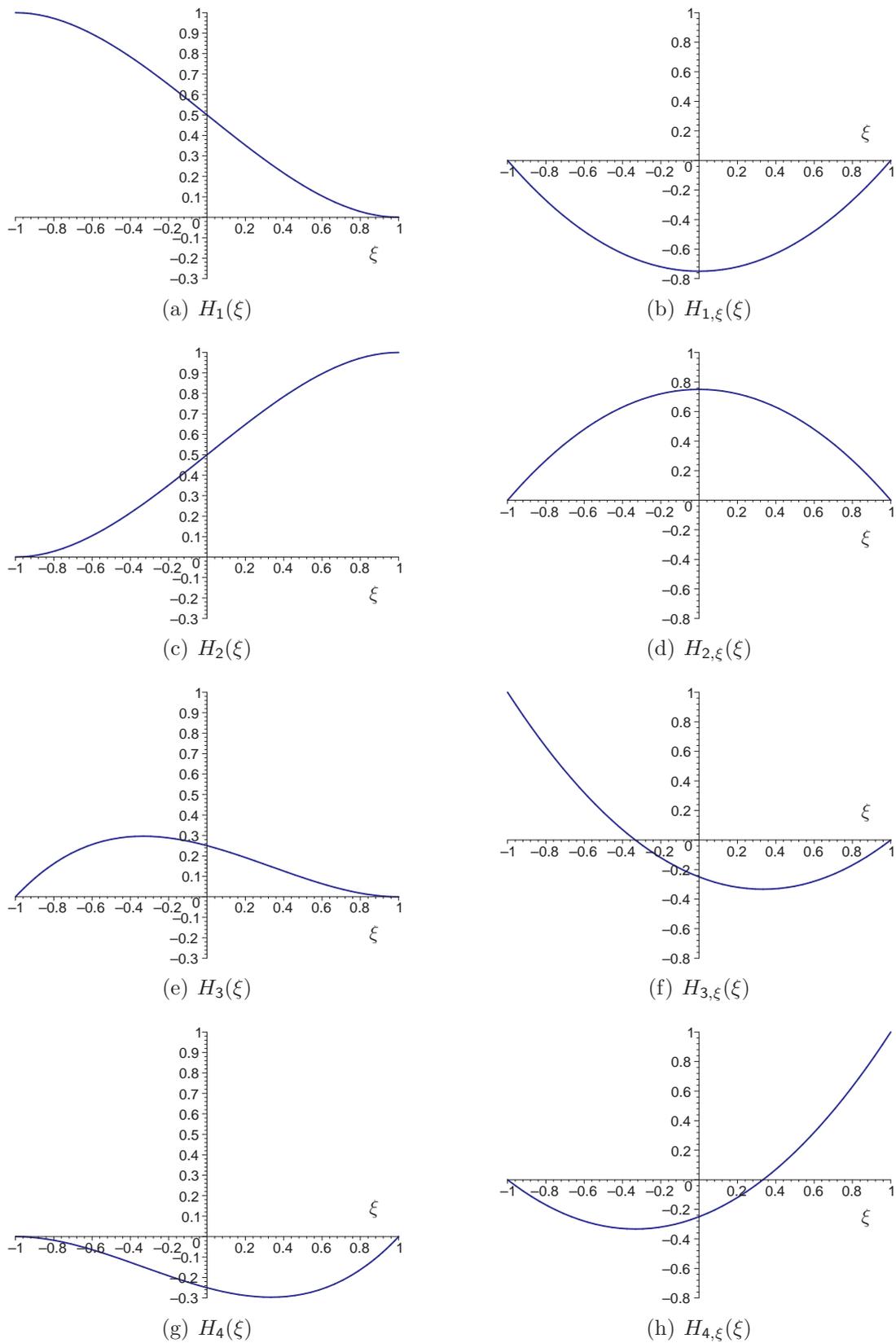


Abbildung 6.2: Hermite-Ansatzfunktionen und ihre Ableitungen

sionslose Zahlen und l_A und l_B Längen sind. Es wird speziell

$$\beta_A = \beta_B = \beta \quad (6.33)$$

und

$$l_A = l_B = l \quad (6.34)$$

mit

$$l = \frac{1}{2} \|\mathbf{x}_B - \mathbf{x}_A\| \quad (6.35)$$

(halber Knotenabstand) gewählt. Es sind auch andere Definitionen der Tangentenwichtungsfaktoren möglich. Die Wichtungsfaktoren sollten aber stets linear von einem Längenmaß abhängen, damit die Hermite-Interpolation invariant gegenüber Maßstabsänderungen ist.

Den Einfluss der Tangentenwichtung auf die Hermite-Interpolation bei unterschiedlichen Winkeln zwischen der Anfangstangente \mathbf{t}_A und der Endtangente \mathbf{t}_B zeigt die Abbildung 6.3. Wird $\beta = 0$ gewählt, entartet die Hermite-Interpolation zu einer linearen Interpolation. Eine sinnvolle Wahl ist $\beta = 1$. Die dazugehörige Interpolation ist fett dargestellt.

Durch Ableitung der globalen Elementkoordinaten nach der lokalen Elementkoordinate wird eine von der lokalen Elementkoordinate abhängige nichtnormierte Tangente $\tilde{\mathbf{t}}$ erhalten

$$\tilde{\mathbf{t}}(\xi) = \frac{\partial \hat{\mathbf{x}}(\xi)}{\partial \xi} = H_{1,\xi}(\xi) \mathbf{x}_A + H_{2,\xi}(\xi) \mathbf{x}_B + \beta l H_{3,\xi}(\xi) \mathbf{t}_A + \beta l H_{4,\xi}(\xi) \mathbf{t}_B \quad . \quad (6.36)$$

Die normierte Tangente ergibt sich aus

$$\mathbf{t}(\xi) = \frac{\tilde{\mathbf{t}}(\xi)}{\|\tilde{\mathbf{t}}(\xi)\|} \quad . \quad (6.37)$$

Für die Tangenten an den Knoten des Hermite-Elementes lässt sich

$$\mathbf{t}(-1) = \dots = \frac{\beta l \mathbf{t}_A}{\beta l} = \mathbf{t}_A \quad \text{und} \quad \mathbf{t}(1) = \dots = \frac{\beta l \mathbf{t}_B}{\beta l} = \mathbf{t}_B \quad (6.38)$$

leicht überprüfen.

Die Normale \mathbf{n} bildet mit der Tangente \mathbf{t} und dem senkrecht auf der Diskretisie-

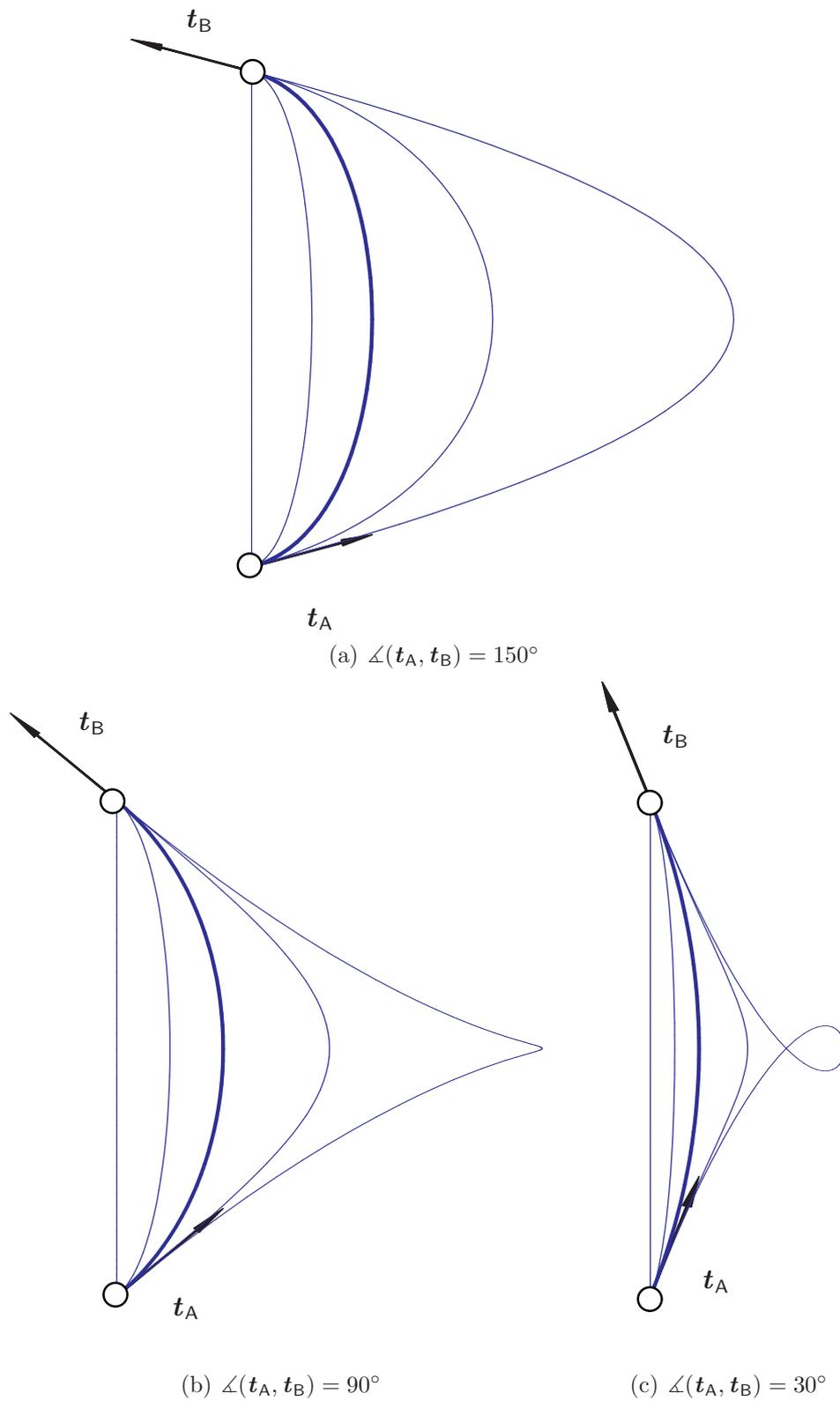


Abbildung 6.3: Hermite-Element, Einfluss des Tangentenwichtungsfaktors βl auf die Hermite-Interpolation bei verschiedenen Winkeln zwischen der Anfangs- und Endtangente, $\beta \in \{0, \frac{1}{2}, 1, 2, 8\}$

rungsebene stehenden Einheitsvektor \mathbf{e}_3 ein Rechtssystem.

$$\mathbf{n} = \mathbf{t} \times \mathbf{e}_3 = \begin{Bmatrix} t_2 \\ -t_1 \\ 0 \end{Bmatrix}, \quad \mathbf{t} = \mathbf{e}_3 \times \mathbf{n} = \begin{Bmatrix} -n_2 \\ n_1 \\ 0 \end{Bmatrix} \quad (6.39)$$

Für die nichtnormierten Vektoren $\tilde{\mathbf{n}}$ und $\tilde{\mathbf{t}}$ gilt entsprechend

$$\tilde{\mathbf{n}} = \tilde{\mathbf{t}} \times \mathbf{e}_3 = \begin{Bmatrix} \tilde{t}_2 \\ -\tilde{t}_1 \\ 0 \end{Bmatrix}, \quad \tilde{\mathbf{t}} = \mathbf{e}_3 \times \tilde{\mathbf{n}} = \begin{Bmatrix} -\tilde{n}_2 \\ \tilde{n}_1 \\ 0 \end{Bmatrix}. \quad (6.40)$$

Damit sind die für die Berechnung der Kontaktgeometrie in Abschnitt 6.3.3 benötigten Beziehungen zusammengestellt.

6.3.2 Kontaktlinien

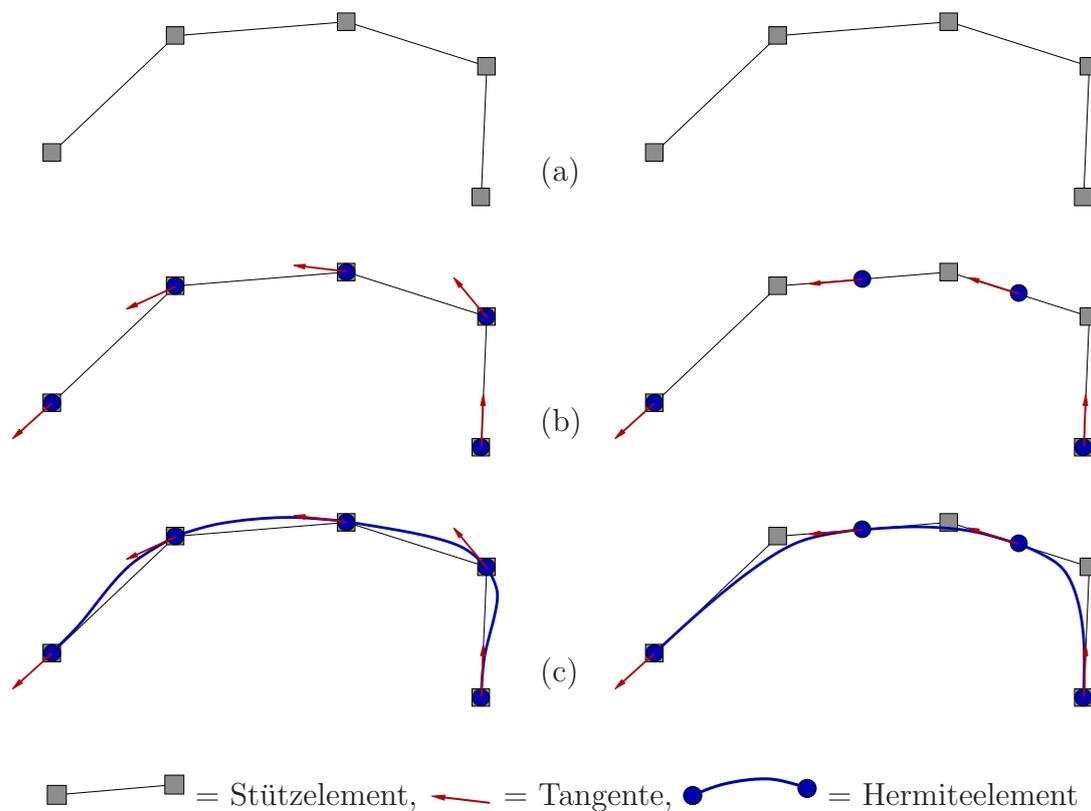


Abbildung 6.4: Kontaktliniendiskretisierung, links Variante 1, rechts Variante 2

Die direkte Definition von Hermite-Elementen ist umständlich, da neben den Knotenpositionen auch die Tangenten in den Knoten angegeben werden müssen. Statt-

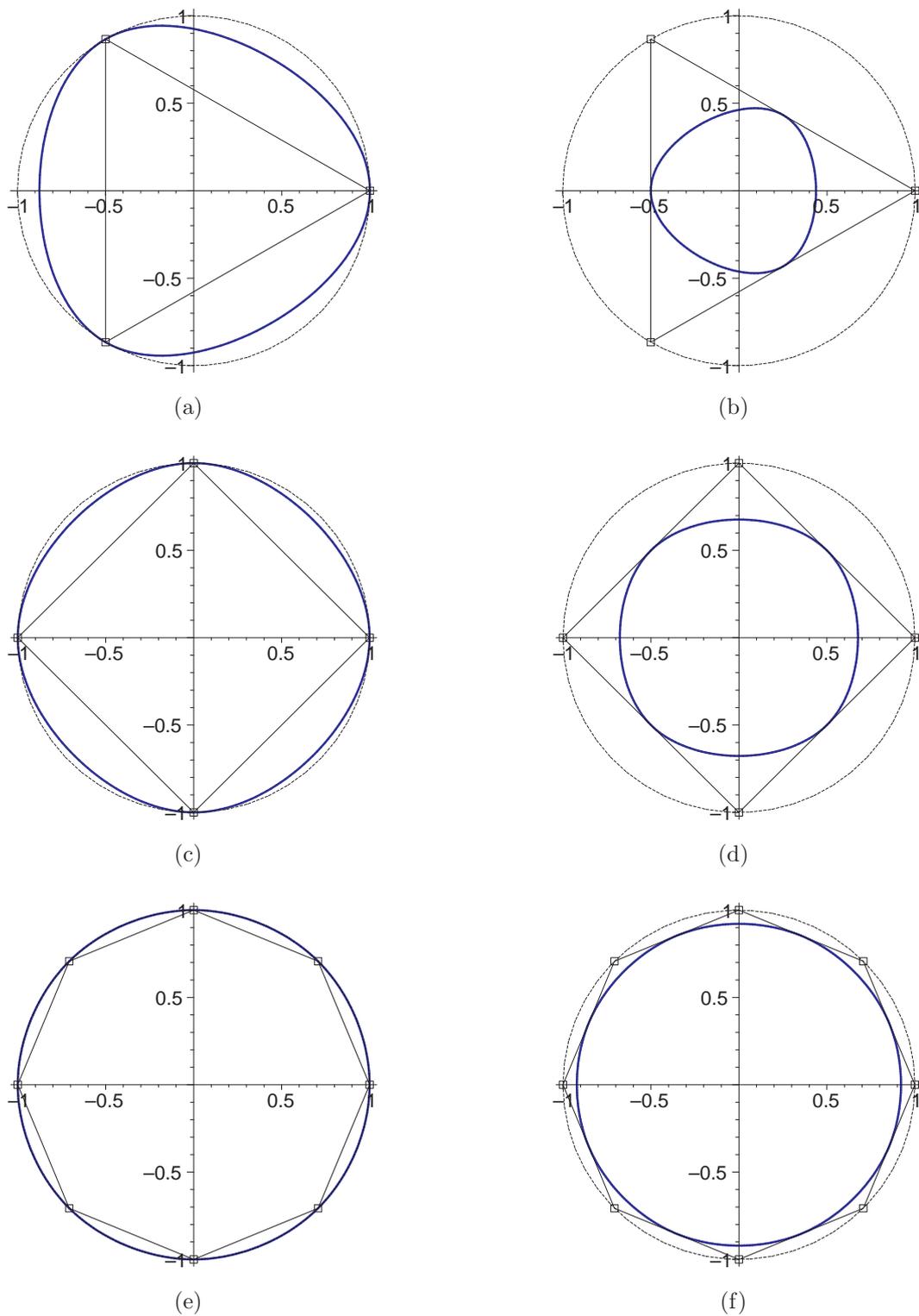


Abbildung 6.5: Hermite-Interpolation eines Kreises mit dem Radius 1, Knoten der Hermite-Elemente auf Knoten der Stützelemente (a), (c), (e) oder auf den Mitten der Stützelemente (b), (d), (f)

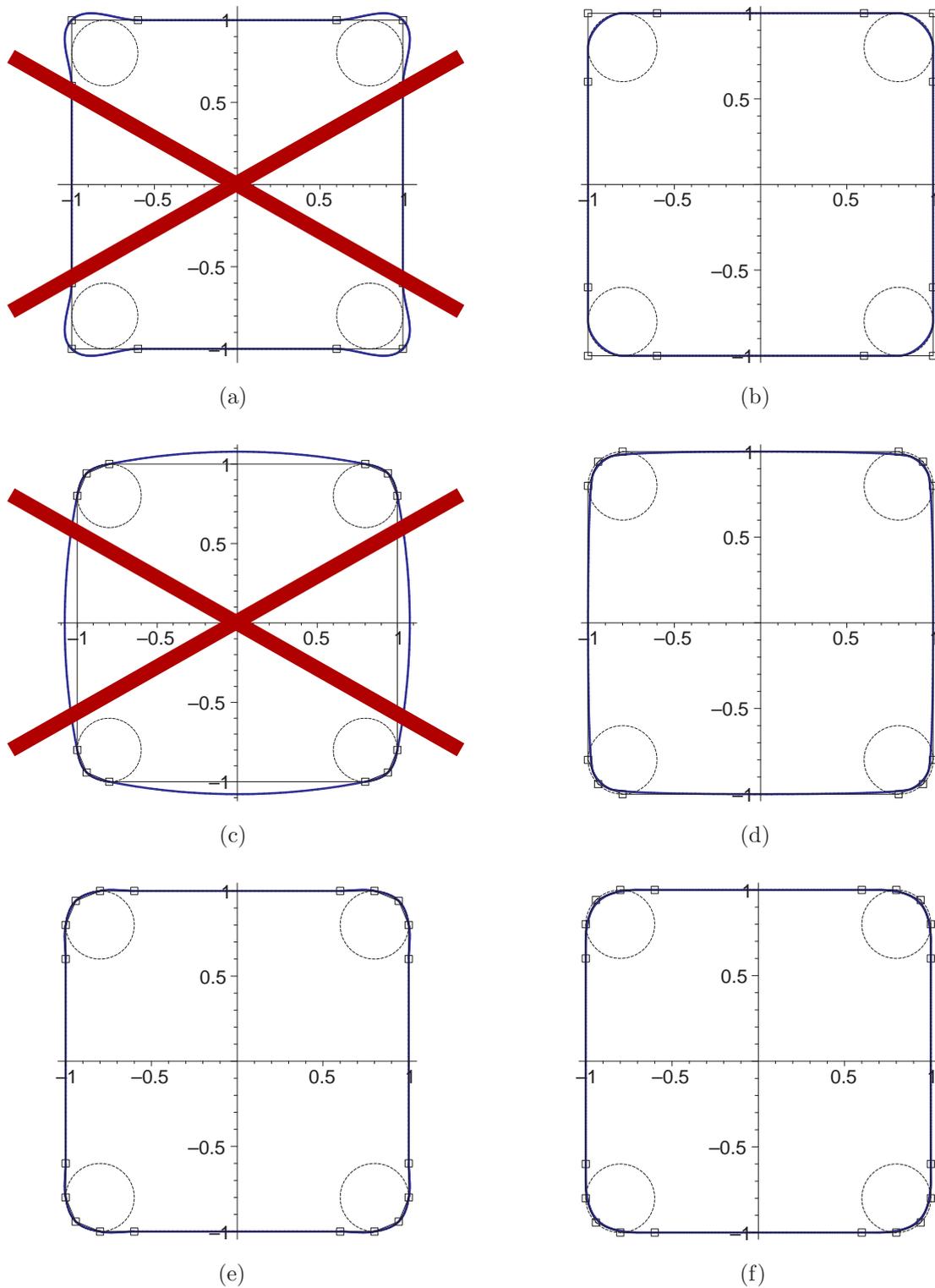


Abbildung 6.6: Hermite-Interpolation eines Quadrates mit abgerundeten Ecken, Knoten der Hermite-Elemente auf Knoten der Stützelemente (a), (c), (e) oder auf den Mitten der Stützelemente (b), (d), (f)

dessen werden aus Hermite-Elementen zusammengesetzte Linien über aus linearen Stützelementen zusammengesetzte Linien definiert, wie in WRIGGERS ET AL. (2001) vorgeschlagen wurde.

Hierzu ist für jede Kontaktlinie nur eine Serie aus Punkten anzugeben. Die Punkte werden linear zu einem polygonalen Linienzug verbunden. Für die Definition der Hermite-Elemente werden zwei Varianten untersucht:

In der ersten Variante werden die Knoten der Hermite-Elemente auf die Knoten der Stützelemente gelegt. Die Tangente an einem Knoten ergibt sich aus den gemittelten Richtungen der beiden an den Knoten angrenzenden Stützelemente.

In der zweiten Variante werden die Knoten der Hermite-Elemente auf die Mitten der Stützelemente gelegt. Die Tangenten ergeben sich unmittelbar aus den Richtungen der Stützelemente.

Die Abbildungen 6.5 und 6.6 zeigen die Hermite-Interpolation eines Kreises in unterschiedlich feiner Diskretisierung und die Hermite-Interpolation eines Quadrates mit abgerundeten Ecken. Die Interpolation nach der ersten Variante ist jeweils auf der linken Seite und die Interpolation nach der zweiten Variante auf der rechten Seite abgebildet. Die Stützelemente und ihre Knoten sind dünn wiedergegeben, die Hermite-Interpolation dick.

Die erste Variante ist bei der Interpolation des Kreises überlegen. Bereits mit 4 Punkten wird der Kreis gut wiedergegeben (siehe Abbildung 6.5c). Zur Abbildung abgerundeter Ecken ist sie allerdings schlecht geeignet. Die Abbildungen 6.6a und 6.6c zeigen offensichtlich unbrauchbare Interpolationen.

Dahingegen lassen sich mit Hilfe der zweiten Variante Ecken sehr gut interpolieren. Zur Definition einer abgerundeten Ecke reichen zwei Stützelemente aus (siehe Abbildung 6.6b). Je kürzer die Stützelemente an der Ecke gewählt werden, desto „schärfer“ wird die Ecke ausgebildet.

Die zweite Variante hat den Vorteil, dass bei einer sinnvollen Wahl der Tangentewichtungsfaktoren aus einem konvexen Polygonzug aus Stützelementen stets eine konvexe C^1 -stetige Linie aus Hermite-Elementen gebildet wird.

6.3.3 Kontaktgeometrie

Abbildung 6.7 zeigt die Geometrie einer inaktiven und einer aktiven Kontaktbedingung. \mathcal{P} ist ein Randpunkt eines deformierbaren Körpers, der in der Abbildung nicht dargestellt ist. Die Kontaktfläche wird durch ein Hermite-Element gebildet.

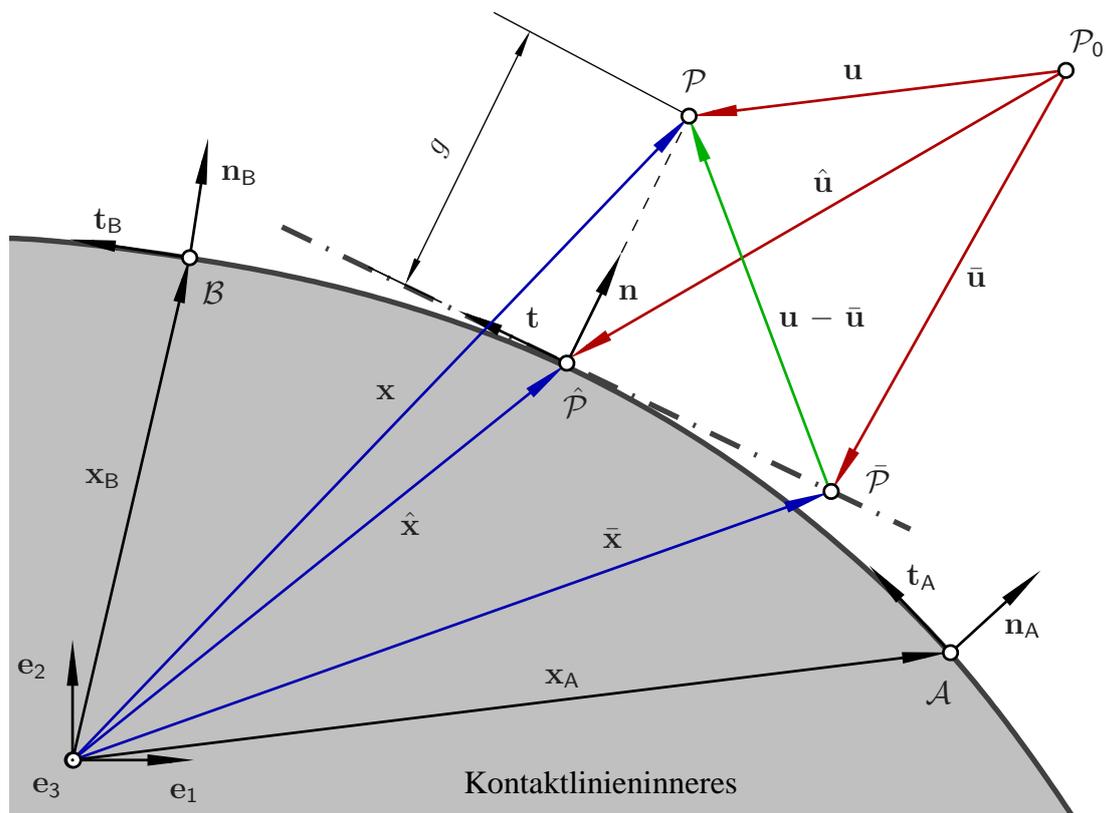
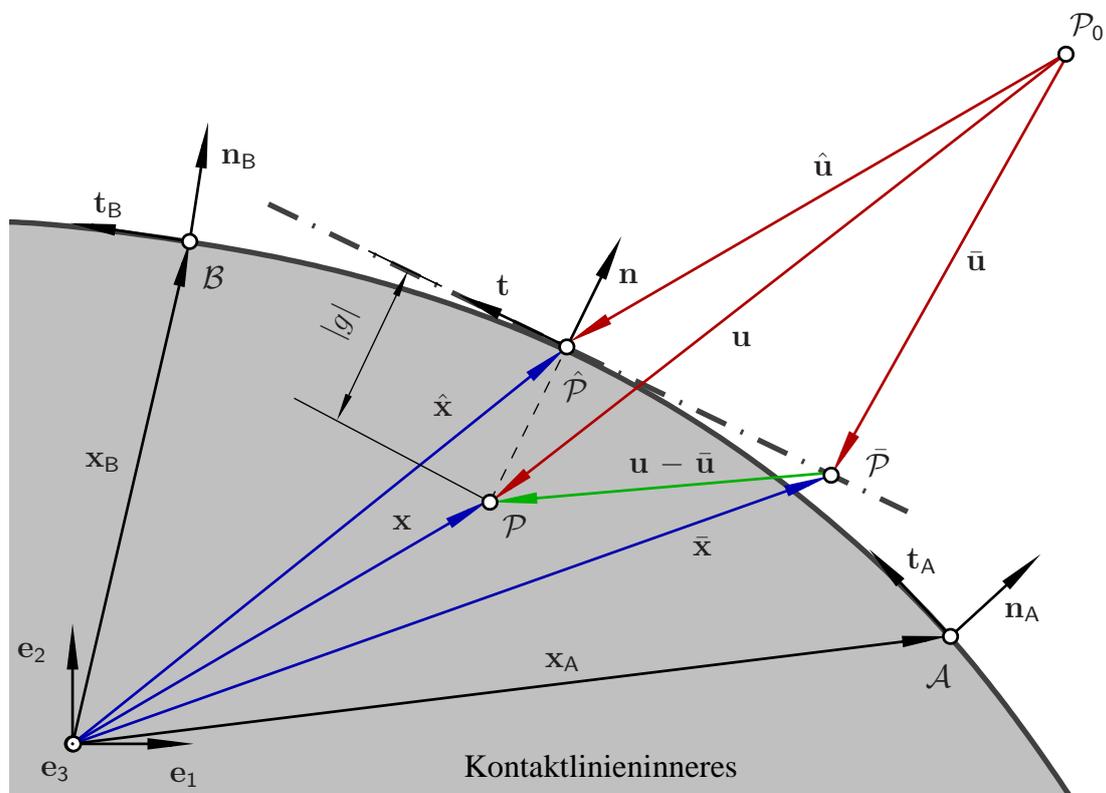
(a) Punkt P außerhalb der Kontaktlinie, $g > 0$, Kontaktbedingung inaktiv(b) Punkt P innerhalb der Kontaktlinie, $g < 0$, Kontaktbedingung aktiv

Abbildung 6.7: Hermite-Element, Geometrie einer Kontaktbedingung

Wie in Abschnitt 4.5 beschrieben wurde, werden zur Berechnung der Kontaktkraft und der Tangente der Kontaktkraft der Abstand g zwischen dem Punkt \mathcal{P} und der Kontaktoberfläche und der Normalenvektor \mathbf{n} der Kontaktoberfläche im Fusspunkt des Lotes benötigt. Zwischen dem Ortsvektor \mathbf{x} des Punktes \mathcal{P} in der Momentan-konfiguration und dem Ortsvektor $\hat{\mathbf{x}}$ des Fußpunktes $\hat{\mathcal{P}}$ des Lotes besteht folgender Zusammenhang:

$$\mathbf{x} = \hat{\mathbf{x}}(\xi) + \mathbf{n}(\xi) g \quad (6.41)$$

(6.41) ist ein nichtlineares Gleichungssystem mit den Unbekannten ξ und g . Es wird iterativ mit dem Newton-Raphson-Verfahren (siehe Abschnitt 7.2.1) gelöst. Der Normalenvektor \mathbf{n} hängt über Gleichung (6.39) von der Tangente \mathbf{t} ab. Diese lässt sich aus (6.36) und (6.37) in Abhängigkeit von der normierten Koordinate ξ berechnen.

Die Linearisierung wird vereinfacht, wenn der Vektor vom Fußpunkt des Lotes $\hat{\mathcal{P}}$ zum Punkt \mathcal{P} durch einen nichtnormierten Stellungsvektor $\tilde{\mathbf{n}}$ ausgedrückt wird:

$$\mathbf{n}(\xi) g = \tilde{\mathbf{n}}(\xi) \tilde{g} \quad (6.42)$$

Die Unbekannten ξ und \tilde{g} werden in einem Vektor zusammengefaßt

$$\mathbf{z} = \begin{Bmatrix} \xi \\ \tilde{g} \end{Bmatrix} \quad (6.43)$$

und die Gleichung (6.41) unter Berücksichtigung von (6.42) in Residualform gebracht

$$\mathbf{r}(\mathbf{z}) = \mathbf{x} - \hat{\mathbf{x}}(\xi) - \tilde{\mathbf{n}}(\xi) \tilde{g} = \mathbf{0} \quad (6.44)$$

Unter Verwendung der Gleichungen (6.40) und (6.36) und der Abkürzungen

$$\begin{aligned}
a_{11} &= \frac{1}{4} [2x_{A1} + 2x_{B1} + \beta l t_{A1} - \beta l t_{B1}] \\
a_{12} &= \frac{1}{4} [2x_{A2} + 2x_{B2} + \beta l t_{A2} - \beta l t_{B2}] \\
a_{21} &= \frac{1}{4} [-3x_{A1} + 3x_{B1} - \beta l t_{A1} - \beta l t_{B1}] \\
a_{22} &= \frac{1}{4} [-3x_{A2} + 3x_{B2} - \beta l t_{A2} - \beta l t_{B2}] \\
a_{31} &= \frac{1}{4} [-\beta l t_{A1} + \beta l t_{B1}] \\
a_{32} &= \frac{1}{4} [-\beta l t_{A2} + \beta l t_{B2}] \\
a_{41} &= \frac{1}{4} [x_{A1} - x_{B1} + \beta l t_{A1} + \beta l t_{B1}] \\
a_{42} &= \frac{1}{4} [x_{A2} - x_{B2} + \beta l t_{A2} + \beta l t_{B2}]
\end{aligned} \tag{6.45}$$

lassen sich die Komponenten des Residualvektor $\mathbf{r}(\mathbf{z})$

$$\begin{aligned}
\{\mathbf{r}(\mathbf{z})\}_1 = r_1 &= a_{41} \xi^3 + a_{31} \xi^2 + a_{21} \xi + a_{11} + \tilde{g} [3a_{42} \xi^2 + 2a_{32} \xi + a_{22}] - x_1 \\
\{\mathbf{r}(\mathbf{z})\}_2 = r_2 &= a_{42} \xi^3 + a_{32} \xi^2 + a_{22} \xi + a_{12} - \tilde{g} [3a_{41} \xi^2 + 2a_{31} \xi + a_{21}] - x_2
\end{aligned} \tag{6.46}$$

und der Tangente $\partial \mathbf{r}(\mathbf{z}) / \partial \mathbf{z}$

$$\begin{aligned}
\left[\frac{\partial \mathbf{r}(\mathbf{z})}{\partial \mathbf{z}} \right]_{11} &= \frac{\partial r_1}{\partial z_1} = 3a_{41} \xi^2 + 2a_{31} \xi + a_{21} + \tilde{g} [6a_{42} \xi + 2a_{32}] \\
\left[\frac{\partial \mathbf{r}(\mathbf{z})}{\partial \mathbf{z}} \right]_{12} &= \frac{\partial r_1}{\partial z_2} = 3a_{42} \xi^2 + 2a_{32} \xi + a_{22} \\
\left[\frac{\partial \mathbf{r}(\mathbf{z})}{\partial \mathbf{z}} \right]_{21} &= \frac{\partial r_2}{\partial z_1} = 3a_{42} \xi^2 + 2a_{32} \xi + a_{22} + \tilde{g} [-6a_{41} \xi - 2a_{31}] \\
\left[\frac{\partial \mathbf{r}(\mathbf{z})}{\partial \mathbf{z}} \right]_{22} &= \frac{\partial r_2}{\partial z_2} = -3a_{41} \xi^2 - 2a_{31} \xi - a_{21}
\end{aligned} \tag{6.47}$$

übersichtlich angeben.

Als Startwert (Iterationsindex $k = 0$) des Unbekanntenvektors wird

$$\mathbf{z}^0 = \left\{ \begin{array}{c} \xi^0 \\ \tilde{g}^0 \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ 0 \end{array} \right\} \tag{6.48}$$

gewählt. Der Unbekanntenvektor wird gemäß

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \left[\frac{\partial \mathbf{r}(\mathbf{z})}{\partial \mathbf{z}} \bigg|_{\mathbf{z}^k} \right]^{-1} \mathbf{r}(\mathbf{z}^k) \quad (6.49)$$

in jedem Iterationsschritt verbessert. Die Iteration wird abgebrochen, wenn der Unbekanntenvektor mit hinreichender Genauigkeit bestimmt wurde.

$$\|\mathbf{r}(\mathbf{z}^{k+1})\| < \varepsilon \quad (6.50)$$

Ergibt sich während der Iteration

$$\tilde{g} > 0 \quad \rightarrow \quad g > 0 \quad (6.51)$$

befindet sich der Punkt \mathcal{P} außerhalb der Kontaktoberfläche; ergibt sich

$$\xi < -1 \quad \text{oder} \quad 1 < \xi \quad (6.52)$$

befindet sich der Fußpunkt des Lotes außerhalb des Hermite-Elementes. In beiden Fällen wird die Iteration vorzeitig abgebrochen, da der Punkt \mathcal{P} mit dem Hermite-Element keinen Kontakt hat.

Nach Abschluss der Iteration (es gilt $-1 \leq \xi \leq 1$ und $\tilde{g} < 0$) werden der Abstand g und der Normalenvektor \mathbf{n} aus

$$g = \text{signum}(\tilde{g}) \|\tilde{\mathbf{n}} \tilde{g}\| \quad (6.53)$$

$$\mathbf{n} = \tilde{\mathbf{n}} \frac{\tilde{g}}{g} \quad (6.54)$$

gewonnen.

6.4 Zeitliche Diskretisierung

Im Rahmen dieser Arbeit wird in der zeitlichen Dimension eine andere Diskretisierung vorgenommen als in den räumlichen Dimensionen. Es werden konventionelle Zeitschrittverfahren (siehe Abschnitt 7.3) eingesetzt.

Eine zeitliche Diskretisierung muss auf Materialebene für die Entwicklungsgleichungen der inneren Unbekannten (4.19) und auf globaler Ebene für das semidiskrete System (6.24) durchgeführt werden. Für die Entwicklungsgleichungen wird ein Verfahren erster Ordnung und für das globale System ein Verfahren zweiter Ordnung

benötigt. In beiden Fällen kommen implizite Zeitschrittverfahren zum Einsatz, die gegenüber expliziten Verfahren größere Zeitschritte erlauben.

6.4.1 Zeitliche Diskretisierung der Entwicklungsgleichungen

Für die Entwicklungsgleichungen der inneren Unbekannten wurde die allgemeine Form

$$\dot{\mathbf{z}} = \mathbf{f}(\boldsymbol{\varepsilon}, \dot{\boldsymbol{\varepsilon}}, \mathbf{z}) \quad (6.55)$$

angenommen (siehe Abschnitt 4.3.4).

Zur zeitlichen Diskretisierung wird das Euler-Rückwärts-Verfahren (siehe Abschnitt 7.3.1) verwendet. Die zeitlichen Ableitungen des Vektors der inneren Variablen und der Dehnungen werden durch Differenzenquotienten ersetzt und die übrigen Größen zum Zeitpunkt t_{n+1} ausgewertet.

$$\begin{aligned} \dot{\mathbf{z}} &\longrightarrow \frac{\mathbf{z}_{n+1} - \mathbf{z}_n}{\Delta t} \\ \dot{\boldsymbol{\varepsilon}} &\longrightarrow \frac{\boldsymbol{\varepsilon}_{n+1} - \boldsymbol{\varepsilon}_n}{\Delta t} \\ \mathbf{z} &\longrightarrow \mathbf{z}_{n+1} \\ \boldsymbol{\varepsilon} &\longrightarrow \boldsymbol{\varepsilon}_{n+1} \end{aligned} \quad (6.56)$$

Damit wird das nichtlineare Gleichungssystem

$$\frac{\mathbf{z}_{n+1} - \mathbf{z}_n}{\Delta t} = \hat{\mathbf{f}}(\boldsymbol{\varepsilon}_{n+1}, \mathbf{z}_{n+1}) \quad (6.57)$$

für die inneren Unbekannten zum Zeitpunkt t_{n+1} erhalten. Dieses wird in Residualform gebracht:

$$\mathbf{r}(\boldsymbol{\varepsilon}_{n+1}, \mathbf{z}_{n+1}) = \mathbf{z}_{n+1} - \mathbf{z}_n - \Delta t \hat{\mathbf{f}}(\boldsymbol{\varepsilon}_{n+1}, \mathbf{z}_{n+1}) = \mathbf{0} \quad (6.58)$$

Um die Darstellung übersichtlicher zu gestalten, wird im Folgenden auf die Notation des Indexes $(\cdot)_{n+1}$ für das Zeitschrittende verzichtet.

$$\mathbf{r}(\boldsymbol{\varepsilon}, \mathbf{z}) = \mathbf{z} - \mathbf{z}_n - \Delta t \hat{\mathbf{f}}(\boldsymbol{\varepsilon}, \mathbf{z}) = \mathbf{0} \quad (6.59)$$

Die vorstehende Beziehung wird in Abschnitt 6.5.1 linearisiert.

6.4.2 Zeitliche Diskretisierung der Bilanzgleichungen

Es wird das Newmark-Verfahrens (siehe Abschnitt 7.3.2) eingesetzt. Als primäre Unbekannte werden die Verschiebungen gewählt. Die entsprechend der Gleichungen (7.48 und 7.49) für die Beschleunigungen und Geschwindigkeiten aufgestellten Ansätze

$$\ddot{\mathbf{U}}_{n+1} = \alpha_1 [\mathbf{U}_{n+1} - \mathbf{U}_n] - \alpha_2 \dot{\mathbf{U}}_n - \alpha_3 \ddot{\mathbf{U}}_n \quad (6.60)$$

$$\dot{\mathbf{U}}_{n+1} = \alpha_4 [\mathbf{U}_{n+1} - \mathbf{U}_n] + \alpha_5 \dot{\mathbf{U}}_n + \alpha_6 \ddot{\mathbf{U}}_n \quad (6.61)$$

werden in das semidiskretisierte System (6.24) und die Beziehungen für die Teilkräfte (6.25 bis 6.30) eingesetzt. Es wird das nichtlineare Gleichungssystem

$$\begin{aligned} \mathbf{G}(\mathbf{U}_{n+1}) &= \mathbf{F}_{\text{int}}(\mathbf{U}_{n+1}) + \mathbf{F}_m(\mathbf{U}_{n+1}) - \mathbf{F}_{\text{extV}} \\ &\quad - \mathbf{F}_{\text{extA}} - \mathbf{F}_{\text{rnd}}(\mathbf{U}_{n+1}) - \mathbf{F}_{\text{kon}}(\mathbf{U}_{n+1}) = \mathbf{0} \end{aligned} \quad (6.62)$$

für die Verschiebungen am Zeitschrittende \mathbf{U}_{n+1} erhalten. Die aus den Volumenintegrationen stammenden Kräfte lauten

$$\mathbf{F}_{\text{int}}(\mathbf{U}_{n+1}) = \bigwedge_{p=1}^{N_p} \tilde{\mathbf{F}}_{\text{int}p}(\mathbf{U}_{n+1p}) V_p = \bigwedge_{p=1}^{N_p} \mathbf{B}_p^T \boldsymbol{\sigma}_p(\mathbf{U}_{n+1p}) V_p \quad (6.63)$$

$$\begin{aligned} \mathbf{F}_m(\mathbf{U}_{n+1}) &= \bigwedge_{p=1}^{N_p} \tilde{\mathbf{F}}_{m p}(\mathbf{U}_{n+1p}) V_p \\ &= \bigwedge_{p=1}^{N_p} \mathbf{L}_p^T \mathbf{L}_p \rho_p V_p \left[\alpha_1 [\mathbf{U}_{n+1p} - \mathbf{U}_{np}] - \alpha_2 \dot{\mathbf{U}}_{np} - \alpha_3 \ddot{\mathbf{U}}_{np} \right] \\ &= \bigwedge_{p=1}^{N_p} \mathbf{M}_p \left[\alpha_1 [\mathbf{U}_{n+1p} - \mathbf{U}_{np}] - \alpha_2 \dot{\mathbf{U}}_{np} - \alpha_3 \ddot{\mathbf{U}}_{np} \right] \\ &= \mathbf{M} \left[\alpha_1 [\mathbf{U}_{n+1} - \mathbf{U}_n] - \alpha_2 \dot{\mathbf{U}}_n - \alpha_3 \ddot{\mathbf{U}}_n \right] \end{aligned} \quad (6.64)$$

$$\mathbf{F}_{\text{extV}} = \bigwedge_{p=1}^{N_p} \tilde{\mathbf{F}}_{\text{extV}p} V_p = \bigwedge_{p=1}^{N_p} \mathbf{L}_p^T \tilde{\mathbf{f}}_{\text{extV}p} V_p \quad (6.65)$$

und die aus den Oberflächenintegrationen stammenden Kräfte lauten

$$\mathbf{F}_{\text{extA}} = \bigwedge_{p=1}^{N_{\text{RP}}} \tilde{\mathbf{F}}_{\text{extA } p} A_p \quad (6.66)$$

$$\begin{aligned} \mathbf{F}_{\text{rnd}}(\mathbf{U}_{n+1}) &= \bigwedge_{p=1}^{N_{\text{RP}}} \tilde{\mathbf{F}}_{\text{rnd } p}(\mathbf{U}_{n+1 p}) A_p = \bigwedge_{p=1}^{N_{\text{RP}}} \mathbf{L}_p^T \tilde{\mathbf{f}}_{\text{rnd } p}(\mathbf{U}_{n+1 p}) A_p \\ &= \bigwedge_{p=1}^{N_{\text{RP}}} \mathbf{L}_p^T \mathbf{f}_{\text{rnd } p}(\mathbf{U}_{n+1 p}) \end{aligned} \quad (6.67)$$

$$\begin{aligned} \mathbf{F}_{\text{kon}}(\mathbf{U}_{n+1}) &= \bigwedge_{p=1}^{N_{\text{RP}}} \tilde{\mathbf{F}}_{\text{kon } p}(\mathbf{U}_{n+1 p}) A_p = \bigwedge_{p=1}^{N_{\text{RP}}} \mathbf{L}_p^T \tilde{\mathbf{f}}_{\text{kon } p}(\mathbf{U}_{n+1 p}) A_p \\ &= \bigwedge_{p=1}^{N_{\text{RP}}} \mathbf{L}_p^T \mathbf{f}_{\text{kon } p}(\mathbf{U}_{n+1 p}) \quad . \end{aligned} \quad (6.68)$$

Die Linearisierung der diskretisierten Bilanzgleichungen (6.62) wird in Abschnitt 6.5.2 vorgenommen. Hierfür werden noch die Ableitungen

$$\frac{\partial \dot{\mathbf{U}}_{n+1}}{\partial \mathbf{U}_{n+1}} = \alpha_4 \quad (6.69)$$

$$\frac{\partial \ddot{\mathbf{U}}_{n+1}}{\partial \mathbf{U}_{n+1}} = \alpha_1 \quad (6.70)$$

benötigt, die sich unmittelbar aus (6.61) und (6.60) ergeben.

Um die Darstellung übersichtlicher zu gestalten, wird im Folgenden auf die Notation des Indexes $(\cdot)_{n+1}$ für das Zeitschrittende verzichtet.

6.5 Linearisierung und Lösung

Sowohl die Entwicklungsgleichungen der inneren Variablen (6.59) als auch das globale Gleichungssystem (6.62) sind im Allgemeinen nichtlinear. Die Lösung erfolgt daher in beiden Fällen iterativ.

6.5.1 Linearisierung und Lösung der Materialgleichungen

Die Materialgleichungen werden an den Orten der Materialpunkte ausgewertet. Auf eine Kennzeichnung mit $(\cdot)_p$ wird in diesem Abschnitt verzichtet, da sie für alle Größen erfolgen müsste.

Auf Materialebene sind die inneren Variablen aus den diskretisierten Entwicklungsgleichungen (6.59), die Spannung aus der konstitutiven Beziehung (4.18) und die Materialtangente durch Linearisierung der konstitutiven Beziehung zu berechnen.

Mit der Berechnung der inneren Variablen wird begonnen. Die zeitlich diskretisierten und in Residualform gebrachten Entwicklungsgleichungen (6.59) beschreiben einen im allgemeinen Fall nichtlinearen Zusammenhang zwischen den Dehnungen und den inneren Zustandsgrößen am Zeitschrittende und müssen daher iterativ gelöst werden.

$$\mathbf{r}(\boldsymbol{\varepsilon}, \mathbf{z}) = \mathbf{0} \quad (6.71)$$

Die Lösung erfolgt mit dem im Abschnitt 7.2.1 beschriebenen Newton-Raphson-Verfahren. Die Dehnung $\boldsymbol{\varepsilon}_{n+1}^j$ wird in jedem Bilanziterationsschritt neu berechnet und bleibt während der Materialiteration konstant. Die Indizes für den Zeitschritt $(\cdot)_{n+1}$ und die Bilanziteration $(\cdot)^j$ (äußere Iteration) werden fortgelassen. Im folgenden bezieht sich der obere Index $(\cdot)^k$ auf die Materialiteration (innere Iteration).

$$\begin{aligned} \mathbf{r}(\mathbf{z}^{k+1}) &= \mathbf{r}(\mathbf{z}^k + \Delta \mathbf{z}^{k+1}) \\ &\approx \mathbf{r}(\mathbf{z}^k) + \left. \frac{\partial \mathbf{r}(\mathbf{z})}{\partial \mathbf{z}} \right|_{\mathbf{z}^k} \Delta \mathbf{z}^{k+1} \\ &= \mathbf{r}(\mathbf{z}^k) + \mathbf{K}_{\text{lok}}(\mathbf{z}^k) \Delta \mathbf{z}^{k+1} \stackrel{!}{=} \mathbf{0} \end{aligned} \quad (6.72)$$

Für die Verbesserung des Vektors der inneren Variablen im Iterationsschritt und den Vektor der inneren Variablen am Ende des Iterationsschrittes ergeben sich:

$$\Delta \mathbf{z}^{k+1} = -\mathbf{K}_{\text{lok}}^{-1}(\mathbf{z}^k) \mathbf{r}(\mathbf{z}^k) \quad (6.73)$$

$$\mathbf{z}^{k+1} = \mathbf{z}^k + \Delta \mathbf{z}^{k+1} \quad (6.74)$$

Als Anfangswert für die Iteration wird der konvergierte Vektor der inneren Variablen des letzten Zeitschrittes verwendet:

$$\mathbf{z}^0 = \mathbf{z}_n \quad (6.75)$$

Die Iteration wird abgebrochen, wenn die Norm des Residuums bzw. die Norm der Änderung des Vektors der inneren Variablen festgelegte Schranken unterschreiten:

$$\left[\|\mathbf{r}(\mathbf{z}^{k+1})\| < \text{tol}_r \quad \text{und/oder} \quad \|\Delta \mathbf{z}^{k+1}\| < \text{tol}_{\Delta z} \right] \rightarrow \text{Abbruch} \quad (6.76)$$

Ist der Vektor der inneren Variablen berechnet, so folgt die Spannung aus der in

Abschnitt 4.3.4 in verallgemeinerter Form angegebenen konstitutiven Beziehung

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{\varepsilon}, \dot{\boldsymbol{\varepsilon}}, \mathbf{z}) \quad . \quad (6.77)$$

Im Zuge der Linearisierung des Vektors der inneren Kräfte muss das totale Differential der Spannungen gebildet werden. Wird eine konstitutive Beziehung entsprechend der obigen Form angenommen, so gilt für das totale Differential der Spannungen

$$\begin{aligned} \Delta \boldsymbol{\sigma} &= \frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}} \Delta \boldsymbol{\varepsilon} + \frac{\partial \boldsymbol{\sigma}}{\partial \dot{\boldsymbol{\varepsilon}}} \Delta \dot{\boldsymbol{\varepsilon}} + \frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{z}} \Delta \mathbf{z} \\ &= \left[\left[\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}} + \frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \boldsymbol{\varepsilon}} \right] \frac{\partial \boldsymbol{\varepsilon}}{\partial \mathbf{U}} + \frac{\partial \boldsymbol{\sigma}}{\partial \dot{\boldsymbol{\varepsilon}}} \frac{\partial \dot{\boldsymbol{\varepsilon}}}{\partial \dot{\mathbf{U}}} \frac{\partial \dot{\mathbf{U}}}{\partial \mathbf{U}} \right] \Delta \mathbf{U} \quad . \end{aligned} \quad (6.78)$$

Mit (6.15),(6.16) und (6.69) folgt

$$\Delta \boldsymbol{\sigma} = \left[\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}} + \frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \boldsymbol{\varepsilon}} + \alpha_4 \frac{\partial \boldsymbol{\sigma}}{\partial \dot{\boldsymbol{\varepsilon}}} \right] \mathbf{B} \Delta \mathbf{U} \quad . \quad (6.79)$$

Für die Ableitung der inneren Variablen nach den Dehnungen $\frac{\partial \mathbf{z}}{\partial \boldsymbol{\varepsilon}}$ wird noch eine Beziehung benötigt. Diese wird aus der Forderung gewonnen, dass das totale Differential der in Residualform gebrachten Entwicklungsgleichungen (6.59) verschwinden muss.

$$\begin{aligned} \Delta \mathbf{r}(\boldsymbol{\varepsilon}, \mathbf{z}) &= \frac{\partial \mathbf{r}}{\partial \boldsymbol{\varepsilon}} \Delta \boldsymbol{\varepsilon} + \frac{\partial \mathbf{r}}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \boldsymbol{\varepsilon}} \Delta \boldsymbol{\varepsilon} \\ &= \left[\frac{\partial \mathbf{r}}{\partial \boldsymbol{\varepsilon}} + \frac{\partial \mathbf{r}}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \boldsymbol{\varepsilon}} \right] \Delta \boldsymbol{\varepsilon} = \mathbf{0} \end{aligned} \quad (6.80)$$

Da die obige Beziehung auch für $\Delta \boldsymbol{\varepsilon} \neq \mathbf{0}$ erfüllt sein muss, wird gefordert, dass der Ausdruck innerhalb der Klammern verschwindet

$$\frac{\partial \mathbf{r}}{\partial \boldsymbol{\varepsilon}} + \frac{\partial \mathbf{r}}{\partial \mathbf{z}} \frac{\partial \mathbf{z}}{\partial \boldsymbol{\varepsilon}} = \mathbf{0} \quad , \quad (6.81)$$

woraus sich die gesuchte Beziehung

$$\frac{\partial \mathbf{z}}{\partial \boldsymbol{\varepsilon}} = - \left[\frac{\partial \mathbf{r}}{\partial \mathbf{z}} \right]^{-1} \frac{\partial \mathbf{r}}{\partial \boldsymbol{\varepsilon}} = - \mathbf{K}_{\text{lok}}^{-1} \frac{\partial \mathbf{r}}{\partial \boldsymbol{\varepsilon}} \quad (6.82)$$

ergibt. Die lokale Steifigkeit \mathbf{K}_{lok} ist in Abhängigkeit von den konvergierten inneren Variablen zu bestimmen.

Schließlich folgt durch Einsetzen von (6.82) in (6.79) für das totale Differential der

Spannungen

$$\begin{aligned} \Delta \boldsymbol{\sigma} &= \frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{U}} \Delta \mathbf{U} = \left[\frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}} - \frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{z}} \mathbf{K}_{\text{lok}}^{-1} \frac{\partial \mathbf{r}}{\partial \boldsymbol{\varepsilon}} + \alpha_4 \frac{\partial \boldsymbol{\sigma}}{\partial \dot{\boldsymbol{\varepsilon}}} \right] \mathbf{B} \Delta \mathbf{U} \\ &= [\mathbf{C}_T + \alpha_4 \mathbf{D}_T] \mathbf{B} \Delta \mathbf{U} \quad . \end{aligned} \quad (6.83)$$

α_4 ist ein Parameter des Newmark-Verfahrens und \mathbf{C}_T und \mathbf{D}_T sind die Materialtangente.

$$\mathbf{C}_T = \frac{\partial \boldsymbol{\sigma}}{\partial \boldsymbol{\varepsilon}} - \frac{\partial \boldsymbol{\sigma}}{\partial \mathbf{z}} \mathbf{K}_{\text{lok}}^{-1} \frac{\partial \mathbf{r}}{\partial \boldsymbol{\varepsilon}} \quad (6.84)$$

$$\mathbf{D}_T = \frac{\partial \boldsymbol{\sigma}}{\partial \dot{\boldsymbol{\varepsilon}}} \quad (6.85)$$

6.5.2 Linearisierung und Lösung der Bilanzgleichungen

Die räumlich und zeitlich diskretisierten und in Residualform gebrachten Bilanzgleichungen (6.62) werden iterativ mit dem Newton-Raphson-Verfahren (siehe Abschnitt 7.2.1) gelöst. Aus der nach dem ersten Glied abgebrochenen Taylorreihenentwicklung für das globale Residuum

$$\begin{aligned} \mathbf{G}(\mathbf{U}^{j+1}) &= \mathbf{G}(\mathbf{U}^j + \Delta \mathbf{U}^{j+1}) \\ &\approx \mathbf{G}(\mathbf{U}^j) + \left. \frac{\partial \mathbf{G}(\mathbf{U})}{\partial \mathbf{U}} \right|_{\mathbf{U}^j} \Delta \mathbf{U}^{j+1} \\ &= \mathbf{G}(\mathbf{U}^j) + \mathbf{K}_T(\mathbf{U}^j) \Delta \mathbf{U}^{j+1} = \mathbf{0} \end{aligned} \quad (6.86)$$

ergeben sich die Verbesserung des globalen Unbekanntenvektors im Iterationsschritt

$$\Delta \mathbf{U}^{j+1} = -\mathbf{K}_T^{-1}(\mathbf{U}^j) \mathbf{G}(\mathbf{U}^j) \quad (6.87)$$

und der globale Unbekanntenvektor am Iterationsschrittende

$$\mathbf{U}^{j+1} = \mathbf{U}^j + \Delta \mathbf{U}^{j+1} \quad . \quad (6.88)$$

Als Startwert der Iteration wird das konvergierte Ergebnis vom Ende des letzten Zeitschrittes

$$\mathbf{U}^0 = \mathbf{U}_n \quad (6.89)$$

verwendet. Die Iteration wird abgebrochen, wenn die Norm des Residualkraftvektors bzw. die Norm der Änderung des Vektors der globalen Variablen festgelegte

Schranken unterschreiten.

$$\left[\begin{array}{l} \| \mathbf{G}(\mathbf{U}^{j+1}) \| < \text{tol}_{\mathbf{G}} \quad \text{und/oder} \quad \| \Delta \mathbf{U}^{j+1} \| < \text{tol}_{\Delta \mathbf{U}} \end{array} \right] \rightarrow \text{Abbruch} \quad (6.90)$$

Beiträge zur Tangente \mathbf{K}_T liefern alle vom Vektor der Verschiebungen am Zeitschrittende \mathbf{U} abhängigen Summanden des Residualkraftvektor \mathbf{G} . Dies sind die inneren Kräfte \mathbf{F}_{int} , die Massenträgheitskräfte \mathbf{F}_m und die Strafräfte \mathbf{F}_{rnd} und \mathbf{F}_{kon} , die die Einhaltung der wesentlichen Randbedingungen und der Kontaktbedingungen erzwingen.

$$\mathbf{K}_T(\mathbf{U}^j) = \mathbf{K}_{T_{\text{int}}}(\mathbf{U}^j) + \mathbf{K}_{T_m}(\mathbf{U}^j) - \mathbf{K}_{T_{\text{rnd}}}(\mathbf{U}^j) - \mathbf{K}_{T_{\text{kon}}}(\mathbf{U}^j) \quad (6.91)$$

Die Linearisierung wird für eine aus einer Volumenintegration stammenden Kraft (\mathbf{F}_{\diamond} steht für \mathbf{F}_{int} oder \mathbf{F}_m)

$$\mathbf{F}_{\diamond}(\mathbf{U}) = \bigwedge_{p=1}^{N_P} \tilde{\mathbf{F}}_{\diamond p}(\mathbf{U}_p) V_p \quad (6.92)$$

und für eine aus einer Oberflächenintegration stammenden Kraft (\mathbf{F}_{\diamond} steht für \mathbf{F}_{rnd} , oder \mathbf{F}_{kon})

$$\mathbf{F}_{\diamond}(\mathbf{U}) = \bigwedge_{p=1}^{N_{RP}} \tilde{\mathbf{F}}_{\diamond p}(\mathbf{U}_p) V_p \quad (6.93)$$

gezeigt. Mit $(\tilde{\cdot})$ sind im ersten Fall volumenbezogene und im zweiten Fall flächenbezogene Größen gekennzeichnet. V_p ist das einem Integrationspunkt zugeordnete Volumen und A_p die einem Randintegrationspunkt zugeordnete Oberfläche. Der Vektor \mathbf{U}_p beinhaltet die Verschiebungen der Knoten, deren Einflussbereiche den Integrationspunkt überdecken.

Das totale Differential einer aus einer Volumenintegration stammenden Kraft ist

$$\begin{aligned} \Delta \mathbf{F}_{\diamond}(\mathbf{U}) &= \Delta \bigwedge_{p=1}^{N_P} \tilde{\mathbf{F}}_{\diamond p}(\mathbf{U}_p) V_p &= \bigwedge_{p=1}^{N_P} \Delta \tilde{\mathbf{F}}_{\diamond p}(\mathbf{U}_p) V_p \\ &= \bigwedge_{p=1}^{N_P} \frac{\partial \tilde{\mathbf{F}}_{\diamond p}(\mathbf{U}_p)}{\partial \mathbf{U}_p} \Delta \mathbf{U}_p V_p &= \bigwedge_{p=1}^{N_P} \tilde{\mathbf{K}}_{T_{\diamond p}}(\mathbf{U}_p) \Delta \mathbf{U}_p V_p \\ &= \left[\bigwedge_{p=1}^{N_P} \tilde{\mathbf{K}}_{T_{\diamond p}}(\mathbf{U}_p) V_p \right] \Delta \mathbf{U} = \mathbf{K}_{T_{\diamond}}(\mathbf{U}) \Delta \mathbf{U} \quad . \end{aligned} \quad (6.94)$$

Zur Berechnung der globalen Tangente $\mathbf{K}_{T_{\diamond}}$ werden für jeden Integrationspunkt volumenbezogene Tangenten $\tilde{\mathbf{K}}_{T_{\diamond p}}$ berechnet, diese mit den Integrationspunktvo-

lumina multipliziert und assembliert:

$$\mathbf{K}_{T\blacklozenge}(\mathbf{U}) = \bigwedge_{p=1}^{N_p} \frac{\partial \tilde{\mathbf{F}}_{\blacklozenge p}(\mathbf{U}_p)}{\partial \mathbf{U}_p} V_p = \bigwedge_{p=1}^{N_p} \tilde{\mathbf{K}}_{T\blacklozenge p}(\mathbf{U}_p) V_p \quad (6.95)$$

Für den Beitrag eines Integrationspunktes zur Tangente der inneren Kräfte ergibt sich unter Ausnutzung der Beziehungen (6.63) und (6.83)

$$\mathbf{K}_{T\text{int } p} = \frac{\partial \tilde{\mathbf{F}}_{\text{int } p}}{\partial \mathbf{U}_p} V_p = \frac{\partial \tilde{\mathbf{F}}_{\text{int } p}}{\partial \boldsymbol{\sigma}_p} \frac{\partial \boldsymbol{\sigma}_p}{\partial \mathbf{U}_p} V_p = \mathbf{B}^T [\mathbf{C}_T + \alpha_4 \mathbf{D}_T] \mathbf{B} V_p \quad (6.96)$$

mit Materialtangente nach (6.84) und (6.85).

Die globale Tangente der Massenkraft lässt sich unmittelbar aus (6.64) berechnen:

$$\mathbf{K}_{Tm} = \frac{\partial \mathbf{F}_m}{\partial \mathbf{U}} = \alpha_1 \mathbf{M} \quad (6.97)$$

Hierin ist \mathbf{M} die globale Massenmatrix und α_1 ein Parameter des Newmark-Verfahrens. Für den Beitrag eines Integrationspunktes zur Tangente der Massenkraft ergibt sich aus (6.64)

$$\mathbf{K}_{Tm p} = \frac{\partial \tilde{\mathbf{F}}_{mp}}{\partial \mathbf{U}_p} V_p = \alpha_1 \mathbf{L}_p^T \mathbf{L}_p \rho_p V_p = \alpha_1 \mathbf{M}_p \quad (6.98)$$

\mathbf{M}_p ist der Beitrag eines Integrationspunktes zur globalen Massenmatrix \mathbf{M} .

Das totale Differential einer aus einer Oberflächenintegration stammenden Kraft ist

$$\begin{aligned} \Delta \mathbf{F}_{\blacklozenge}(\mathbf{U}) &= \Delta \bigwedge_{p=1}^{N_{RP}} \tilde{\mathbf{F}}_{\blacklozenge p}(\mathbf{U}_p) A_p = \bigwedge_{p=1}^{N_{RP}} \Delta \mathbf{F}_{\blacklozenge p}(\mathbf{U}_p) \\ &= \bigwedge_{p=1}^{N_{RP}} \frac{\partial \mathbf{F}_{\blacklozenge p}(\mathbf{U}_p)}{\partial \mathbf{U}_p} \Delta \mathbf{U}_p = \bigwedge_{p=1}^{N_{RP}} \mathbf{K}_{T\blacklozenge p}(\mathbf{U}_p) \Delta \mathbf{U}_p \\ &= \left[\bigwedge_{p=1}^{N_p} \mathbf{K}_{T\blacklozenge p}(\mathbf{U}_p) \right] \Delta \mathbf{U} = \mathbf{K}_{T\blacklozenge}(\mathbf{U}) \Delta \mathbf{U} \quad (6.99) \end{aligned}$$

Zur Berechnung der globalen Tangente $\mathbf{K}_{T\blacklozenge}$ werden für jeden Randintegrationspunkt die Tangente $\mathbf{K}_{T\blacklozenge p}$ berechnet und assembliert:

$$\mathbf{K}_{T\blacklozenge}(\mathbf{U}) = \bigwedge_{p=1}^{N_{RP}} \frac{\partial \mathbf{F}_{\blacklozenge p}(\mathbf{U}_p)}{\partial \mathbf{U}_p} = \bigwedge_{p=1}^{N_p} \mathbf{K}_{T\blacklozenge p}(\mathbf{U}_p) \quad (6.100)$$

Mit (4.31) und (6.67) ergibt sich für den Beitrag eines Randintegrationspunktes zur

globalen Tangente der Randbedingungsstrafkräfte:

$$\begin{aligned} \mathbf{K}_{T \text{ rnd p}} &= \frac{\partial \mathbf{F}_{\text{rnd p}}}{\partial \mathbf{U}_{\text{p}}} = \mathbf{L}_{\text{p}}^{\text{T}} \frac{\partial \mathbf{f}_{\text{rnd p}}}{\partial \mathbf{U}_{\text{p}}} = \mathbf{L}_{\text{p}}^{\text{T}} \frac{\partial \mathbf{f}_{\text{rnd p}}}{\partial g_{\text{p}}} \frac{\partial g_{\text{p}}}{\partial \mathbf{u}_{\text{p}}} \frac{\partial \mathbf{u}_{\text{p}}}{\partial \mathbf{U}_{\text{p}}} \\ &= -\alpha \mathbf{L}_{\text{p}}^{\text{T}} \mathbf{n}_{\text{p}} \mathbf{n}_{\text{p}}^{\text{T}} \mathbf{L}_{\text{p}} \end{aligned} \quad (6.101)$$

Entsprechend wird mit (4.33) und (6.68) der Beitrag eines Randintegrationspunktes zur globalen Tangente der Kontaktstrafkräfte berechnet:

$$\begin{aligned} \mathbf{K}_{T \text{ kon p}} &= \frac{\partial \mathbf{F}_{\text{kon p}}}{\partial \mathbf{U}_{\text{p}}} = \mathbf{L}_{\text{p}}^{\text{T}} \frac{\partial \mathbf{f}_{\text{kon p}}}{\partial \mathbf{U}_{\text{p}}} = \mathbf{L}_{\text{p}}^{\text{T}} \frac{\partial \mathbf{f}_{\text{kon p}}}{\partial g_{\text{p}}} \frac{\partial g_{\text{p}}}{\partial \mathbf{u}_{\text{p}}} \frac{\partial \mathbf{u}_{\text{p}}}{\partial \mathbf{U}_{\text{p}}} \\ &= \begin{cases} -\alpha \mathbf{L}_{\text{p}}^{\text{T}} \mathbf{n}_{\text{p}} \mathbf{n}_{\text{p}}^{\text{T}} \mathbf{L}_{\text{p}} & \text{für } g < 0 \\ \mathbf{0} & \text{für } g \geq 0 \end{cases} \end{aligned} \quad (6.102)$$

Damit sind alle Bestandteile der globalen Steifigkeitsmatrix nach (6.91) berechnet. Die Linearisierung der Material- und Bilanzgleichungen ist abgeschlossen.

6.5.3 Erweiterung auf große Verformungen

Der größte Nachteil des Deformators als vollständig linearisiertes Verzerrungsmaß ist die fehlende Rotationsinvarianz: Wird ein Körper um endliche Winkel gedreht, so ändert er dabei infolge der Drehung sein Volumen. Wird die Volumenänderung behindert, so entstehen durch die Drehung Spannungen.

Liegen Probleme mit finiten Rotationen aber kleinen Verzerrungen vor, reicht es den Deformator durch ein rotationsinvariantes Verzerrungsmaß zu ersetzen. Die Materialgesetze können beibehalten werden, solange die Verzerrungen hinreichend klein bleiben.

Wenn man bei einer *total Lagrangeschen Betrachtungsweise* bleibt (d. h. wenn die Linearisierung der schwachen Form in der Ausgangskonfiguration erfolgt), ist der Deformator durch den *Green-Lagrangeschen Verzerrungstensor* zu ersetzen. Die \mathbf{B} -Matrizen sind entsprechend anzupassen. Bei der Berechnung der tangentialen Steifigkeitsmatrix ist zusätzlich der Term der *geometrischen Matrix* bzw. *Anfangsspannungsmatrix* zu berücksichtigen (siehe z.B. WRIGGERS (2001)).

Kapitel 7

Algorithmen

In diesem Kapitel werden die verwendeten Algorithmen in knapper Form angegeben. Ausführliche Darstellungen der mathematischer Sachverhalte finden sich in der Literatur zur Numerischen Mathematik (z.B. in VOSS (2001) oder SZYLER (1997)). ENGELN-MÜLLGES & UHLIG (1996) und PRESS (2002) bieten zusätzlich Hinweise zur numerischen Umsetzung und in C bzw. C++ implementierte mathematische Bibliotheken.

7.1 Lösung linearer Gleichungssysteme und linearer Ausgleichsprobleme

Lineare Gleichungssysteme

$$\underset{\langle M \times N \rangle}{\mathbf{A}} \underset{\langle N \rangle}{\mathbf{x}} = \underset{\langle N \rangle}{\mathbf{b}} \quad (7.1)$$

sind nur für den Sonderfall einer quadratischen ($M = N$) und regulären ($\text{Rang}(\mathbf{A})=N$) Koeffizientenmatrix \mathbf{A} lösbar. Im allgemeinen Fall ist stattdessen das dazugehörige Ausgleichsproblem

$$\| \underset{\langle M \times N \rangle}{\mathbf{A}} \underset{\langle N \rangle}{\mathbf{x}} - \underset{\langle N \rangle}{\mathbf{b}} \| \stackrel{!}{=} \min \quad (7.2)$$

zu lösen.

7.1.1 Lösung linearer Gleichungssysteme mit quadratischer und regulärer Koeffizientenmatrix

Zur Lösung linearer Gleichungssysteme mit quadratischer und regulärer Koeffizientenmatrix werden in der Regel Standardverfahren wie z.B. der *Gauß-Algorithmus* oder die *Cholesky-Zerlegung* verwendet. Der numerische Aufwand beider Verfahren ist vergleichbar. Rechenzeit und Speicherbedarf lassen sich reduzieren, wenn die Koeffizientenmatrix besondere Eigenschaften hat und daher spezialisierte Löser eingesetzt werden können. Sind Gleichungssysteme sehr groß, so werden diese aus numerischen Gründen meist iterativ gelöst. Eine ausführliche Diskussion verschiedener Löser findet sich in PRESS (2002).

7.1.2 Lösung linearer Ausgleichsprobleme

Ist \mathcal{L} die Lösungsmenge des Ausgleichsproblems (7.2), so heißt $\tilde{\mathbf{x}} \in \mathcal{L}$ Pseudonormallösung des Ausgleichsproblems, falls $\|\tilde{\mathbf{x}}\| \leq \|\mathbf{x}\|$ für alle $\mathbf{x} \in \mathcal{L}$.

Es gibt genau eine Pseudonormallösung.

Die Matrix \mathbf{A}^\dagger für die durch

$$\underset{\langle N \rangle}{\mathbf{x}} = \underset{\langle N \times M \rangle}{\mathbf{A}^\dagger} \underset{\langle M \rangle}{\mathbf{b}} \quad (7.3)$$

für alle \mathbf{b} die Pseudonormallösung des Ausgleichsproblems gegeben ist, heißt Pseudoinverse von \mathbf{A} .

Die Lösung des Ausgleichsproblems (7.2) mit Hilfe einer Pseudoinvertierung ist mathematisch gleichwertig mit der Anwendung der *Methode der kleinsten Quadrate* (*Least Squares Method*).

Alternativ kann die Pseudoinverse auch über die *Moore-Penrose-Bedingungen* definiert werden:

$$\mathbf{A} \mathbf{A}^\dagger = [\mathbf{A} \mathbf{A}^\dagger]^\top \quad (7.4)$$

$$\mathbf{A}^\dagger \mathbf{A} = [\mathbf{A}^\dagger \mathbf{A}]^\top \quad (7.5)$$

$$\mathbf{A} \mathbf{A}^\dagger \mathbf{A} = \mathbf{A} \quad (7.6)$$

$$\mathbf{A}^\dagger \mathbf{A} \mathbf{A}^\dagger = \mathbf{A}^\dagger \quad (7.7)$$

Die Pseudoinverse hat die Eigenschaften

$$[\mathbf{A}^\dagger]^\dagger = \mathbf{A} \quad (7.8)$$

$$[\mathbf{A}^\dagger]^T = [\mathbf{A}^T]^\dagger, \quad (7.9)$$

aber im Allgemeinen ist

$$[\mathbf{A} \mathbf{B}]^\dagger \neq \mathbf{B}^\dagger \mathbf{A}^\dagger. \quad (7.10)$$

Für den Fall einer regulären quadratischen Matrix \mathbf{A} stimmt die Pseudoinverse mit der Inversen überein ($\mathbf{A}^\dagger = \mathbf{A}^{-1}$). Die Pseudoinverse kann daher als Verallgemeinerung der Inversen angesehen werden.

Bei der Bestimmung der Pseudoinversen ist es sinnvoll hinsichtlich der Dimensionen und des Ranges der Matrix \mathbf{A} drei Fälle zu unterscheiden:

1. $M = N$, $\text{Rang}(\mathbf{A}) = N$:

Ist die Matrix \mathbf{A} quadratisch und regulär, so stimmt ihre Pseudoinverse mit ihrer Inversen überein:

$$\underset{\langle N \times N \rangle}{\mathbf{A}^\dagger} = \underset{\langle N \times N \rangle}{\mathbf{A}^{-1}} \quad (7.11)$$

2. $M > N$, $\text{Rang}(\mathbf{A}) = N$:

Die Beziehung (7.1) wird von links mit \mathbf{A}^T multipliziert:

$$\underset{\langle N \times M \rangle}{\mathbf{A}^T} \underset{\langle M \times N \rangle}{\mathbf{A}} \underset{\langle N \rangle}{\mathbf{x}} = \underset{\langle N \times M \rangle}{\mathbf{A}^T} \underset{\langle M \rangle}{\mathbf{b}} \quad (7.12)$$

Da \mathbf{A} den Rang N hat, hat auch die quadratische Matrix $\mathbf{A}^T \mathbf{A}$ den Rang N , ist regulär und lässt sich invertieren. Damit kann (7.12) nach \mathbf{x} aufgelöst werden:

$$\underset{\langle N \rangle}{\mathbf{x}} = \left[\underset{\langle N \times N \rangle}{\mathbf{A}^T \mathbf{A}} \right]^{-1} \underset{\langle N \times M \rangle}{\mathbf{A}^T} \underset{\langle M \rangle}{\mathbf{b}} \quad (7.13)$$

Wird (7.13) mit der Definition der Pseudoinversen (7.3) verglichen, so ergibt sich für die Pseudoinverse von \mathbf{A} :

$$\underset{\langle N \times M \rangle}{\mathbf{A}^\dagger} = \left[\underset{\langle N \times N \rangle}{\mathbf{A}^T \mathbf{A}} \right]^{-1} \underset{\langle N \times M \rangle}{\mathbf{A}^T} \quad (7.14)$$

3. M und N beliebig, $\text{Rang}(\mathbf{A})$ beliebig :

Die Bestimmung der Pseudoinversen oder ggfs. die numerische Bestimmung der effektiven Pseudoinversen über eine Singulärwertzerlegung sind stets möglich. Die Singulärwertzerlegung wird im Abschnitt 7.1.3 beschrieben.

7.1.3 Singulärwertzerlegung

Eine ausführliche Herleitung der Singulärwertzerlegung findet sich in VOSS (2001). Berechnungsverfahren werden in PRESS (2002) beschrieben. An dieser Stelle werden nur kurz die Eigenschaften der Singulärwertzerlegung und der Zusammenhang zur Pseudoinvertierung dargestellt.

Jede Matrix \mathbf{A} lässt sich in zwei orthogonale Matrizen \mathbf{U} und \mathbf{V} und eine Diagonalmatrix \mathbf{S} zerlegen:

$$\begin{aligned}
 \mathbf{A} &= \mathbf{U} \mathbf{S} \mathbf{V}^T \\
 \langle M \times N \rangle & \quad \langle M \times M \rangle \quad \langle M \times N \rangle \quad \langle N \times N \rangle \\
 \mathbf{U} \mathbf{U}^T &= \mathbf{1} \\
 \langle M \times M \rangle \quad \langle M \times M \rangle & \quad \langle M \times M \rangle \\
 \mathbf{V} \mathbf{V}^T &= \mathbf{1} \\
 \langle N \times N \rangle \quad \langle N \times N \rangle & \quad \langle N \times N \rangle \\
 \left[\mathbf{S} \right]_{ij} &= \begin{cases} s_i & \text{für } i=j \text{ und } i \leq r \\ 0 & \text{ansonsten} \end{cases} \\
 \text{mit } s_1 &\geq s_2 \geq \dots \geq s_r > 0 \\
 \text{und } r &= \text{Rang}(\mathbf{A})
 \end{aligned} \tag{7.15}$$

Die Werte auf der Diagonalen der Matrix \mathbf{S} werden als Singulärwerte bezeichnet. Ist \mathbf{A} symmetrisch, so stimmen die Singulärwerte s_i mit den Eigenwerten λ_i überein. Andernfalls gilt:

$$s_i(\mathbf{A}) = \sqrt{\lambda_i(\mathbf{A} \mathbf{A}^T)} \tag{7.16}$$

D.h. die Singulärwerte von \mathbf{A} sind gleich der Quadratwurzel aus den Eigenwerten von $\mathbf{A} \mathbf{A}^T$. Nach erfolgter Singulärwertzerlegung lässt sich die Pseudoinverse einfach berechnen:

$$\begin{aligned}
 \mathbf{A}^\dagger &= \mathbf{V} \mathbf{S}^\dagger \mathbf{U}^T \\
 \langle N \times M \rangle & \quad \langle N \times N \rangle \quad \langle N \times M \rangle \quad \langle M \times M \rangle \\
 \left[\mathbf{S}^\dagger \right]_{ij} &= \begin{cases} s_i^{-1} & \text{für } i=j \text{ und } s_i > 0 \\ 0 & \text{ansonsten} \end{cases}
 \end{aligned} \tag{7.17}$$

Sehr kleine Singulärwerte kennzeichnen eine schlecht konditionierte Matrix \mathbf{A} . Die Matrixkondition wird verbessert, wenn Singulärwerte, die eine Grenze ε unterschreiten unterdrückt werden. ε ist eine von der Rechnergenauigkeit abhängige kleine Zahl.

Die *effektive Pseudoinverse* wird dann mit

$$\mathbf{A}_{\langle N \times M \rangle}^{\dagger(\text{eff})} = \mathbf{V}_{\langle N \times N \rangle} \mathbf{S}_{\langle N \times M \rangle}^{\dagger(\text{eff})} \mathbf{U}_{\langle M \times M \rangle}^{\text{T}}$$

$$\left[\mathbf{S}_{\langle N \times M \rangle}^{\dagger(\text{eff})} \right]_{ij} = \begin{cases} s_i^{-1} & \text{für } i = j \text{ und } s_i > \varepsilon \\ 0 & \text{ansonsten} \end{cases} \quad (7.18)$$

erhalten.

7.1.4 Matrixkondition

Besteht die Gefahr, dass eine zu invertierende Matrix schlecht konditioniert ist, so sollte die Güte des Ergebnisses der Inversion überprüft werden.

Die Kondition einer Matrix \mathbf{A} wird als das Produkt der (quadratischen) Normen der Matrix und ihrer Inversen definiert:

$$\kappa(\mathbf{A}) = \|\mathbf{A}^{-1}\| \|\mathbf{A}\| \quad (7.19)$$

Dann kann der Verlust an Rechengenauigkeit näherungsweise wie folgt abgeschätzt werden (Voss (2001)):

$$\kappa(\mathbf{A}) = 10^\gamma \quad \leftrightarrow \quad \gamma = \log_{10}(\kappa(\mathbf{A})) \quad (7.20)$$

γ gibt den Verlust an geltenden Ziffern an. Wird z.B. mit doppelter Genauigkeit (d.h. auf üblichen Rechnern mit 15 gültigen Ziffern) gerechnet und verlangt, dass das Ergebnis auf 6 Stellen genau ist, so darf die Konditionszahl κ höchstens $10^\gamma = 10^{15-6} = 10^9$ betragen.

7.2 Iterative Lösung nichtlinearer Gleichungssysteme

Nichtlineare Gleichungssysteme können stets in die Residualform

$$\mathbf{r}_{\langle M \rangle}(\mathbf{x}_{\langle N \rangle}) = \mathbf{0}_{\langle M \rangle} \quad (7.21)$$

gebracht werden. Gesucht wird eine Lösung (oder ggfs. Pseudonormallösung) \mathbf{x} des Gleichungssystems in der Nähe eines Startwertes $\mathbf{x}_{\text{Start}}$. Die Lösung wird im Allgemeinen iterativ bestimmt.

7.2.1 Newton-Raphson-Verfahren

Es wird beschränkt auf nichtlineare Gleichungssysteme, die ebenso viele Gleichungen wie Unbekannte haben ($M = N$):

$$\underset{\langle N \rangle}{\mathbf{r}}(\underset{\langle N \rangle}{\mathbf{x}}) = \underset{\langle N \rangle}{\mathbf{0}} \quad (7.22)$$

\mathbf{x}^k sei ein Vektor, der das Gleichungssystem (7.22) nicht hinreichend erfüllt:

$$\|\mathbf{r}(\mathbf{x}^k)\| \not\leq \varepsilon \quad (7.23)$$

ε ist die Abbruchschranke des Iterationsverfahrens. Von einem verbesserten Lösungsvektor

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta \mathbf{x}^{k+1} \quad (7.24)$$

wird die Erfüllung von (7.22) verlangt:

$$\mathbf{r}(\mathbf{x}^{k+1}) \stackrel{!}{=} \mathbf{0} \quad (7.25)$$

$\mathbf{r}(\mathbf{x}^{k+1})$ wird durch eine nach dem ersten Glied abgebrochene Taylorreihenentwicklung um die Stelle \mathbf{x}^k angenähert:

$$\begin{aligned} \mathbf{r}(\mathbf{x}^{k+1}) &= \mathbf{r}(\mathbf{x}^k + \Delta \mathbf{x}^{k+1}) \\ &= \mathbf{r}(\mathbf{x}^k) + \left. \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}^k} \Delta \mathbf{x}^{k+1} + \text{Restglied}(\mathcal{O}^2) \\ &\approx \mathbf{r}(\mathbf{x}^k) + \left. \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}^k} \Delta \mathbf{x}^{k+1} \end{aligned} \quad (7.26)$$

Aus der vorstehenden Beziehung und der Forderung (7.25) ergibt sich für die Verbesserung des Lösungsvektors:

$$\Delta \mathbf{x}^{k+1} = - \left[\left. \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}^k} \right]^{-1} \mathbf{r}(\mathbf{x}^k) = -\mathbf{K}^{-1}(\mathbf{x}^k) \mathbf{r}(\mathbf{x}^k) \quad (7.27)$$

Damit kann das *Newton-Raphson-Verfahren* zusammengestellt werden.

- ① Der erste Iterationsschritt (Iterationsschrittindex $k = 0$) wird mit der Zuweisung eines Startwert begonnen:

$$\mathbf{x}^0 = \mathbf{x}_{\text{Start}} \quad (7.28)$$

② Der Unbekanntenvektor wird gemäß

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{K}^{-1}(\mathbf{x}^k) \mathbf{r}(\mathbf{x}^k) \quad , \quad \mathbf{K}(\mathbf{x}^k) = \left. \frac{\partial \mathbf{r}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}^k} \quad (7.29)$$

verbessert.

③ Ist das Abbruchkriterium

$$\|\mathbf{r}(\mathbf{x}^{k+1})\| < \varepsilon \quad (7.30)$$

erfüllt, so wird die Iteration beendet, andernfalls wird der Iterationsindex inkrementiert und die Iteration bei ② fortgesetzt.

In der Nähe der Lösung \mathbf{x} weist das Newton-Raphson-Verfahren eine quadratische Konvergenz auf:

$$\|\mathbf{x}^{k+1} - \mathbf{x}\| \leq \|\mathbf{x}^k - \mathbf{x}\|^2 \quad (7.31)$$

7.2.2 Modifiziertes Newton-Raphson-Verfahren

Bei großen Systemen ist die Berechnung und Inversion der Tangente aufwändig. Um den Rechenaufwand zu reduzieren kann die Tangente für mehrere Iterationsschritte konstant gehalten werden. Das Verfahren konvergiert in der Nähe der Lösung dann allerdings nicht mehr quadratisch.

7.2.3 Gauß-Newton-Verfahren

Werden nichtlineare Gleichungssysteme betrachtet, bei denen die Anzahl der Gleichungen die Anzahl der Unbekannten übersteigt ($M > N$), so kann, wenn wie beim Newton-Raphson-Verfahren vorgegangen wird, die Tangente nicht invertiert werden, da sie nicht quadratisch ist. Wird die Invertierung durch eine Pseudoinvertierung (siehe Abschnitt 7.1.2) ersetzt, so erhält man das *Gauß-Newton-Verfahren*. Hierbei wird durch die Newton-Iteration das nichtlineare Ausgleichsproblem in eine Folge linearer Ausgleichprobleme zerlegt, die mittels der *Methode der kleinsten Quadrate* gelöst werden.

7.2.4 Diskretisiertes Newton-Verfahren

Die analytische Berechnung der Tangente $\mathbf{K}(\mathbf{x})$ ist unter Umständen aufwändig und fehleranfällig. Ersatzweise oder zur Kontrolle kann die Tangente auch numerisch

berechnet werden. Hierzu sind nur Auswertungen des Residualvektors notwendig (der Iterationsindex k wird hier zur Vereinfachung fortgelassen):

$$[\mathbf{K}_{\text{num}}]_{ij} = K_{\text{num}ij} = \frac{\partial r_i}{\partial x_j} \approx \frac{r_i(x_1, \dots, x_j + \Delta x_j, \dots, x_N) - r_i(x_1, \dots, x_N)}{\Delta x_j} \quad (7.32)$$

Die Größen Δx_j sind hinreichend klein zu wählen, um eine gute Näherung für die analytische Tangente zu erhalten, und hinreichend groß, um eine numerische Ungenauigkeit durch die Division zweier zu kleiner Zahlen zu vermeiden. In WRIGGERS (2001) wird die Abschätzung

$$\Delta x_j = \nu [\|x_j\| + \tau] \quad \text{mit} \quad \nu = 10^{-3} \dots 10^{-5} > \sqrt{\eta} \quad \text{und} \quad \tau = 10^{-3} \quad (7.33)$$

vorgeschlagen, wobei η die Computergenauigkeit ist.

7.3 Zeitschrittverfahren

Eine ausführliche Darstellung von Zeitschrittverfahren findet sich in WOOD (1990).

7.3.1 Zeitschrittverfahren erster Ordnung

Die allgemeine Form eines Zeitschrittverfahrens erster Ordnung, die das explizite und implizite Euler-Verfahren und die Mittelpunktsregel umfaßt gewinnt man, wenn das Differentialgleichungssystem

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \quad (7.34)$$

zum Zeitpunkt

$$t_{n+\Theta} = [1 - \Theta] t_n + \Theta t_{n+1} = t_n + \Theta [t_{n+1} - t_n] \quad \text{mit} \quad 0 \leq \Theta \leq 1 \quad (7.35)$$

ausgewertet wird

$$\dot{\mathbf{x}}_{n+\Theta} = \mathbf{f}(\mathbf{x}_{n+\Theta}) \quad , \quad (7.36)$$

$\mathbf{x}_{n+\Theta}$ durch einen gewichteten Wert

$$\mathbf{x}_{n+\Theta} \approx [1 - \Theta] \mathbf{x}_n + \Theta \mathbf{x}_{n+1} \quad (7.37)$$

und $\dot{\mathbf{x}}_{n+\Theta}$ durch einen Differenzenquotienten

$$\dot{\mathbf{x}}_{n+\Theta} \approx \frac{\mathbf{x}_{n+1} - \mathbf{x}_n}{\Delta t} \quad (7.38)$$

ersetzt werden.

$$\frac{\mathbf{x}_{n+1} - \mathbf{x}_n}{\Delta t} = \mathbf{f}([1 - \Theta] \mathbf{x}_n + \Theta \mathbf{x}_{n+1}) \quad (7.39)$$

$$\rightarrow \mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \mathbf{f}([1 - \Theta] \mathbf{x}_n + \Theta \mathbf{x}_{n+1}) \quad (7.40)$$

Mit $\Theta = 0$ wird das Euler-Vorwärts-Verfahren, mit $\Theta = \frac{1}{2}$ die Mittelpunktsregel und mit $\Theta = 1$ wird das das Euler-Rückwärts-Verfahren erhalten. Das Euler-Vorwärts-Verfahren ist ein explizites Zeitschrittverfahren, da nach dem Unbekanntenvektor \mathbf{x}_{n+1} aufgelöst werden kann; die übrigen Verfahren sind implizite Zeitschrittverfahren.

7.3.2 Zeitschrittverfahren zweiter Ordnung

Das Standardverfahren zur zeitlichen Diskretisierung von Differentialgleichungssystemen zweiter Ordnung in der Zeit sind das *Newmark-Verfahren* (NEWMARK (1959)) oder Modifikationen dieses Verfahrens (z.B. WOOD (1981), KUHL (1996), DETTMER & PERIC (2003)).

Das Differentialgleichungssystem

$$\mathbf{g}(\underset{\langle N \rangle}{\ddot{\mathbf{x}}}, \underset{\langle N \rangle}{\dot{\mathbf{x}}}, \underset{\langle N \rangle}{\mathbf{x}}) = \underset{\langle N \rangle}{\mathbf{0}} \quad (7.41)$$

wird für den Zeitpunkt t_{n+1} ausgewertet:

$$\mathbf{g}(\ddot{\mathbf{x}}_{n+1}, \dot{\mathbf{x}}_{n+1}, \mathbf{x}_{n+1}) = \mathbf{0} \quad (7.42)$$

Werden in (7.42) die Vektoren \mathbf{x}_{n+1} und $\dot{\mathbf{x}}_{n+1}$ gemäß

$$\mathbf{x}_{n+1} \approx \mathbf{x}_n + \Delta t \dot{\mathbf{x}}_n + \frac{[\Delta t]^2}{2} [1 - 2\beta] \ddot{\mathbf{x}}_n + 2\beta \ddot{\mathbf{x}}_{n+1} \quad (7.43)$$

$$\dot{\mathbf{x}}_{n+1} \approx \dot{\mathbf{x}}_n + \Delta t [1 - \gamma] \ddot{\mathbf{x}}_n + \gamma \ddot{\mathbf{x}}_{n+1} \quad (7.44)$$

ersetzt, so verbleiben die Komponenten des Vektors $\ddot{\mathbf{x}}_{n+1}$ als primäre Unbekannte.

$$\hat{\mathbf{g}}(\ddot{\mathbf{x}}_{n+1}) = \mathbf{0} \quad (7.45)$$

Das Gleichungssystem (7.45) ist im Allgemeinen nichtlinear und muss daher iterativ gelöst werden (siehe Abschnitt 7.2).

Die Wahl der Parameter β und γ beeinflusst die Genauigkeit und die Stabilität des

Verfahrens. Sie sind innerhalb der Grenzen

$$0 \leq \beta \leq \frac{1}{2} \quad \text{und} \quad 0 \leq \gamma \leq 1 \quad (7.46)$$

festzulegen (BATHE (2002)). Newmark hat als unbedingt stabiles Integrationschema mit der Fehlerordnung $\mathcal{O}(\Delta t^2)$ die Methode der konstanten mittleren Beschleunigung (Trapezregel) vorgeschlagen, die mit $\beta = \frac{1}{4}$ und $\gamma = \frac{1}{2}$ erhalten wird. Das Verfahren der zentralen Differenzen wird mit $\beta = 0$ und $\gamma = \frac{1}{2}$ erhalten. Es hat die gleiche Fehlerordnung, ist aber nicht unbedingt stabil.

Werden Probleme aus dem Bereich der Festkörpermechanik behandelt, so liegen meist Anfangs- und Randbedingungen für Verschiebungen vor und es werden Verschiebungsfelder gesucht. Es ist daher vorteilhaft, das nichtlineare Gleichungssystem für Verschiebungen und nicht für deren zweite zeitliche Ableitungen aufzustellen. Hierfür werden die Gleichungen (7.43 und 7.44) unter Verwendung der Abkürzungen

$$\begin{aligned} \alpha_1 &= \frac{1}{\beta [\Delta t]^2} & \alpha_2 &= \frac{1}{\beta \Delta t} & \alpha_3 &= \frac{1 - 2\beta}{2\beta} \\ \alpha_4 &= \frac{\gamma}{\beta \Delta t} & \alpha_5 &= 1 - \frac{\gamma}{\beta} & \alpha_6 &= \left[1 - \frac{\gamma}{2\beta}\right] \Delta t \end{aligned} \quad (7.47)$$

nach $\ddot{\mathbf{x}}_{n+1}$ und $\dot{\mathbf{x}}_{n+1}$ umgestellt

$$\ddot{\mathbf{x}}_{n+1} \approx \alpha_1 [\mathbf{x}_{n+1} - \mathbf{x}_n] - \alpha_2 \dot{\mathbf{x}}_n - \alpha_3 \ddot{\mathbf{x}}_n \quad (7.48)$$

$$\dot{\mathbf{x}}_{n+1} \approx \alpha_4 [\mathbf{x}_{n+1} - \mathbf{x}_n] + \alpha_5 \dot{\mathbf{x}}_n + \alpha_6 \ddot{\mathbf{x}}_n \quad (7.49)$$

und in (7.42) eingesetzt (BATHE (2002)). Das Ergebnis ist ein nichtlineares Gleichungssystem für die Unbekannten \mathbf{x}_{n+1} .

$$\tilde{\mathbf{g}}(\mathbf{x}_{n+1}) = \mathbf{0} \quad (7.50)$$

7.4 Zeitschrittsteuerung

Zur zeitlichen Diskretisierung werden die in Abschnitt 7.3 beschriebenen Zeitschrittverfahren eingesetzt. Die Zeitschrittlänge sollte dergestalt gewählt werden, dass einerseits das Lösungsverfahren konvergiert und die zeitliche Auflösung und die Genauigkeit der Ergebnisse hinreichend hoch sind und andererseits der Rechenaufwand erträglich bleibt. Abschätzungsformeln zur Wahl einer sinnvollen Zeitschrittlänge finden sich in der FE-Literatur (z.B. in BATHE (2002)). Möglichkeiten zu Fehler-

analysen im Rahmen numerischer Berechnungsverfahren werden in GORN (1954) und speziell im Rahmen der Finite-Elemente-Methode in WOOD (1990) dargestellt. Das bekannte implizite FE-Programm *ABAQUS/Standard* nutzt ein auf HIBBITT (1979) zurückgehendes Konzept mit der Bezeichnung *Half-Step Residual Control*. Durch Verfolgung des Residuums der Bilanzgleichungen für den Zeitpunkt $t_n + \Delta t/2$ nachdem die Lösung für $t_n + \Delta t$ bestimmt wurde, kann die Lösungsgenauigkeit abgeschätzt und der Zeitschritt angepasst werden.

Eine für eine Problemstellung sinnvoll abgeschätzte Zeitschrittlänge wird im Folgenden als *Grundzeitschrittlänge* bezeichnet.

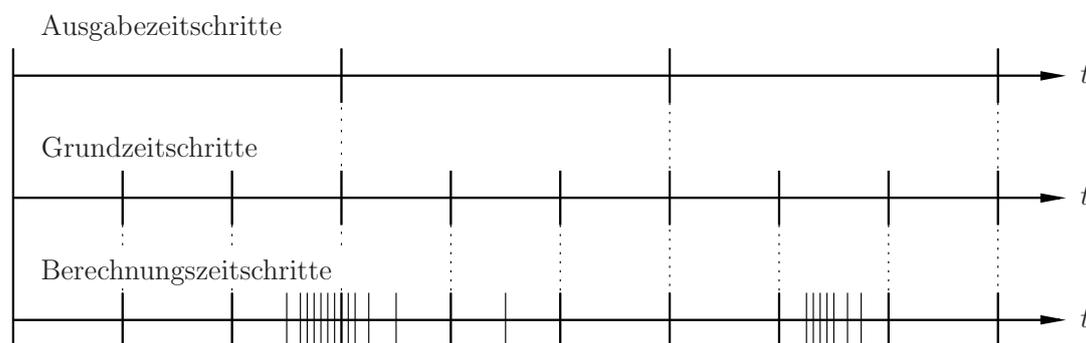


Abbildung 7.1: Zeitschrittsteuerung, Zeitunterteilung

Ist für die Konvergenz des Verfahrens eine niedrigere Zeitschrittlänge erforderlich als für die zeitliche Auflösung der Ergebnisse, so ist es sinnvoll, die Ergebnisse nicht in jedem Zeitschritt auszugeben, da ansonsten unnötig große Datenmengen entstehen und Schreibvorgänge (wenn sie schlecht gepuffert sind) die Rechenzeit erhöhen können. Häufig werden Ergebnisse in Form von „Filmen“ visualisiert. Hierfür sollte der zeitliche Abstand der Ausgaben konstant sein, da andernfalls ein Zeitlupeneffekt (bei verkleinertem Abstand) oder ein Zeitraffereffekt (bei vergrößertem Abstand) im Film auftraten. Für die Ausgabe der Daten wird ein ganzzahliges Vielfaches der Grundzeitschrittlänge festgelegt.

Im Laufe einer Berechnung kann es notwendig werden, die Zeitschrittlänge zu verkleinern, wenn Zeitschritte nicht erfolgreich abgeschlossen werden können. Ein nicht erfolgreicher Zeitschritt liegt vor, wenn es zu keiner oder zu nicht hinreichender (zu langsamer) Konvergenz in der Materialiteration oder in der Bilanziteration kommt oder wenn das Berechnungsergebnis nicht hinreichend genau ist. Ursachen hierfür können z.B. das Auftreten von Kontakt oder eine sprunghafte Materialantwort auf die Belastung sein.

Der verwendete Algorithmus zur automatischen Zeitschrittlängen Anpassung erfüllt folgende Bedingungen:

1. Die Zeitschrittlänge wird verkleinert, wenn ein Zeitschritt nicht erfolgreich ist.
2. Die Zeitschrittlänge wird nach einem erfolgreichen Zeitschritt bis maximal zur Grundzeitschrittlänge vergrößert, sofern dies im Rahmen des Algorithmus zulässig ist.
3. Alle Zeitpunkte, die Vielfache der Grundzeitschrittlänge sind, werden abgedeckt.

Das Schema in Abbildung 7.1 veranschaulicht anhand eines Beispiels die zeitliche Diskretisierung.

Kapitel 8

Rechenregeln

8.1 Voigtsche Notation

Mit Hilfe der Voigtschen Notation können symmetrische Größen platzsparend abgespeichert und Rechenoperationen zwischen symmetrischen Größen ökonomischer ausgeführt werden. Dies wird erreicht, indem die symmetrischen Komponenten einer Größe nur einmal abgespeichert werden.¹ Ein symmetrischer Tensor zweiter Stufe wird in einen Voigtschen Vektor und ein in den ersten beiden und letzten beiden Indizes symmetrischer Tensor vierter Stufe in einen Voigtschen Tensor zweiter Stufe umgewandelt. Das doppelte Skalarprodukt wird auf ein einfaches Skalarprodukt zwischen Objekten in Voigtscher Darstellung zurückgeführt.

Häufig findet man in der Literatur ein uneinheitliches Übersetzungsschema: Spannungsmaße werden anders übersetzt als Verzerrungsmaße, womit sich dann auch andere Rechenregeln für Spannungsmaße und Verzerrungsmaße ergeben.

In dieser Arbeit wird einem symmetrischen Übersetzungsschema der Vorzug gegeben. Eine ausführliche Darstellung findet sich in TROSTEL (1993). An dieser Stelle werden nur die Übersetzungsschemata für symmetrische Tensoren zweiter und vierter Stufe im dreidimensionalen Raum angegeben. Voigtsche Größen werden durch das hochgestellte Symbol „ $\overset{3}{(\cdot)}$ “ gekennzeichnet. Die Übersetzungsschemata für Größen im zweidimensionalen Raum ergeben sich, indem bei den ursprünglichen Größen alle mit einem Index „3“ behafteten Komponenten und bei den übersetzten Größen alle mit den Indizes „5“ und „6“ behafteten Komponenten gestrichen werden.

¹Ein modernerer Ansatz ist die Nutzung einer geeigneter Matrixklasse. Der Matrixklasse werden bei der Konstruktion eines Objektes die Eigenschaften dieses Objektes (symmetrisch, antimetrisch, dünn besetzt, u.s.w.) übergeben. Es ist dann Aufgabe der Matrixklasse, dieses Objekt ökonomisch abzuspeichern und optimierte Rechenoperationen zwischen Objekten mit besonderen Eigenschaften zur Verfügung zu stellen.

Ein symmetrischer Tensor zweiter Stufe

$$\mathbf{A} = \sum_{i,j=1}^3 A_{ij} \mathbf{e}_i \otimes \mathbf{e}_j \stackrel{\wedge}{=} \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{12} & A_{22} & A_{23} \\ A_{13} & A_{23} & A_{33} \end{bmatrix} \quad (\text{bzgl. Basis } \mathbf{e}_i) \quad (8.1)$$

hat die Voigtsche Darstellung

$$\mathbf{A} = \sum_{i=1}^6 \overset{\vee}{A}_i \overset{\vee}{\mathbf{e}}_i \stackrel{\wedge}{=} \begin{pmatrix} \overset{\vee}{A}_1 \\ \overset{\vee}{A}_2 \\ \overset{\vee}{A}_3 \\ \overset{\vee}{A}_4 \\ \overset{\vee}{A}_5 \\ \overset{\vee}{A}_6 \end{pmatrix} \quad (\text{bzgl. Basis } \overset{\vee}{\mathbf{e}}_i) = \begin{pmatrix} A_{11} \\ A_{22} \\ A_{33} \\ \sqrt{2} A_{12} \\ \sqrt{2} A_{23} \\ \sqrt{2} A_{13} \end{pmatrix} \quad (\text{bzgl. Basis } \overset{\vee}{\mathbf{e}}_i) \quad (8.2)$$

Die Voigtschen Basisvektoren $\overset{\vee}{\mathbf{e}}_i$ (mit $i = 1, \dots, 6$) lassen sich durch die Basisdyaden $\mathbf{e}_i \otimes \mathbf{e}_j$ (mit $i = 1, \dots, 3$ und $j = 1, \dots, 3$) darstellen:

$$\begin{aligned} \overset{\vee}{\mathbf{e}}_1 &= \mathbf{e}_1 \otimes \mathbf{e}_1 & \overset{\vee}{\mathbf{e}}_2 &= \mathbf{e}_2 \otimes \mathbf{e}_2 & \overset{\vee}{\mathbf{e}}_3 &= \mathbf{e}_3 \otimes \mathbf{e}_3 \\ \overset{\vee}{\mathbf{e}}_4 &= \sqrt{2} \mathbf{e}_1 \otimes \mathbf{e}_2 & \overset{\vee}{\mathbf{e}}_5 &= \sqrt{2} \mathbf{e}_2 \otimes \mathbf{e}_3 & \overset{\vee}{\mathbf{e}}_6 &= \sqrt{2} \mathbf{e}_1 \otimes \mathbf{e}_3 \end{aligned} \quad (8.3)$$

Für einen in den ersten beiden und letzten beiden Indizes symmetrischen Tensor vierter Stufe

$$\overset{(4)}{\mathbf{C}} = \sum_{i,j,k,l=1}^3 C_{ijkl} \mathbf{e}_i \otimes \mathbf{e}_j \otimes \mathbf{e}_k \otimes \mathbf{e}_l \stackrel{\wedge}{=} \begin{bmatrix} C_{1111} & C_{1112} & C_{1113} & C_{1211} & C_{1212} & C_{1213} & C_{1311} & C_{1312} & C_{1313} \\ C_{1112} & C_{1122} & C_{1123} & C_{1212} & C_{1222} & C_{1223} & C_{1312} & C_{1322} & C_{1323} \\ C_{1113} & C_{1123} & C_{1133} & C_{1213} & C_{1223} & C_{1233} & C_{1313} & C_{1323} & C_{1333} \\ \hline C_{1211} & C_{1212} & C_{1213} & C_{2211} & C_{2212} & C_{2213} & C_{2311} & C_{2312} & C_{2313} \\ C_{1212} & C_{1222} & C_{1223} & C_{2212} & C_{2222} & C_{2223} & C_{2312} & C_{2322} & C_{2323} \\ C_{1213} & C_{1223} & C_{1233} & C_{2213} & C_{2223} & C_{2233} & C_{2313} & C_{2323} & C_{2333} \\ \hline C_{1311} & C_{1312} & C_{1313} & C_{2311} & C_{2312} & C_{2313} & C_{3311} & C_{3312} & C_{3313} \\ C_{1312} & C_{1322} & C_{1323} & C_{2312} & C_{2322} & C_{2323} & C_{3312} & C_{3322} & C_{3323} \\ C_{1313} & C_{1323} & C_{1333} & C_{2313} & C_{2323} & C_{2333} & C_{3313} & C_{3323} & C_{3333} \end{bmatrix} \quad (\text{bzgl. Basis } \mathbf{e}_i) \quad (8.4)$$

(auf der rechten Seite der vorstehenden Gleichung wurde eine Hypermatrixdarstellung gewählt) ergibt sich

$$\begin{aligned}
\stackrel{(4)}{\mathbf{C}} &= \sum_{i,j=1}^6 \overset{\mathfrak{V}}{C}_{ij} \overset{\mathfrak{V}}{\mathbf{e}}_i \otimes \overset{\mathfrak{V}}{\mathbf{e}}_j \\
&\stackrel{\wedge}{=} \begin{bmatrix} \overset{\mathfrak{V}}{C}_{11} & \overset{\mathfrak{V}}{C}_{12} & \overset{\mathfrak{V}}{C}_{13} & \overset{\mathfrak{V}}{C}_{14} & \overset{\mathfrak{V}}{C}_{15} & \overset{\mathfrak{V}}{C}_{16} \\ \overset{\mathfrak{V}}{C}_{21} & \overset{\mathfrak{V}}{C}_{22} & \overset{\mathfrak{V}}{C}_{23} & \overset{\mathfrak{V}}{C}_{24} & \overset{\mathfrak{V}}{C}_{25} & \overset{\mathfrak{V}}{C}_{26} \\ \overset{\mathfrak{V}}{C}_{31} & \overset{\mathfrak{V}}{C}_{32} & \overset{\mathfrak{V}}{C}_{33} & \overset{\mathfrak{V}}{C}_{34} & \overset{\mathfrak{V}}{C}_{35} & \overset{\mathfrak{V}}{C}_{36} \\ \overset{\mathfrak{V}}{C}_{41} & \overset{\mathfrak{V}}{C}_{42} & \overset{\mathfrak{V}}{C}_{43} & \overset{\mathfrak{V}}{C}_{44} & \overset{\mathfrak{V}}{C}_{45} & \overset{\mathfrak{V}}{C}_{46} \\ \overset{\mathfrak{V}}{C}_{51} & \overset{\mathfrak{V}}{C}_{52} & \overset{\mathfrak{V}}{C}_{53} & \overset{\mathfrak{V}}{C}_{54} & \overset{\mathfrak{V}}{C}_{55} & \overset{\mathfrak{V}}{C}_{56} \\ \overset{\mathfrak{V}}{C}_{61} & \overset{\mathfrak{V}}{C}_{62} & \overset{\mathfrak{V}}{C}_{63} & \overset{\mathfrak{V}}{C}_{64} & \overset{\mathfrak{V}}{C}_{65} & \overset{\mathfrak{V}}{C}_{66} \end{bmatrix} \quad (\text{bzgl. Basis } \overset{\mathfrak{V}}{\mathbf{e}}_i) \\
&= \begin{bmatrix} C_{1111} & C_{1122} & C_{1133} & \sqrt{2} C_{1112} & \sqrt{2} C_{1123} & \sqrt{2} C_{1113} \\ C_{2211} & C_{2222} & C_{2233} & \sqrt{2} C_{2212} & \sqrt{2} C_{2223} & \sqrt{2} C_{2213} \\ C_{3311} & C_{3322} & C_{3333} & \sqrt{2} C_{3312} & \sqrt{2} C_{3323} & \sqrt{2} C_{3313} \\ \sqrt{2} C_{1211} & \sqrt{2} C_{1222} & \sqrt{2} C_{1233} & 2 C_{1212} & 2 C_{1223} & 2 C_{1213} \\ \sqrt{2} C_{2311} & \sqrt{2} C_{2322} & \sqrt{2} C_{2333} & 2 C_{2212} & 2 C_{2323} & 2 C_{2313} \\ \sqrt{2} C_{1311} & \sqrt{2} C_{1322} & \sqrt{2} C_{1333} & 2 C_{1312} & 2 C_{1323} & 2 C_{1313} \end{bmatrix} \quad (\text{bzgl. Basis } \overset{\mathfrak{V}}{\mathbf{e}}_i) \\
&\quad (8.5)
\end{aligned}$$

Dieser Voigtsche Tensor zweiter Stufe ist im Allgemeinen unsymmetrisch. Nur vollständig symmetrische Tensoren vierter Stufe (neben $C_{ijkl} = C_{ijlk}$ und $C_{ijkl} = C_{jikl}$ muss auch noch $C_{ijkl} = C_{klij}$ gelten) ergeben symmetrische Voigtsche Tensoren zweiter Stufe.

Es ergeben sich einfache Rechenregeln für die in die konsistente Voigtsche Notation übersetzten Größen, da eine doppelte Basisvektorüberschiebung der ursprünglichen Größen durch eine einfache Basisvektorüberschiebung der Voigtschen Größen ersetzt wird. Z.B. gilt für einen symmetrischen Tensor zweiter Stufe \mathbf{A} und einen in den ersten beiden und den letzten beiden Indizes symmetrischen Tensor vierter Stufe $\stackrel{(4)}{\mathbf{C}}$, wenn das einfache skalare Produkt zwischen Voigtschen Größen mit „ \odot “ symbolisiert wird

$$\|\mathbf{A}\| = \sqrt{\mathbf{A} : \mathbf{A}} = \sqrt{\overset{\mathfrak{V}}{\mathbf{A}} \odot \overset{\mathfrak{V}}{\mathbf{A}}}$$

und

$$\stackrel{(4)}{\mathbf{C}} : \mathbf{A} = \overset{\mathfrak{V}}{\mathbf{C}} \odot \overset{\mathfrak{V}}{\mathbf{A}} \quad .$$

8.2 Einheitensysteme

Das Programmsystem nimmt keine Einheitenrechnungen vor. Alle Eingaben haben in zueinander passenden Einheiten zu erfolgen. Die Tabelle 8.1 gibt Hilfestellung, sofern von den SI-Grundeinheiten abgewichen werden soll.

Zeit	Länge	Masse	Kraft	Dichte	Beschleunigung	Spannung
ms	mm	g	N	$\frac{\text{g}}{\text{mm}^3}$	$\frac{\text{mm}}{\text{ms}^2} = \frac{\text{N}}{\text{g}} = \frac{\text{kN}}{\text{kg}}$	$\frac{\text{N}}{\text{mm}^2} = \frac{\text{MN}}{\text{m}^2}$
ms	mm	kg	kN	$\frac{\text{kg}}{\text{mm}^3}$	$\frac{\text{mm}}{\text{ms}^2} = \frac{\text{kN}}{\text{kg}}$	$\frac{\text{kN}}{\text{mm}^2} = \frac{\text{GN}}{\text{m}^2}$
ms	mm	t	MN	$\frac{\text{t}}{\text{mm}^3}$	$\frac{\text{mm}}{\text{ms}^2} = \frac{\text{MN}}{\text{t}} = \frac{\text{kN}}{\text{kg}}$	$\frac{\text{MN}}{\text{mm}^2} = \frac{\text{TN}}{\text{m}^2}$
ms	m	g	kN	$\frac{\text{g}}{\text{m}^3}$	$\frac{\text{m}}{\text{ms}^2} = \frac{\text{kN}}{\text{g}} = \frac{\text{MN}}{\text{kg}}$	$\frac{\text{kN}}{\text{m}^2}$
ms	m	kg	MN	$\frac{\text{kg}}{\text{m}^3}$	$\frac{\text{m}}{\text{ms}^2} = \frac{\text{MN}}{\text{kg}}$	$\frac{\text{MN}}{\text{m}^2}$
ms	m	t	GN	$\frac{\text{t}}{\text{m}^3}$	$\frac{\text{m}}{\text{ms}^2} = \frac{\text{GN}}{\text{t}} = \frac{\text{MN}}{\text{kg}}$	$\frac{\text{GN}}{\text{m}^2}$
s	mm	g	μN	$\frac{\text{g}}{\text{mm}^3}$	$\frac{\text{mm}}{\text{s}^2} = \frac{\mu\text{N}}{\text{g}} = \frac{\text{mN}}{\text{kg}}$	$\frac{\mu\text{N}}{\text{mm}^2} = \frac{\text{N}}{\text{m}^2}$
s	mm	kg	mN	$\frac{\text{kg}}{\text{mm}^3}$	$\frac{\text{mm}}{\text{s}^2} = \frac{\text{mN}}{\text{kg}}$	$\frac{\text{mN}}{\text{mm}^2} = \frac{\text{kN}}{\text{m}^2}$
s	mm	t	N	$\frac{\text{t}}{\text{mm}^3}$	$\frac{\text{mm}}{\text{s}^2} = \frac{\text{N}}{\text{t}} = \frac{\text{mN}}{\text{kg}}$	$\frac{\text{N}}{\text{mm}^2} = \frac{\text{MN}}{\text{m}^2}$
s	m	g	mN	$\frac{\text{g}}{\text{m}^3}$	$\frac{\text{m}}{\text{s}^2} = \frac{\text{mN}}{\text{g}} = \frac{\text{N}}{\text{kg}}$	$\frac{\text{mN}}{\text{m}^2}$
s	m	kg	N	$\frac{\text{kg}}{\text{m}^3}$	$\frac{\text{m}}{\text{s}^2} = \frac{\text{N}}{\text{kg}}$	$\frac{\text{N}}{\text{m}^2}$
s	m	t	kN	$\frac{\text{t}}{\text{m}^3}$	$\frac{\text{m}}{\text{s}^2} = \frac{\text{kN}}{\text{t}} = \frac{\text{N}}{\text{kg}}$	$\frac{\text{kN}}{\text{m}^2}$

Tabelle 8.1: Einheitensysteme

Kapitel 9

Beispiele

9.1 Übersicht

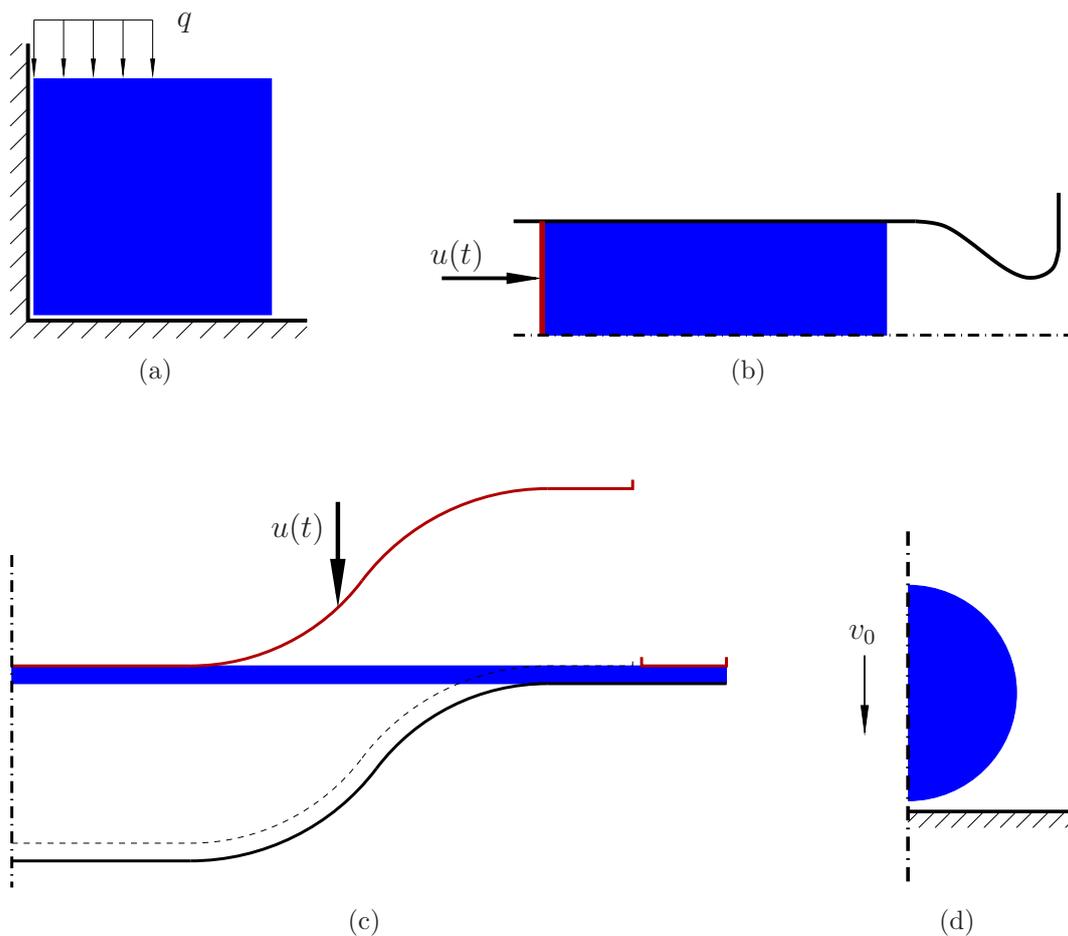


Abbildung 9.1: Berechnungsbeispiele: (a) Quadratscheibe unter Teillast, (b) Extrusion, (c) Tiefziehen, (d) Aufprall

Anhand der folgenden numerischen Beispiele (siehe Abbildung 9.1) wird das Pro-

grammsystem überprüft. Außerdem wird die *Finite-Punkte-Methode* mit der *Finite-Elemente-Methode* verglichen.

Bei den Berechnungsbeispielen wird, sofern nichts anderes angegeben ist, ein ebener Verzerrungszustand angenommen. Alle Programmeingaben und Programmausgaben erfolgen in zueinander passenden Einheiten. Es wird entweder das [kg, m, s, N]-Einheitensystem oder das [t, mm, s, N]-Einheitensystem verwendet (siehe auch Tabelle 8.1).

Unter <http://www.schilling-berlin.de/diss.htm> stehen das mit dem Compiler *gcc 4.02* unter dem Betriebssystem *Suse Linux V10.0* (Kernel 2.6.13-158) übersetzte Programm und die Eingabedateien zu den Berechnungsbeispielen zum Herunterladen zur Verfügung.

9.2 Zeitmessungen

Bei der Gestaltung des Programmsystems wurde Übersichtlichkeit und Erweiterbarkeit ein höherer Stellenwert beigemessen als der Ausführungsgeschwindigkeit. Da absolute Rechenzeiten durch die eingesetzte Hardware und den Optimierungsgrad des Programmsystems bestimmt werden und ein schnellerer Rechner oder ein höherer Optimierungsgrad für einen schnelleren Berechnungsablauf aller Methoden sorgt, ist das Ziel der Zeitmessungen in erster Linie die Ermittlung von Rechenzeitverhältnissen. Die Ergebnisse der Zeitmessungen beider Methoden sind gut vergleichbar, da die *Finite-Punkte-Methode* und die *Finite-Elemente-Methode* größtenteils den selben Quellcode nutzen und die restlichen Quellcodeteile „im selben Stil“ programmiert wurden.

Die Zeitmessungen wurden auf einem *IBM Thinkpad R52* (Typ 1846) mit 2 GB Hauptspeicher unter *Suse Linux V10.0* (Kernel 2.6.13-158) durchgeführt. Zeitmessungen und Berechnungen wurden über ein Shellsript gestartet. Um gegebenenfalls Messfehler erkennen zu können, wurden die Messreihen dreimal wiederholt.

9.3 Quadratscheibe unter Teillast

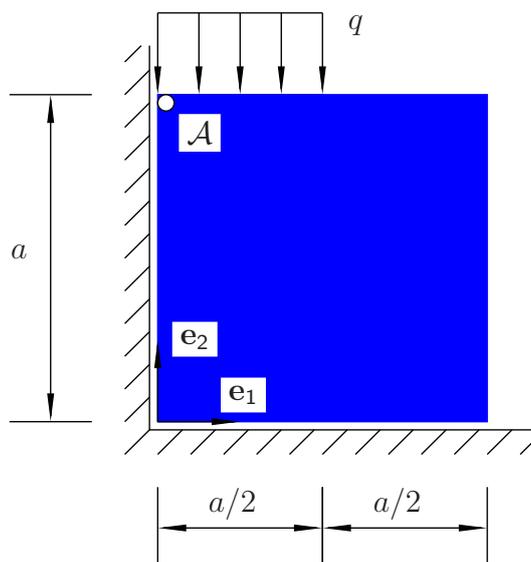


Abbildung 9.2: Beispiel Quadratscheibe unter Teillast, System

Die in der Abbildung 9.2 gezeigte quadratische Scheibe mit der Abmessung $a = 2$ m aus linear elastischem Material (Elastizitätsmodul $E = 100 \text{ N/m}^2$ und Querkontraktionszahl $\nu \in \{0.400, 0.495, 0.499\}$) wird unter einer Teilstreckenlast ($q = 50 \text{ N/m}$) inhomogen verformt. Ein ähnliches Modellproblem wird von FERNÁNDEZ-MÉNDEZ & HUERTA (2004) verwendet, um die Erzwingung wesentlicher Randbedingungen im Rahmen netzfreier Methoden mit unterschiedlichen Verfahren zu untersuchen.

Es wird keine Massenträgheit berücksichtigt, d. h. die Berechnung erfolgt quasi-statisch. Da die Theorie kleiner Verformungen und linear elastisches Materialverhalten verwendet werden, sind die Bestimmungsgleichungen linear. Die Last kann daher in einem Schritt aufgebracht werden, und Konvergenz stellt sich innerhalb eines Iterationsschrittes ein.

Anhand des ersten Berechnungsbeispiels werden die optimalen Wichtungsfunktionsausdehnungen für die MLS-Approximation (siehe Abschnitt 5.5.3) ermittelt und die Konvergenzeigenschaften der *Finite-Punkte-Methode* mit denen der *Finite-Elemente-Methode* verglichen.

Die Abbildung 9.3 zeigt eine mögliche Diskretisierung des Körpers mit finiten Punkten. Da der Körper regelmäßig berandet ist, ist es leicht möglich das Knotennetz so zu legen, dass Knoten genau auf der Berandung des Körpers liegen, womit die Verschiebungsrandbedingungen richtig vorgegeben werden können. Die Definitionsbereiche der Wichtungsfunktionen der Randknoten umfassen dann aber auch Bereiche, in denen sich kein Material befindet. Daher wird im Randbereich das Volumen

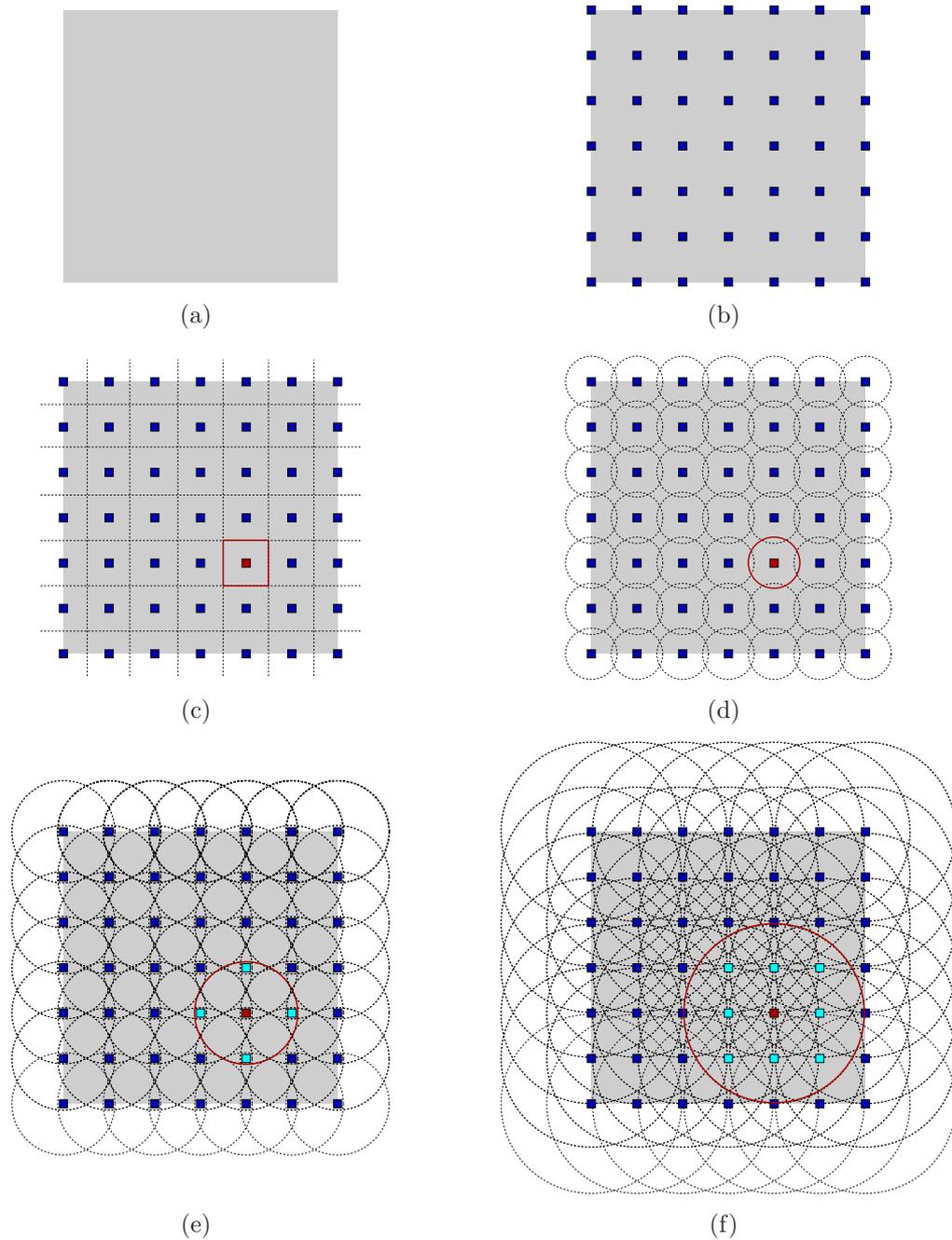


Abbildung 9.3: Diskretisierung mit finiten Punkten: (a) zu diskretisierender Körper, (b) regelmäßiges Knotennetz, (c) den Knoten zugeordnete quadratische Flächen, (d) flächengleiche Kreise mit Radius h_0 , (e) Wichtungsfunktionsbereiche mit $h = 2 h_0$, (f) Wichtungsfunktionsbereiche mit $h = 3.50 h_0$

des Körpers überschätzt, wenn für die Randknoten keine an den Randverlauf angepassten Wichtungsfunktionen definiert werden.

Wird umgekehrt das Knotennetz so gewählt, dass die Summe der den Knoten zugeordneten Volumina gleich dem Volumen des Körpers ist, so liegen die äußersten Knoten innerhalb des Randes und die Verschiebungsrandbedingungen werden ungenau vorgegeben.

Im allgemeinen Falle werden zu diskretisierende Körper unregelmäßig berandet und mit einem regelmäßigen Knotennetz überdeckt sein. Der räumliche Diskretisierungsfehler bei der Angabe von Randbedingungen auf Randknoten hat dann die Größenordnung des halben Knotenabstandes. Damit ist erkennbar, dass das Verfahren schlecht geeignet ist für die Berechnung dünnwandiger Bauteile.

9.3.1 Wichtungsfunktionsausdehnungen

Die Ausdehnung der Wichtungsfunktionen um die Knoten hat bei der Finite-Punkte-Methode einen entscheidenden Einfluss auf die Güte der Berechnungsergebnisse.

Je höher die gewünschte Ansatzfunktionsordnung ist, desto größer muss der Knoteneinflussbereich gewählt werden, damit die Ansatzfunktionen dieser Ordnung berechnet werden können. Bei einer zu kleinen Ausdehnung der Wichtungsfunktionen können nur noch MLS-Ansatzfunktionen 0-ter Ordnung (*Shepard-Interpolation*) berechnet werden. Das Strukturverhalten wird zu steif wiedergegeben (*Locking*).

Eine zu große Ausdehnung der Wichtungsfunktionen führt zu numerischen Schwierigkeiten. Die Knoten weisen unphysikalische gegenseitige Verschiebungen auf. Hier ist eine Parallele zum *Hourglassing* in der Finite-Elemente-Methode zu erkennen, bei dem Elementknoten unphysikalische Verschiebungsmuster zeigen, da einzelnen Elementeigenformen unzureichende Steifigkeiten zugeordnet sind.

Variiert wird der Überlappungsparameter α , der die Ausdehnung der Wichtungsfunktionen gemäß Gleichung (5.50) definiert. Es wird eine Diskretisierung mit $7 \cdot 7 = 49$ Knoten verwendet. Als Querkontraktionszahl wird $\nu = 0.4$ gewählt. Verglichen werden die Vertikalverschiebungen des Punktes \mathcal{A} (Tabelle 9.1) und die Bilder der deformierten Netze (Abbildungen 9.4 bis 9.8). In der Tabelle (9.1) sind die Ergebnisse für die Vertikalverschiebungen, bei denen die zugehörigen Netzbilder ein unphysikalisches Strukturverhalten widerspiegeln, hellgrau dargestellt. Die übrigen Tabellenzellen definieren die zulässigen Variationsbereiche des Überlappungsparameters α in Abhängigkeit von der Ansatzfunktionsordnung N_G . Optimale (sichere) Werte für α liegen in den Mitten der zulässigen Bereiche. Eine Zusammenstellung

zeigt Tabelle 9.2. In den folgenden Berechnungsbeispielen werden stets die optimalen Werte für die Wichtungsfunktionsüberlappung verwendet.

α	Ansatzfunktionsordnung / Anzahl Basisfunktionen				
	0 / 1	1 / 3	2 / 6	3 / 10	4 / 15
2.00	0.2997	0.6560	0.6560	0.6560	0.6560
2.50	0.4017	0.8135	0.8135	0.8135	0.8135
3.00	0.6004	0.7748	0.7847	0.7846	0.7845
3.50	0.7644	0.7596	0.7515	0.7522	0.7522
4.00	0.8777	0.7612	0.7695	0.7708	0.7710
4.50	0.9818	0.7613	0.7717	0.7762	0.7763
5.00	1.0960	0.7619	0.7744	0.7817	0.7814
5.50	—	0.7617	0.7717	0.7879	0.7898
6.00	—	0.7643	0.7701	0.7890	0.7941
6.50	—	0.7918	0.7682	0.7871	0.8149
7.00	—	0.9484	0.7682	0.7877	0.8201
7.50	—	—	—	0.7819	0.8242
8.00	—	—	—	—	0.8286

Tabelle 9.1: Beispiel Quadratscheibe unter Teillast, berechnet mit Finite-Punkte-Methode, Vertikalverschiebungen [m] des Punktes \mathcal{A} in Abhängigkeit vom Wichtungsfunktionsüberlappungsparameter α

N_G	0	1	2	3	4
α	(3.25)	3.50	4.00	4.50	4.50

Tabelle 9.2: Optimaler Wichtungsfunktionsüberlappungsparameter α in Abhängigkeit von der Ansatzfunktionsordnung N_G

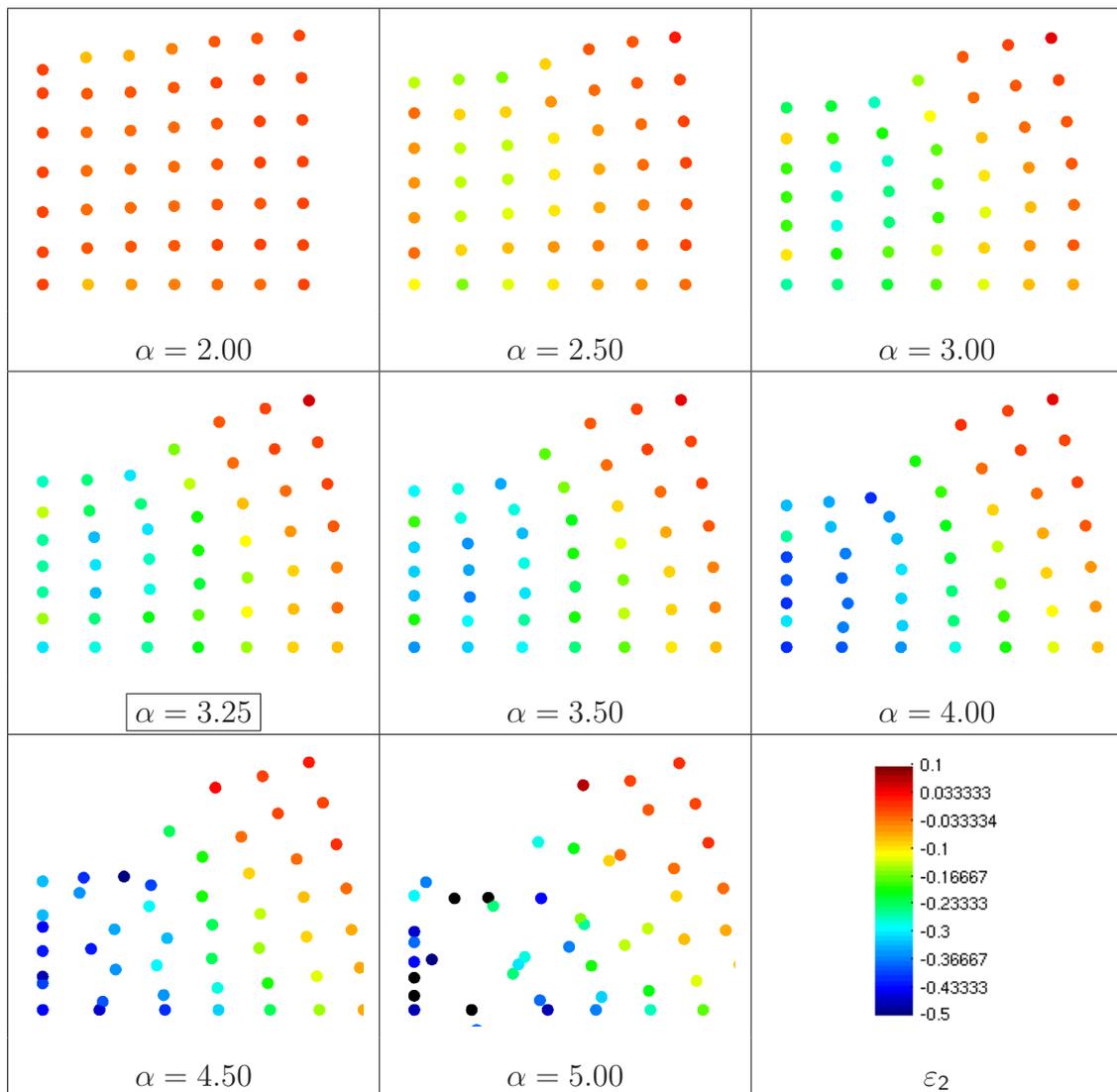


Abbildung 9.4: Beispiel Quadratscheibe unter Teillast, verformte Netze und Dehnungen ε_2 , berechnet mit FPM, Ansatzfunktionsordnung $N_G = 0$, Variation des Wichtungsfunktionsüberlappungsparameters α

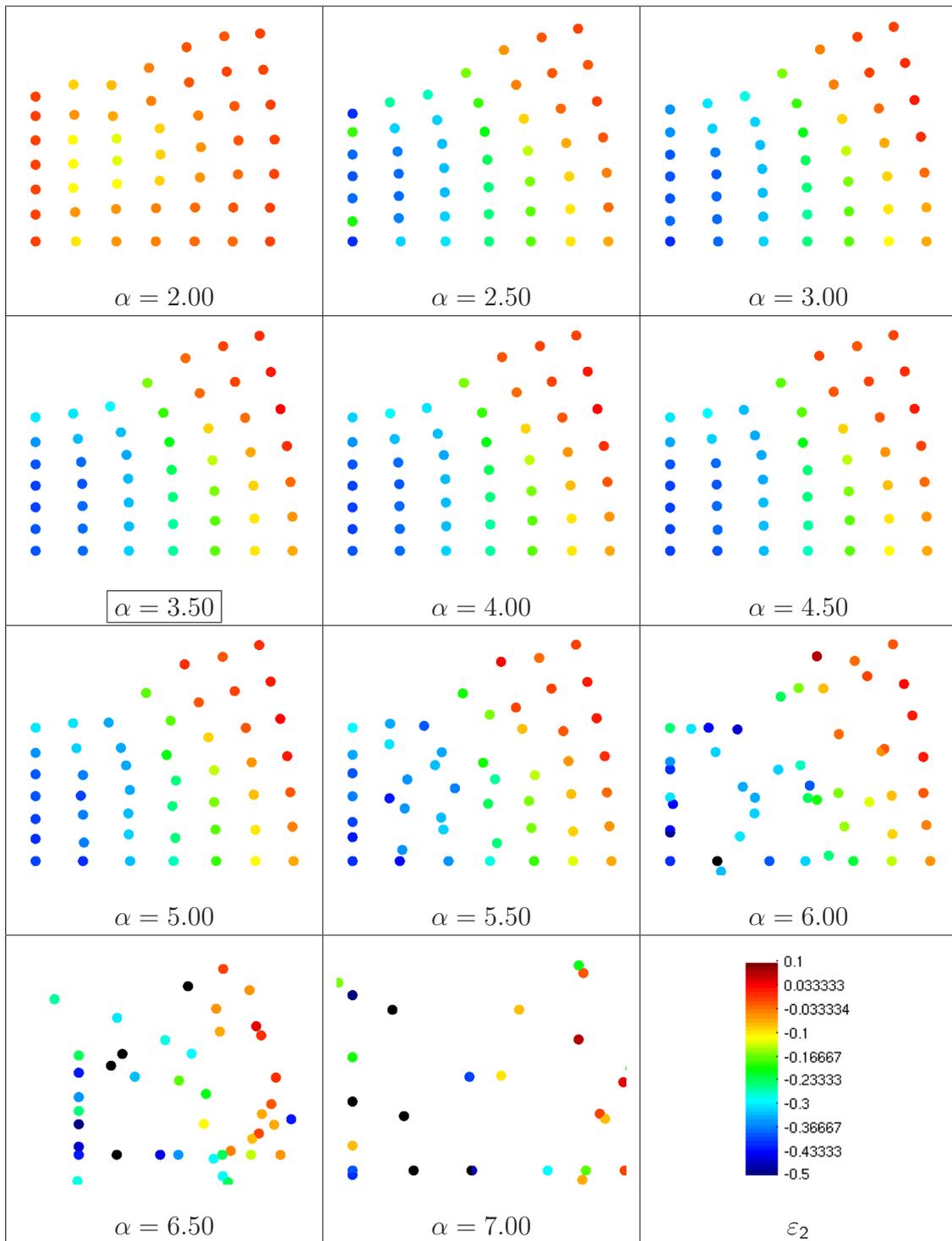


Abbildung 9.5: Beispiel Quadratscheibe unter Teillast, verformte Netze und Dehnungen ε_2 , berechnet mit FPM, Ansatzfunktionsordnung $N_G = 1$, Variation des Wichtungsfunktionsüberlappungsparameters α

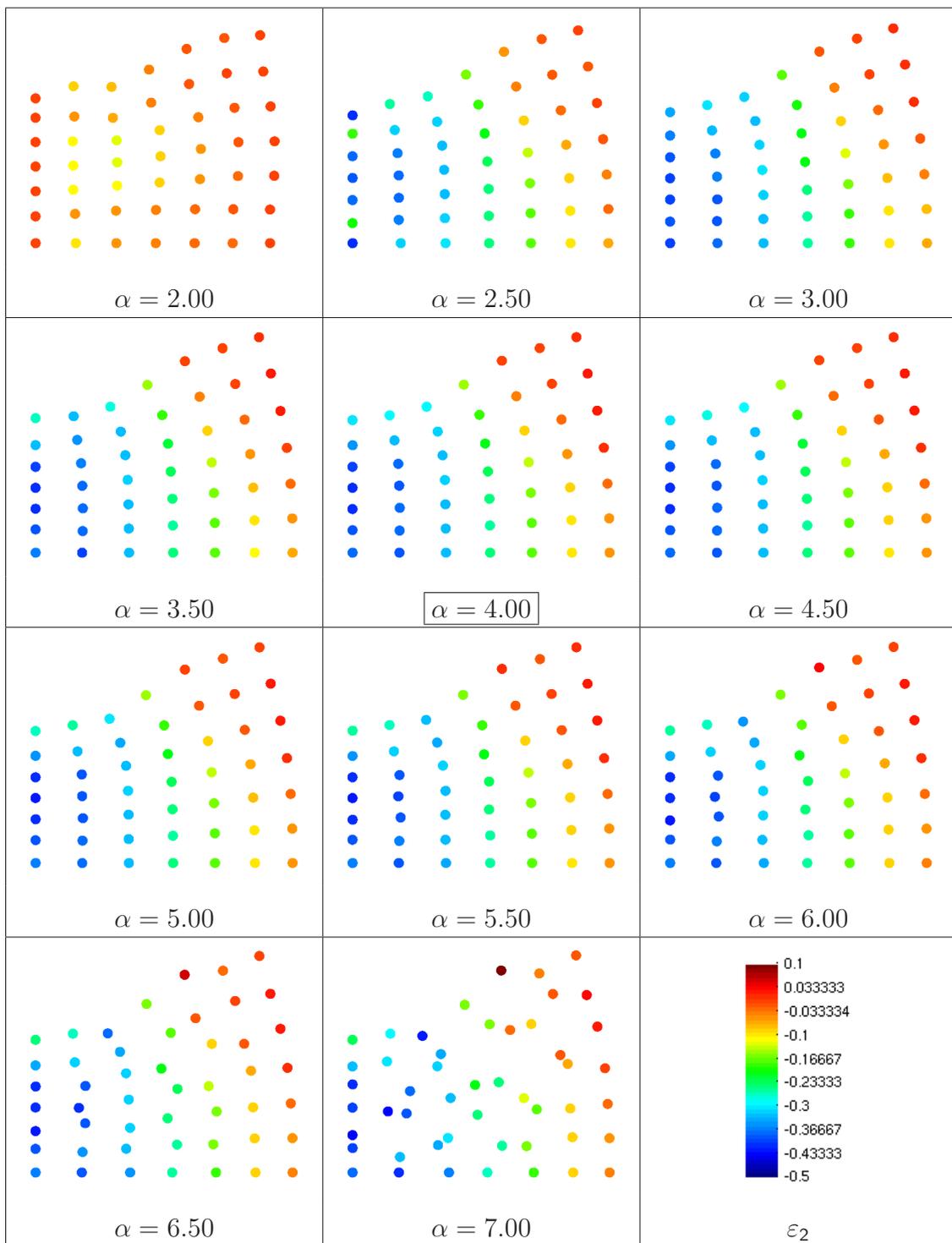


Abbildung 9.6: Beispiel Quadratscheibe unter Teillast, verformte Netze und Dehnungen ε_2 , berechnet mit FPM, Ansatzfunktionsordnung $N_G = 2$, Variation des Wichtungsfunktionsüberlappungsparameters α

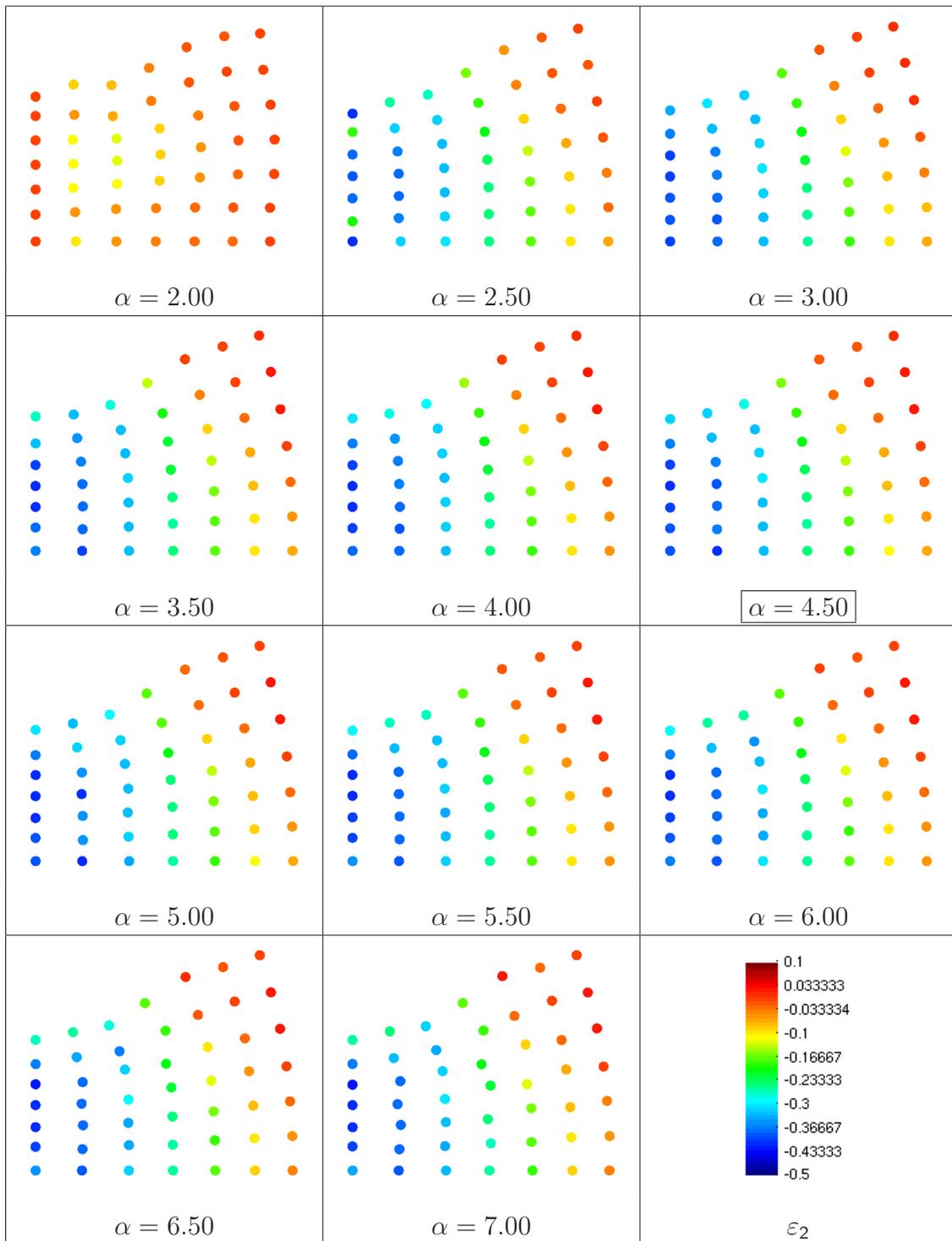


Abbildung 9.7: Beispiel Quadratscheibe unter Teillast, verformte Netze und Dehnungen ε_2 , berechnet mit FPM, Ansatzfunktionsordnung $N_G = 3$, Variation des Wichtungsfunktionsüberlappungsparameters α

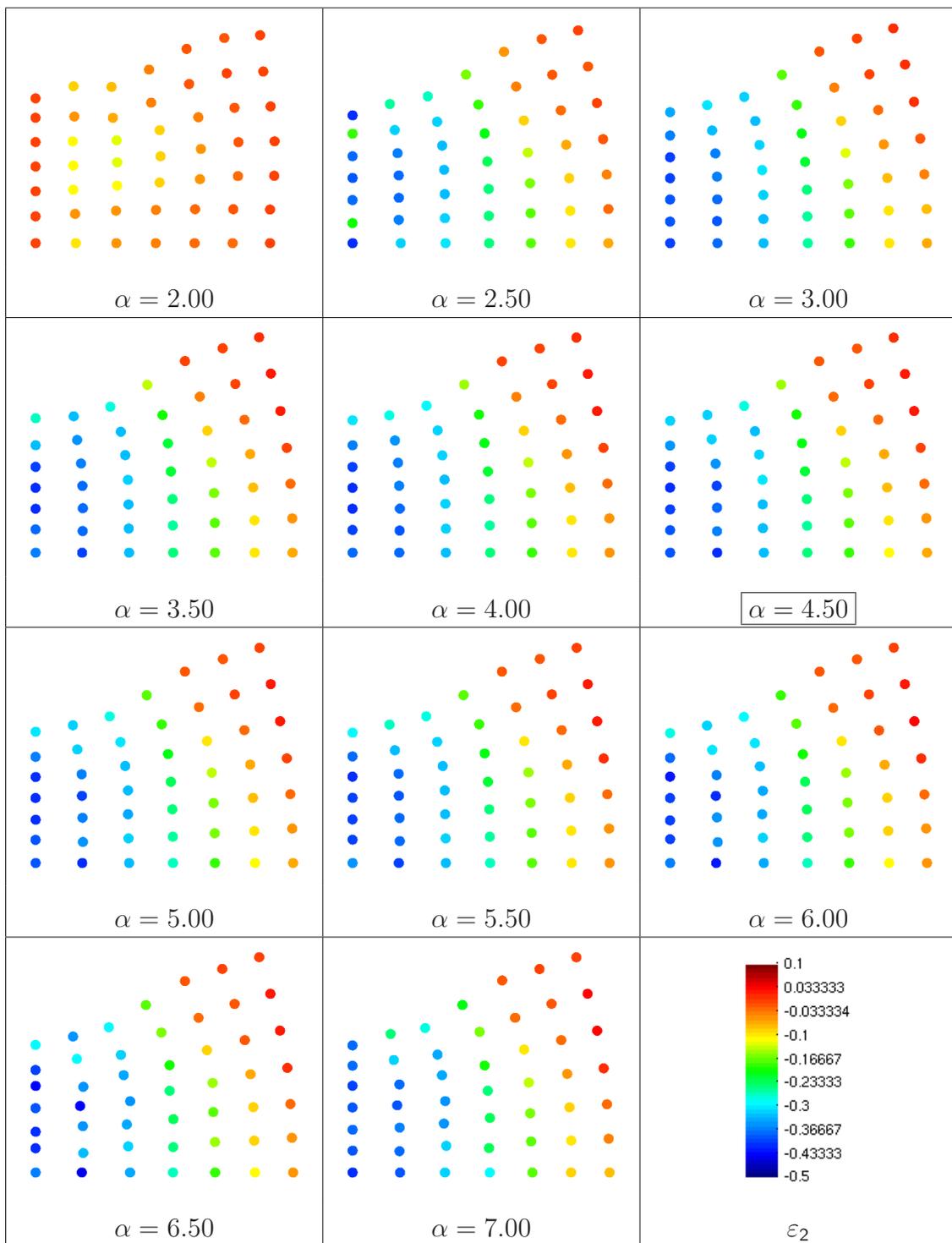


Abbildung 9.8: Beispiel Quadratscheibe unter Teillast, verformte Netze und Dehnungen ε_2 , berechnet mit FPM, Ansatzfunktionsordnung $N_G = 4$, Variation des Wichtungsfunktionsüberlappungsparameters α

9.3.2 Konvergenzverhalten FPM / FEM

Zum Vergleich der Konvergenzeigenschaften der Finite-Punkte-Methode mit denen der Finite-Elemente-Methode wird die Quadratscheibe unter Teillast mit beiden Methoden und unterschiedlich feinen Diskretisierungen berechnet (siehe Tabelle 9.3). Dabei werden außerdem unterschiedliche Ansatzfunktionsordnungen verwendet: Die Finite-Elemente-Berechnungen erfolgen mit 4-Knoten-, 9-Knoten- oder 16-Knoten-Viereckselementen; die Finite-Punkte-Methode nutzt für die MLS-Approximation Funktionsbasen mit bis zu konstanten, linearen, quadratischen oder kubischen Monomen.

Unterteilung je Richtung N_h	Charakterist. Länge [m] $h = a/N_h$	Unterteilungs- stufe $\log_2(h_0/h)$	Knoten- anzahl $N_K = (N_h + 1)^2$	Elementanzahl		
				lin. N_E	quad. N_E	kub. N_E
6	0.33333 = h_0	0	49	36	9	4
12	0.16667	1	169	144	36	16
24	0.041667	2	625	576	144	64

Tabelle 9.3: Beispiel Quadratscheibe unter Teillast, Diskretisierungen

Die Ergebnisse zeigt Abbildung 9.9. Das Konvergenzverhalten der Finite-Punkte-Methode mit der MLS-Ansatzfunktionsordnung 0 ist so schlecht, dass die Ergebnisse größtenteils unterhalb der Darstellungsbereiche liegen. Diese Methode ist damit für praktische Berechnungen unbrauchbar.

Bei nahezu inkompressiblem Materialverhalten ($\nu = 0.495$ oder $\nu = 0.499$) zeigt die Finite-Punkte-Methode mit linearen Ansatzfunktionen eine schnellere Konvergenz als die Finite-Elemente-Methode mit bilinearen 4-Knotenelementen (siehe Abbildung 9.9).

Die besseren Konvergenzeigenschaften der FPM werden mit einem deutlich höheren Rechenzeitbedarf erkauft wie die Zeitmessungen zeigen werden.

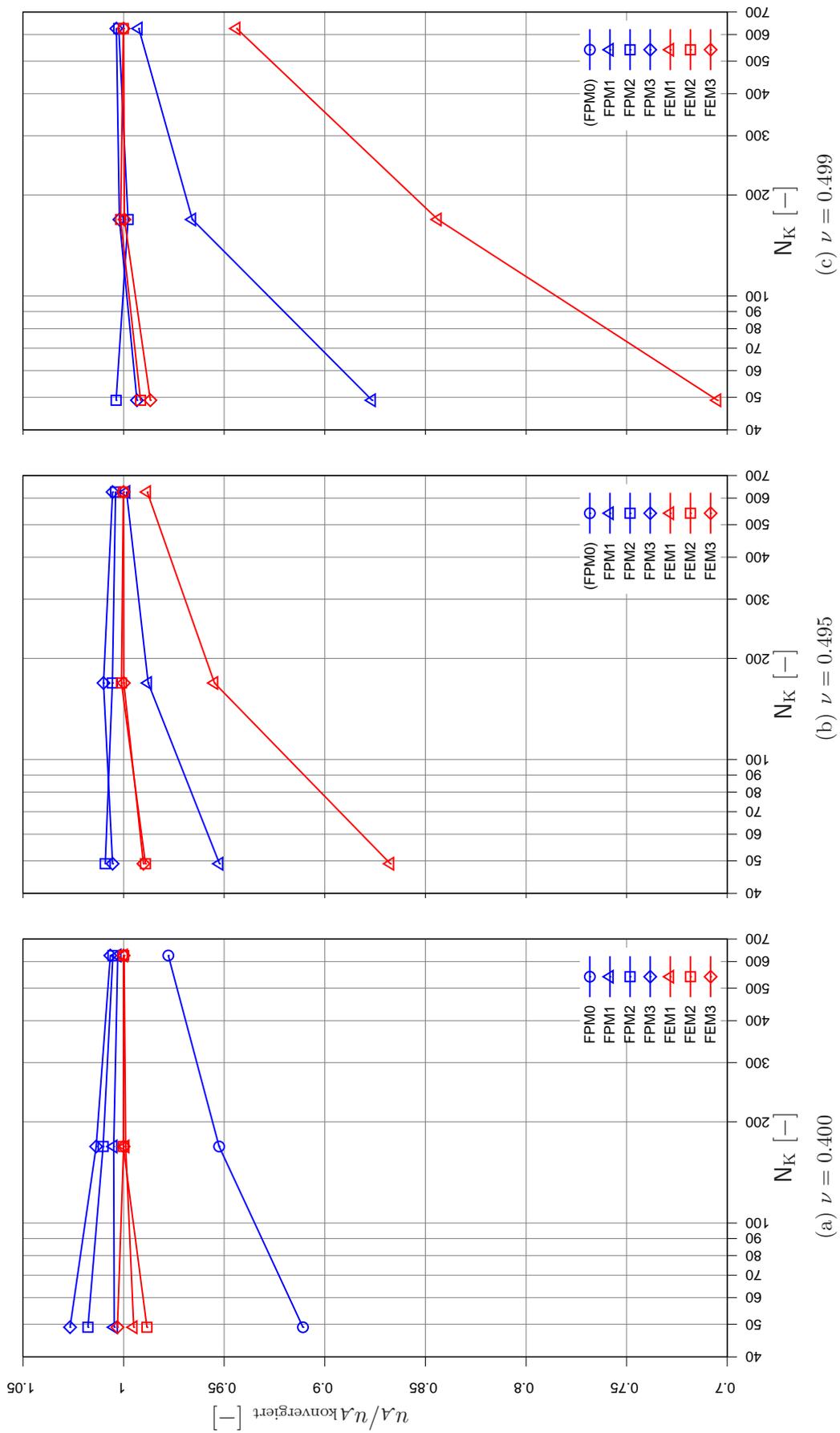


Abbildung 9.9: Beispiel Quadratscheibe unter Teillast, Konvergenzvergleich von FEM u. FPM, Ansatzfunktionsordnungen (0), 1, 2, 3

9.3.3 Zeitmessungen

Der Rechenzeitbedarf der FPM wird durch folgende Faktoren gegenüber der FEM erhöht:

- Der Aufwand zur Berechnung der MLS-Ansatzfunktionswerte ist hoch (siehe Abschnitt 5.5). Da für FPM und FEM in dieser Arbeit eine total Lagrangesche Betrachtungsweise verwendet wird, sind die Ansatzfunktionswerte nur einmal am Berechnungsbeginn für den unverformten Zustand zu ermitteln. Damit tritt der Aufwand zur Berechnung der Ansatzfunktionen dann gegenüber anderen Faktoren in den Hintergrund, wenn mehrere Zeit- und Iterationsschritte berechnet werden.
- Bei der FEM werden als Integrationspunkte die Gaußpunkte der Elemente verwendet, sodass Polynome mit für die gewählte Integrationspunktzahl maximal möglicher Genauigkeit numerisch integriert werden. Bei der elementnetzfreien FPM wird ein regelmäßiges Raster aus Integrationspunkten verwendet. Bei gleicher Integrationspunktanzahl wäre damit die Qualität der numerischen Integration gegenüber der Gaußintegration um eine Ordnung geringer. Daher muß bei der FPM die Integrationspunktanzahl gegenüber der FEM erhöht werden. In den Berechnungsbeispielen wurden beispielsweise für lineare 4-Knoten-Elemente 4 Gaußpunkte und für eine lineare MLS-Approximation auf der gleichen Fläche 9 Integrationspunkte eingesetzt.
- Im Abschnitt 9.3.1 wurde gezeigt, dass zur zuverlässigen Berechnung von MLS-Ansatzfunktionen einer gewünschten Ordnung Mindestwerte für die Wichtungsfunktionsüberlappungen eingehalten werden müssen. Damit wird die Bandbreite des globalen Gleichungssystems gegenüber der FEM erhöht.

Da es sich bei der Quadratscheibe unter Teillast um ein vollständig lineares Berechnungsbeispiel handelt (das Materialverhalten ist linear elastisch), erfolgt die Berechnung in einem Schritt.

Abbildung 9.10 zeigt in doppeltlogarithmischer Darstellung die Rechenzeiten von FEM und FPM unterschiedlicher Ansatzfunktionsordnung über der Knotenanzahl. Die Rechenzeiten der FEM liegen noch bei doppelt so feiner Diskretisierung unter den Rechenzeiten der FPM.

In der Darstellung 9.11 (a) kann abgelsen werden, wie sich die Ansatzfunktionsordnung auf die Rechenzeit auswirkt. Hierzu werden die Rechenzeiten der FPM und

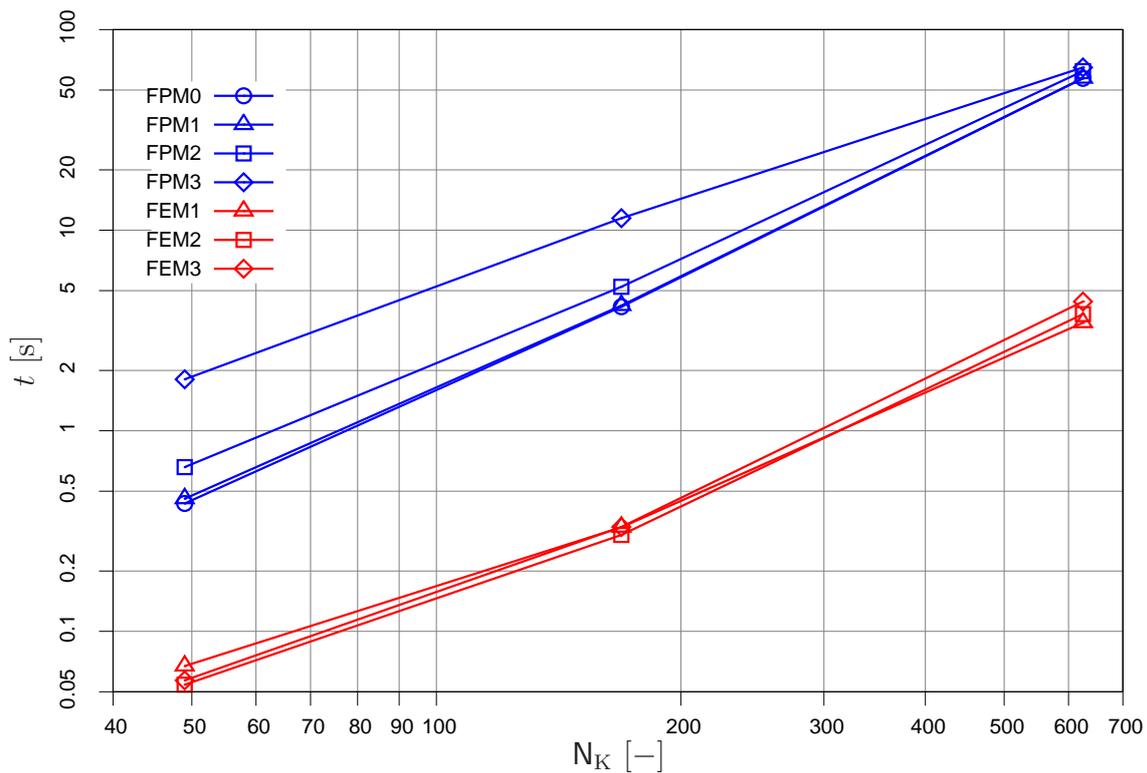
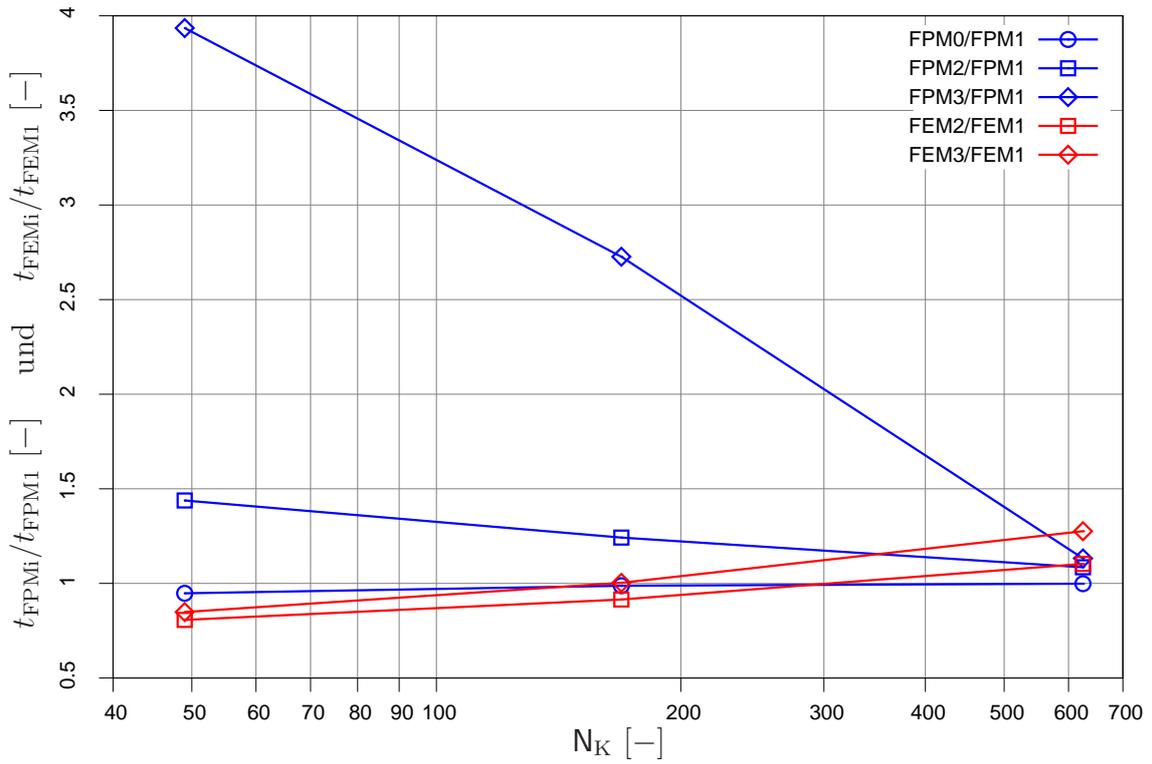


Abbildung 9.10: Beispiel Quadratscheibe, Zeitmessungen, Berechnungen mit FPM und FEM, Variation der räumlichen Diskretisierung

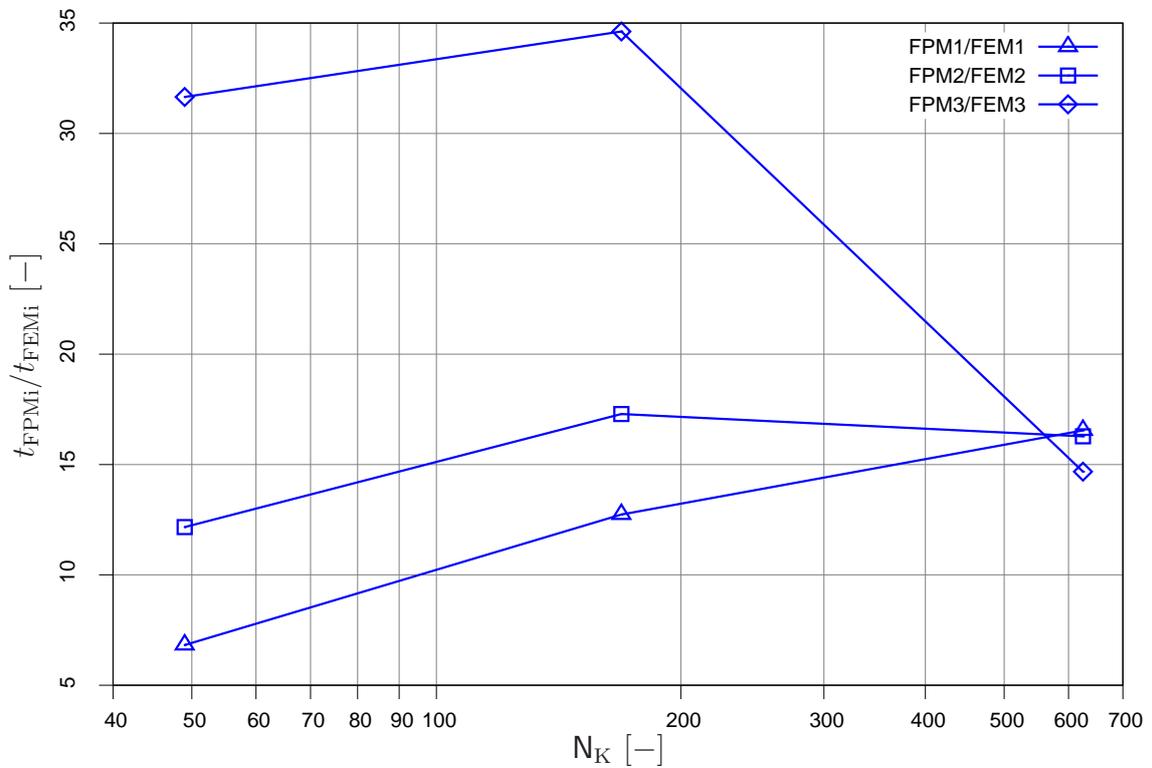
der FEM mit unterschiedlicher Ansatzfunktionsordnung jeweils auf die Rechenzeiten der linearen FPM und der linearen FEM bezogen. Bei der FPM bestehen deutlich größere Rechenzeitunterschiede für unterschiedliche Ansatzfunktionsordnungen. Bei größerer Knotenanzahl werden die Rechenzeitunterschiede geringer, da die Zeit zur Lösung des globalen Gleichungssystems gegenüber der Zeit zur Berechnung der Ansatzfunktionswerte dominiert. Bei der FEM wird bei geringer Knotenanzahl sogar Rechenzeit mit einer höheren Ansatzfunktionsordnung gespart, da für die Ansatzfunktionen von 4 kubische Elementen die Ansatzfunktionen von 9 quadratischen Elementen oder 36 linearen Elementen berechnet werden müssen. Bei einer höheren Knotenanzahl bewirkt die größere Bandbreite des globalen Gleichungssystems bei Verwendung von Elementen höherer Ordnung einen Anstieg der Rechenzeit gegenüber linearen Elementen.

In der Darstellung 9.11 (b) sind die Rechenzeiten der FPM bezogen auf die Rechenzeiten der FEM mit gleicher Ansatzfunktionsordnung. Bei niedriger Knotenanzahl dominiert der Aufwand zur Berechnung der Ansatzfunktionen, bei höherer Knotenanzahl die Einfluß der Bandbreite des globalen Gleichungssystems. Im Mittel ist die Rechenzeit der FPM um den Faktor 16 höher als die Rechenzeit der FEM.

Abschließend wird in Abbildung 9.12 der Einfluss der Wichtungsfunktionsgröße in der FPM auf die Rechenzeit untersucht. Es wird eine lineare MLS-Approximation verwendet. Der Wichtungsfunktionsüberlappungsparameter und die Knotenanzahl werden variiert. Abbildung 9.12 (a) zeigt die absoluten Rechenzeiten und 9.12 (b) die auf die Rechenzeit bei empfohlener Wichtungsfunktionsüberlappung bezogenen Rechenzeiten. Mit ca 10% Rechenzeitunterschied bei extremer Wahl des Wichtungsfunktionsüberlappungsparameters ist der Einfluss gering.

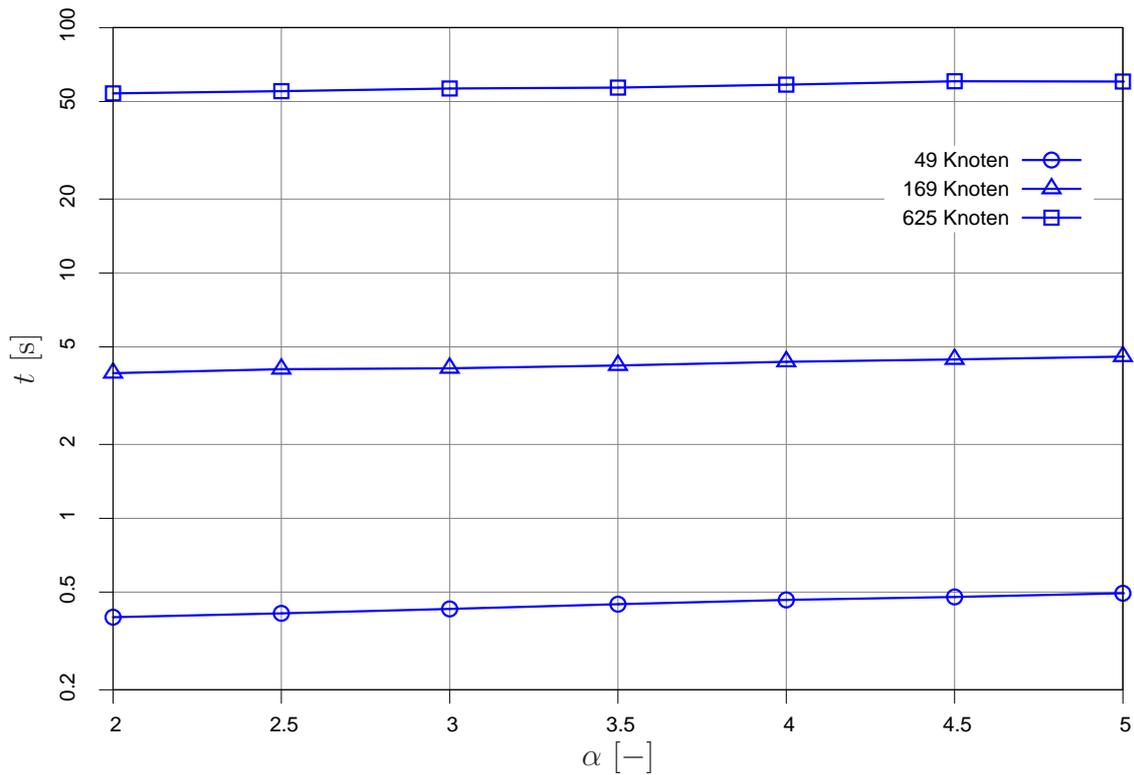


(a) Rechenzeitverhältnis über Knotenanzahl

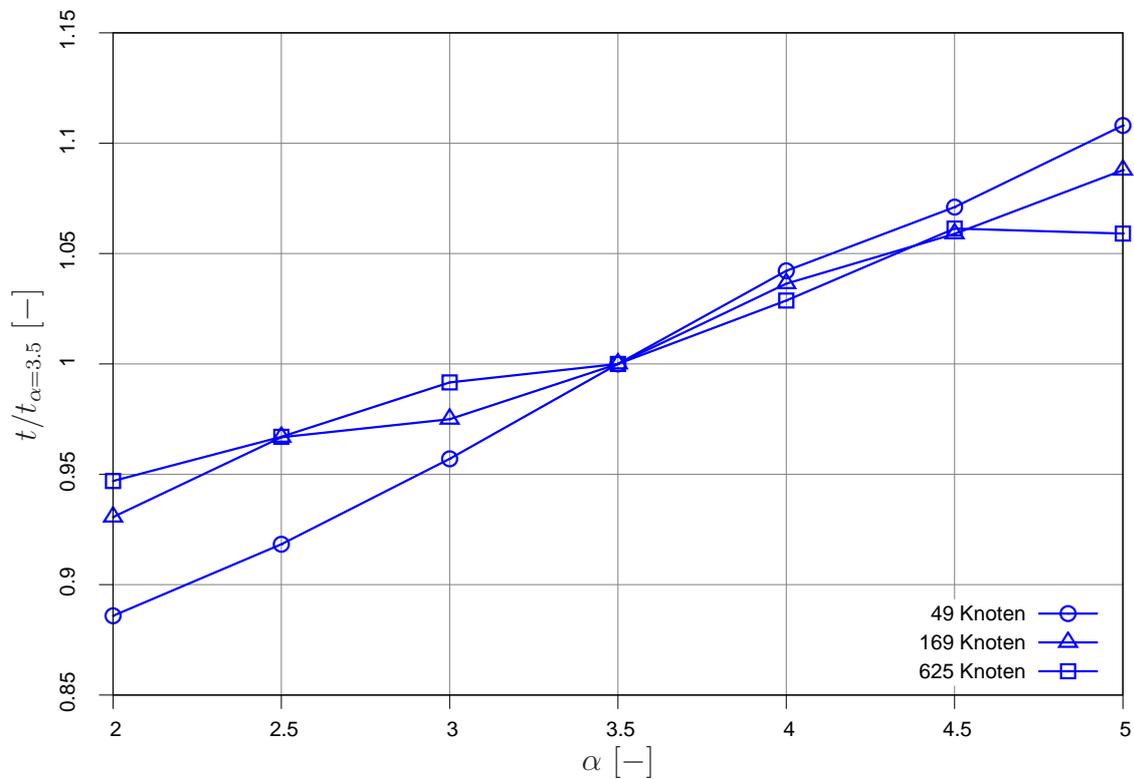


(b) Rechenzeitverhältnis über Knotenanzahl

Abbildung 9.11: Beispiel Quadratscheibe, Zeitmessungen, Berechnungen mit FPM und FEM, Variation der räumlichen Diskretisierung



(a) Rechenzeit über Wichtungsfunktionsüberlappungsparameter



(b) Rechenzeitverhältnis über Wichtungsfunktionsüberlappungsparameter

Abbildung 9.12: Beispiel Quadratscheibe, Zeitmessungen, Berechnungen mit FPM 1. Ordnung, Variation des Wichtungsfunktionsüberlappungsparameters und der räumlichen Diskretisierung

9.4 Extrusion

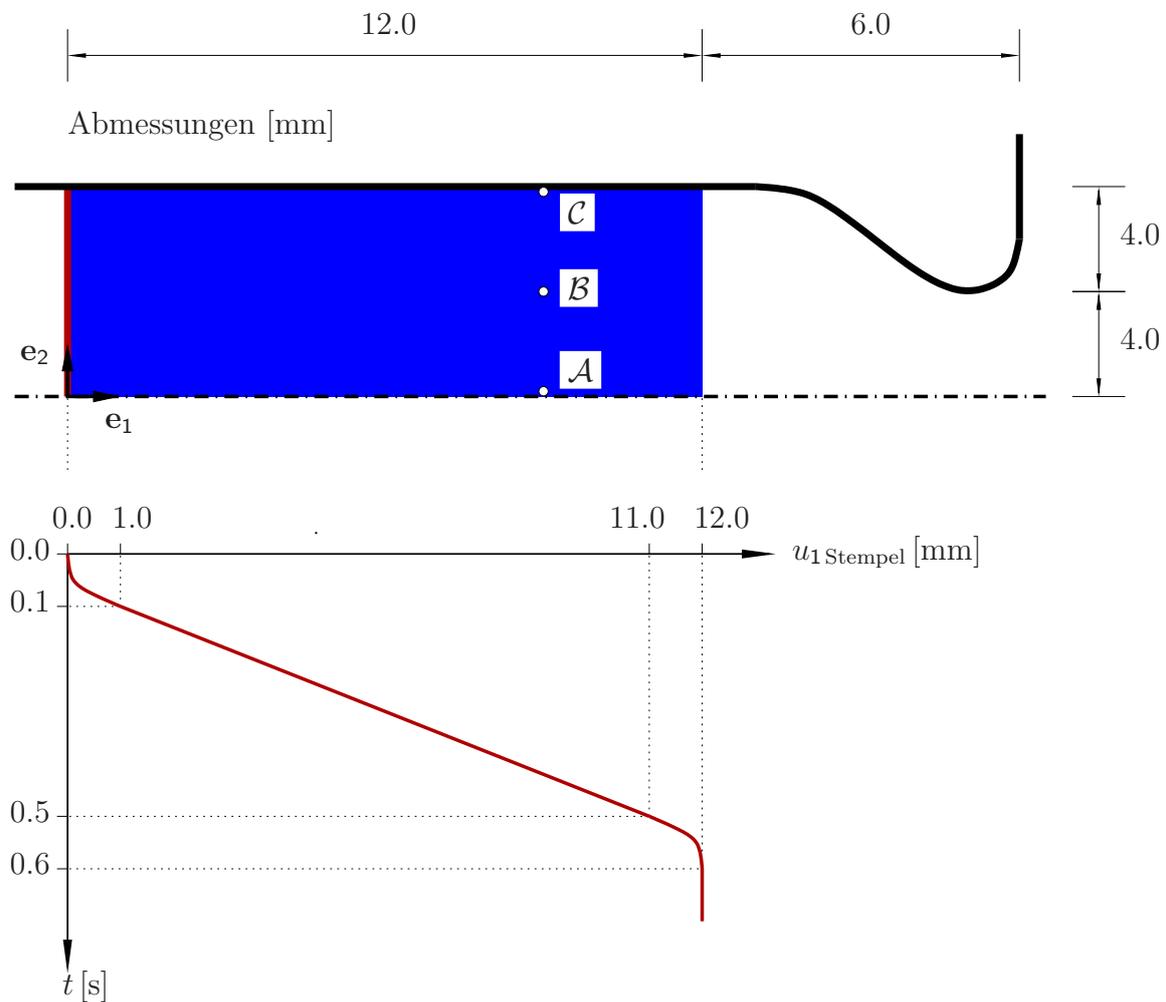


Abbildung 9.13: Beispiel Extrusion, System

Ein Block aus Aluminium (Dichte $\rho = 2700 \text{ kg/m}^3$, Elastizitätsmodul $E = 72000 \cdot 10^6 \text{ N/m}^2$, Querkontraktionszahl $\nu = 0.33$, Anfangsfließspannung $\sigma_{y_0} = 150 \cdot 10^6 \text{ N/m}^2$ und isotroper Verfestigungsparameter $H_{\text{iso}} = 200 \cdot 10^6 \text{ N/m}^2$) wird mit einem Stempel durch eine Düse gedrückt.

Das Materialverhalten des Aluminiums wird mit dem verwendeten elasto-plastischen Materialmodell mit isotroper Verfestigung (siehe Abschnitt 4.3.3) nur angenähert, da Aluminium und Aluminiumlegierungen keine ausgeprägte Fließgrenze besitzen.

Stempel und Wandung werden starr und mit C^1 -stetigen Kontaktelementen diskretisiert. Die Verschiebung des Stempels wird vorgegeben.

Ein Vergleich der Bilder der deformierten Netze (Abbildungen 9.14 und 9.15) zeigt eine gute Übereinstimmung zwischen den Ergebnissen der Finite-Punkte-Methode und der Finite-Elemente-Methode.

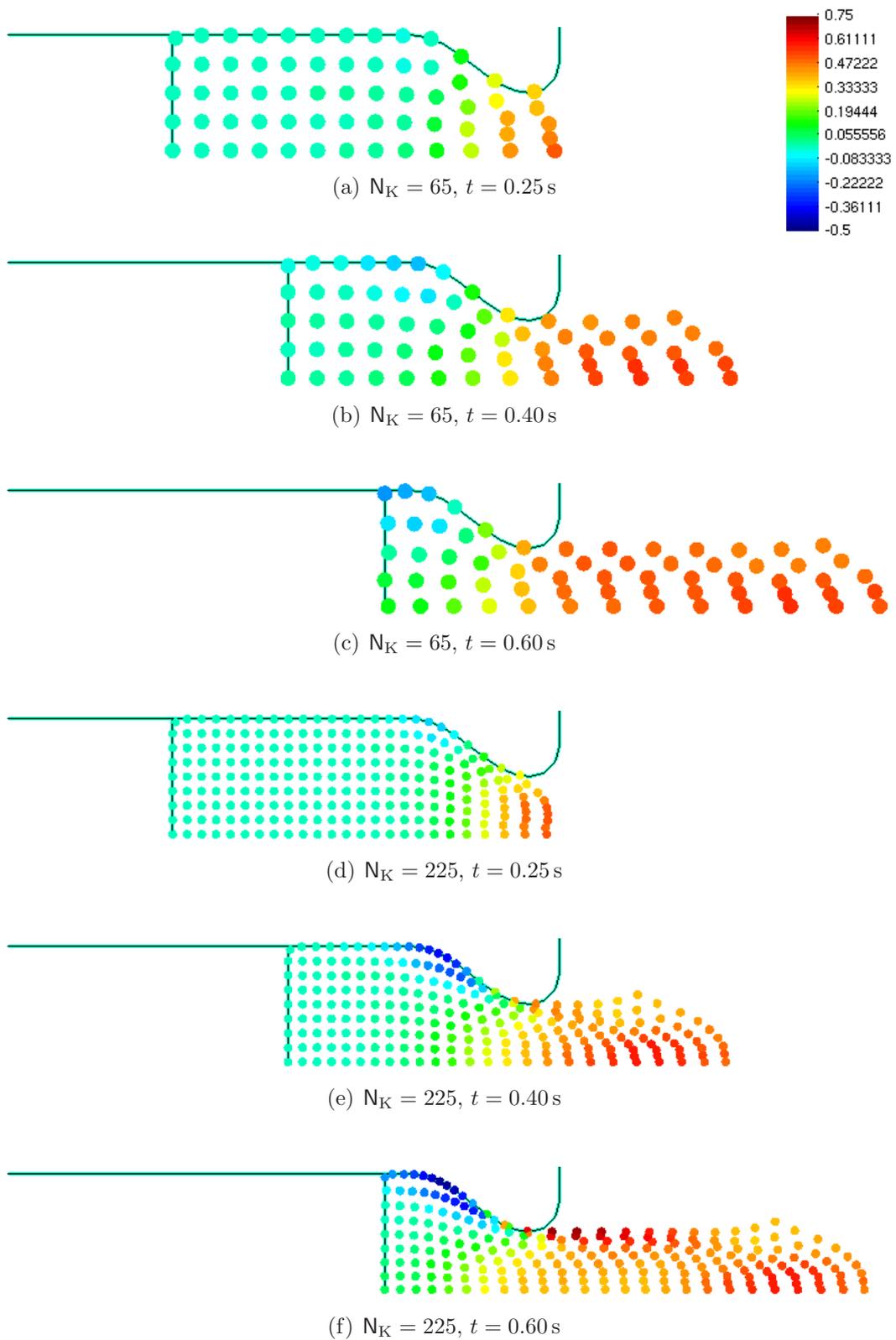


Abbildung 9.14: Beispiel Extrusion, verformte Netze und Dehnung ε_1 , berechnet mit FPM

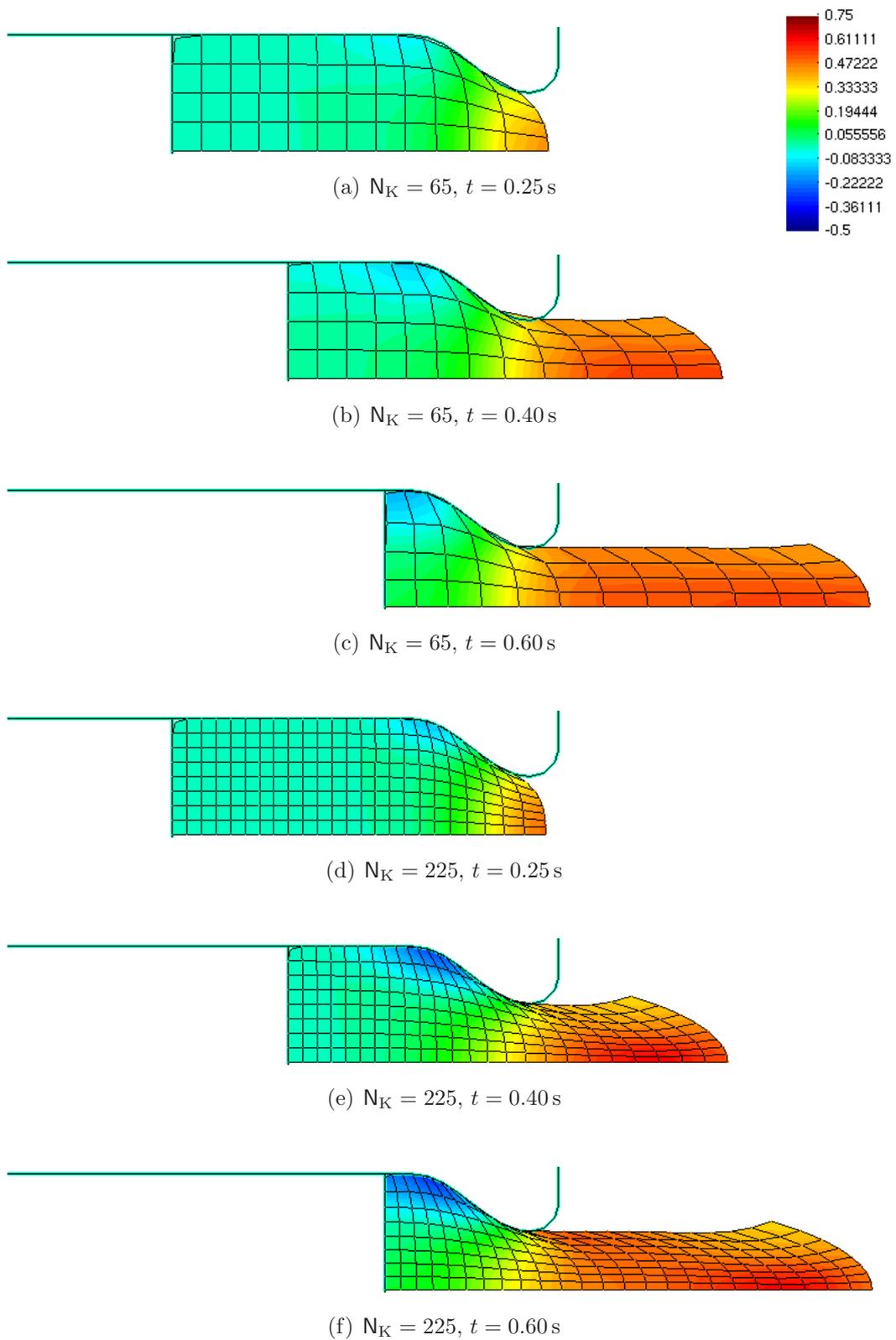


Abbildung 9.15: Beispiel Extrusion, verformte Netze und Dehnung ε_1 , berechnet mit FEM

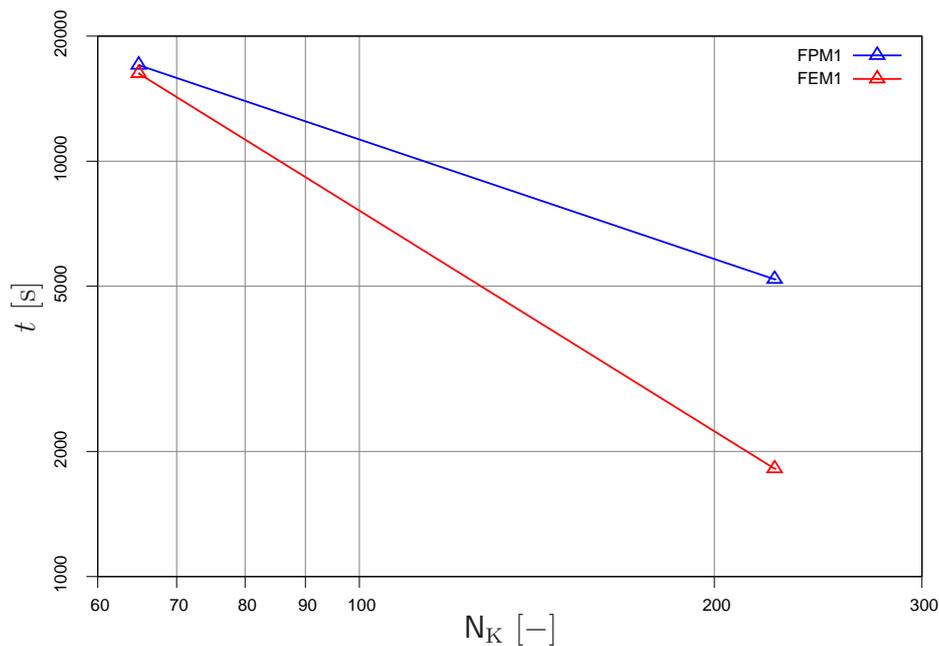


Abbildung 9.16: Beispiel Extrusion, Zeitmessungen, Berechnungen mit FPM und FEM, Variation der räumlichen Diskretisierung

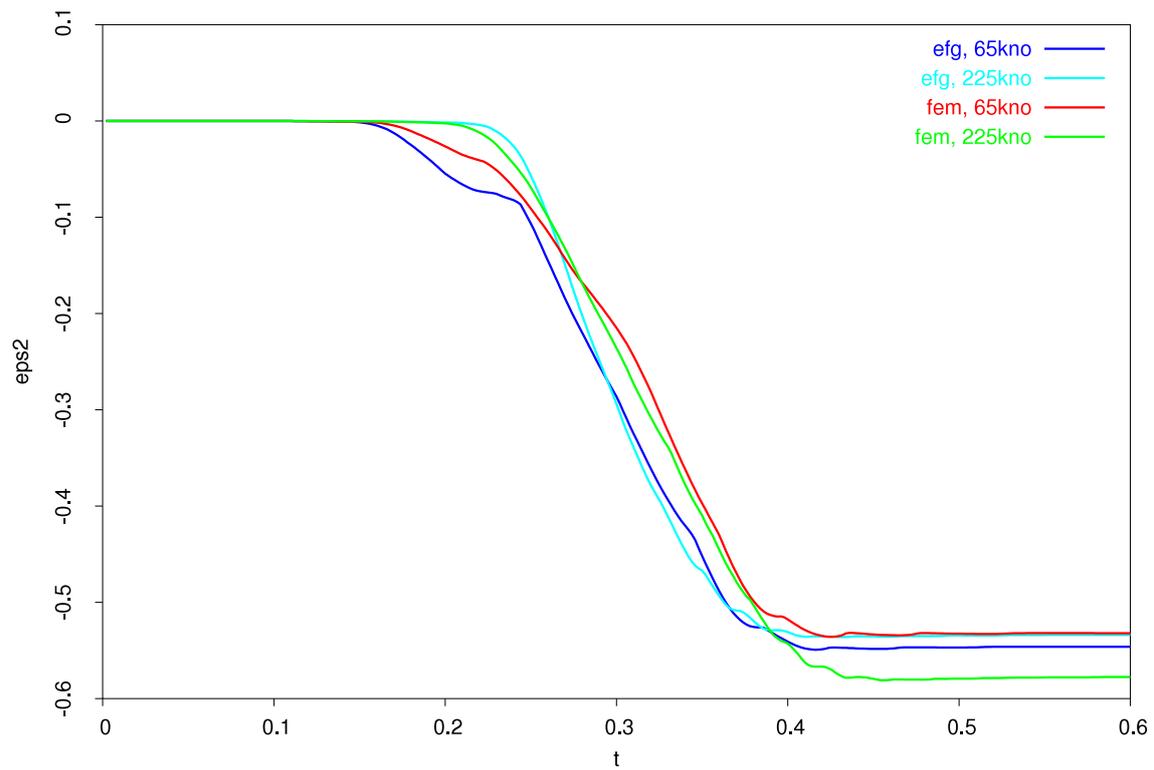
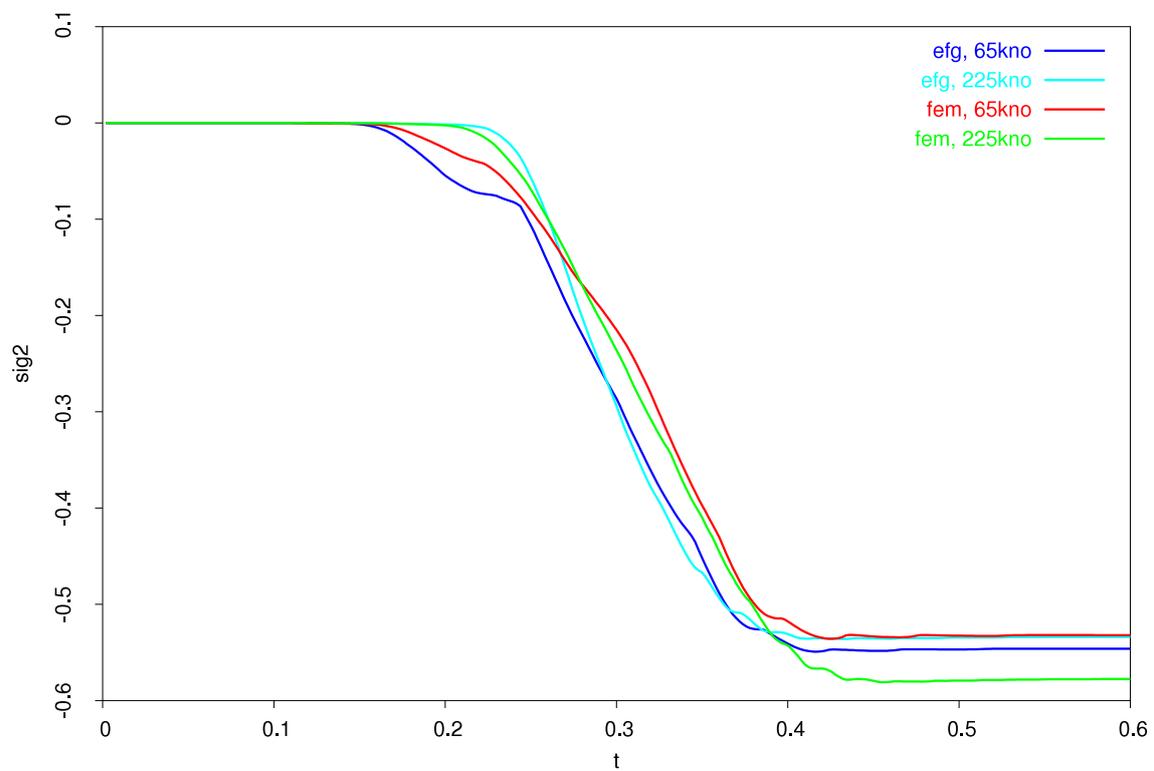
Die Knoten der FPM-Berechnung werden durch Kreisscheiben visualisiert. Hierdurch entsteht der fälschliche Eindruck, die Knoten würden die Kontaktlinie überschreiten. Tatsächlich ist die Penetration, die sich bei dem gewählten Strafparameter $\alpha = 1 \cdot 10^{-6}$ ergibt, nur sehr gering.

Da ein Knoten-zu-Segment Kontaktalgorithmus eingesetzt wird, schneiden die Ränder der linearen finiten Elemente die gekrümmte Kontaktlinie in Form von Sehnen zwischen den Kontaktpunkten.

Da die Kontaktlinien C^1 -stetig diskretisiert sind, sind auch die zeitlicher Zustandsgrößenverläufe der Punkte \mathcal{A} und \mathcal{C} (Abbildungen 9.17 bis 9.19) glatt.

Der Zeitbedarf für die Berechnung der Ansatzfunktionen tritt bei diesem wie auch bei den folgenden Berechnungsbeispielen in den Hintergrund, da eine hohe Anzahl an Zeit- und Iterationsschritten benötigt wird. (Der FPM und FEM wurde eine total Lagrangesche Betrachtungsweise zugrunde gelegt. Die Ansatzfunktionen werden also nur einmal für die Ausgangskonfiguration berechnet.)

Als Unterschiede zwischen FPM und FEM, die einen Einfluss auf den Zeitbedarf der Rechnungen haben, verbleiben die Art der Ansatzfunktionen, die Integrationspunktanzahl und die Bandbreite des globalen Gleichungssystems. Bei der FPM reduzieren die „besseren“ MLS-Ansatzfunktionen die Rechenzeit, da gegenüber der FEM weniger Iterationsschritte benötigt werden, und erhöhen die größere Integrationspunktanzahl und höhere Bandbreite des globalen Gleichungssystems die Rechenzeit.

(a) Verzerrung $\epsilon_{2,\mathcal{A}}(t)$ (b) Spannung $\sigma_{2,\mathcal{A}}(t)$ Abbildung 9.17: Beispiel Extrusion, Verzerrung und Spannung des Punktes \mathcal{A} in 2-Richtung, berechnet mit FPM und FEM, Variation der räumlichen Diskretisierung

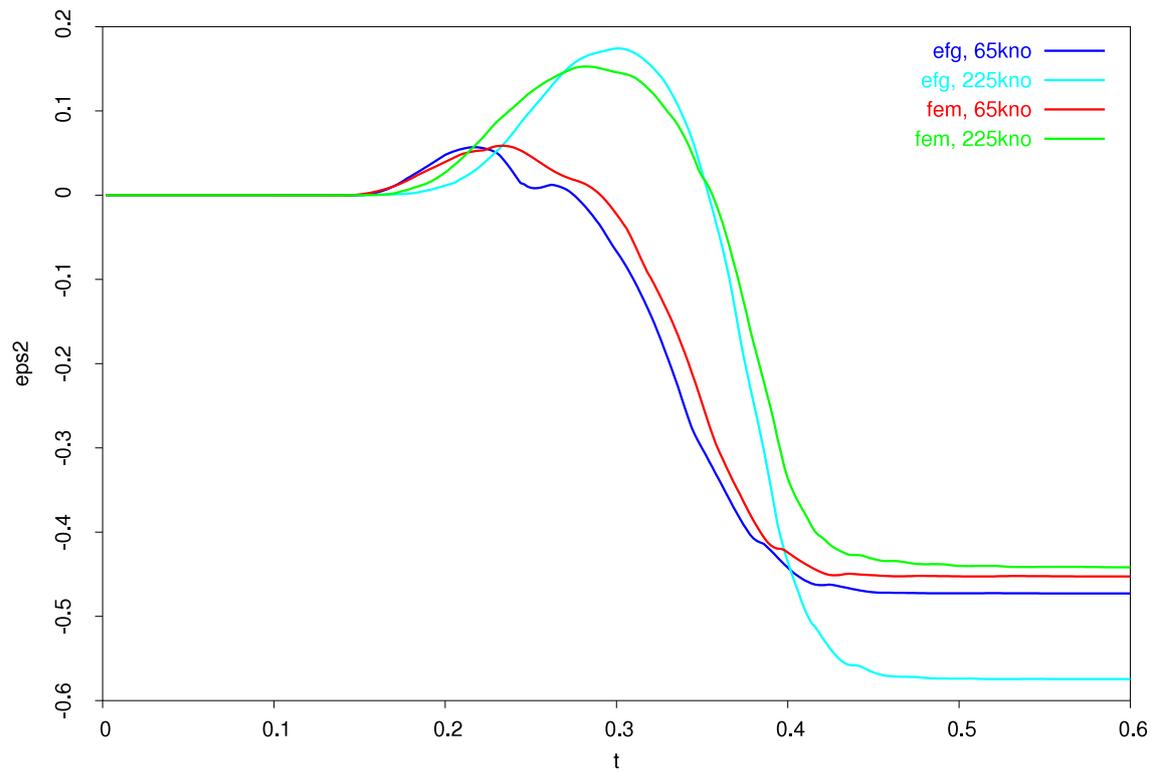
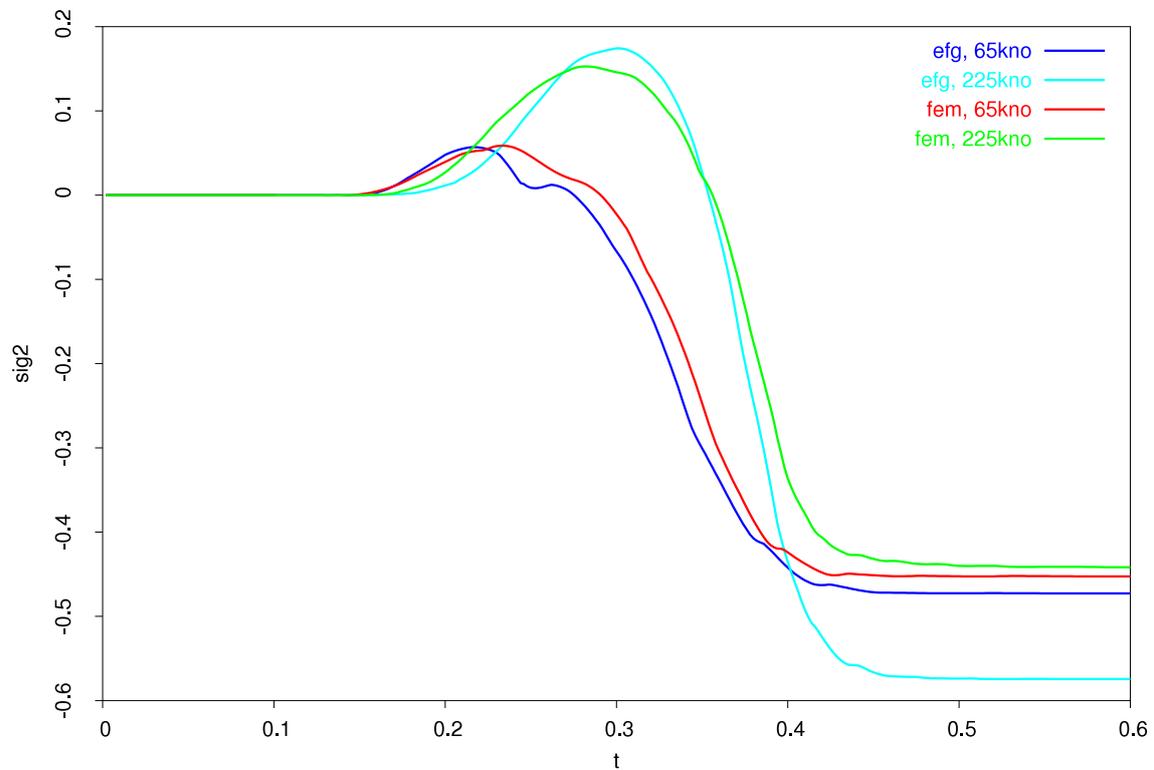
(a) Verzerrung $\varepsilon_{2c}(t)$ (b) Spannung $\sigma_{2c}(t)$

Abbildung 9.18: Beispiel Extrusion, Verzerrung und Spannung des Punktes \mathcal{C} in 2-Richtung, berechnet mit FPM und FEM, Variation der räumlichen Diskretisierung

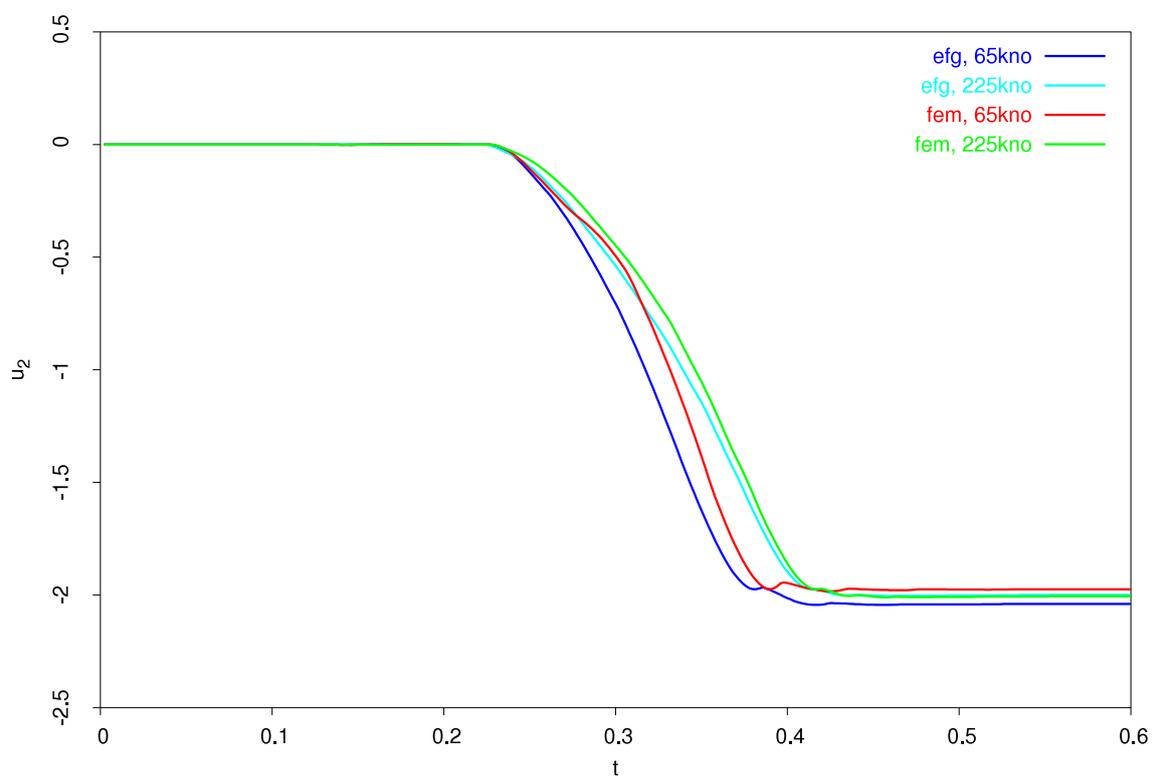


Abbildung 9.19: Beispiel Extrusion, Verschiebung des Punktes \mathcal{C} in 2-Richtung, berechnet mit FPM und FEM, Variation der räumlichen Diskretisierung

9.5 Tiefziehen

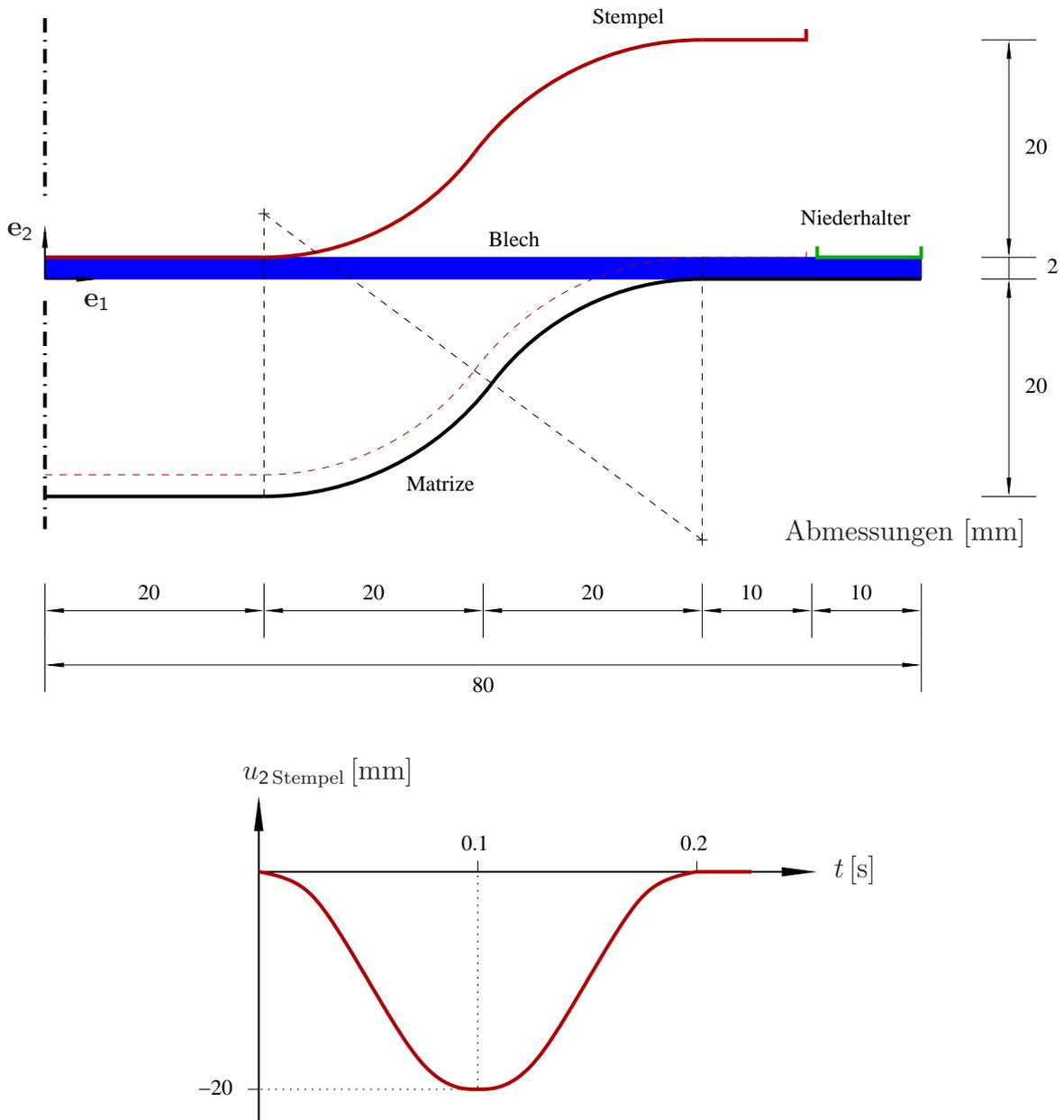


Abbildung 9.20: Beispiel Tiefziehen, System

Ein Aluminiumblech (Materialeigenschaften wie beim vorangegangenen Extrusionsbeispiel) wird tiefgezogen. Matrize und Stempel sind so geformt, dass bei niedrigster Stempelposition der Abstand zwischen beiden an allen Stellen genau der Blechdicke entspricht. Die Krümmungsradien bezogen auf die Mittellinie des Bleches betragen 25 mm. Ein Niederhalter verhindert am Rand Vertikalverschiebungen des Bleches.

Alle Werkzeuge sind starr und mit C^1 -stetigen Kontaktelementen diskretisiert. Die Verschiebungen des Stempels und des Niederhalters werden vorgegeben.

Bei diesem Beispiel zeigen sich die Beschränkungen der Theorie kleiner Verformungen: Die Verformung des Bleches in vertikaler Richtung führt zu keinem Einzug des Bleches am Rand und die Rotation des Bleches im mittleren Bereich führt zu einer unphysikalischen Blechdickenänderung. Letzterer Effekt ist darauf zurückzuführen, dass der Deformator nicht invariant gegenüber Starrkörperdrehungen ist (siehe Abschnitt 4.2). Die durch Schrägstellung unphysikalisch verdickten Bereiche des Bleches verformen sich beim Schließen des Stempels plastisch. Tatsächlich dominieren bei Tiefziehvorgängen plastische Verformungen infolge Biegung und Zug.

Auch wenn die Ergebnisse der Simulation durch die unzureichende kinematische Beschreibung des Kontinuums unphysikalisch sind, ist das Beispiel gut geeignet, um den Kontaktalgorithmus zu überprüfen, da sich die Kontaktsituation zwischen dem Werkstück und den Werkzeugen während des Umformvorganges ständig verändert.

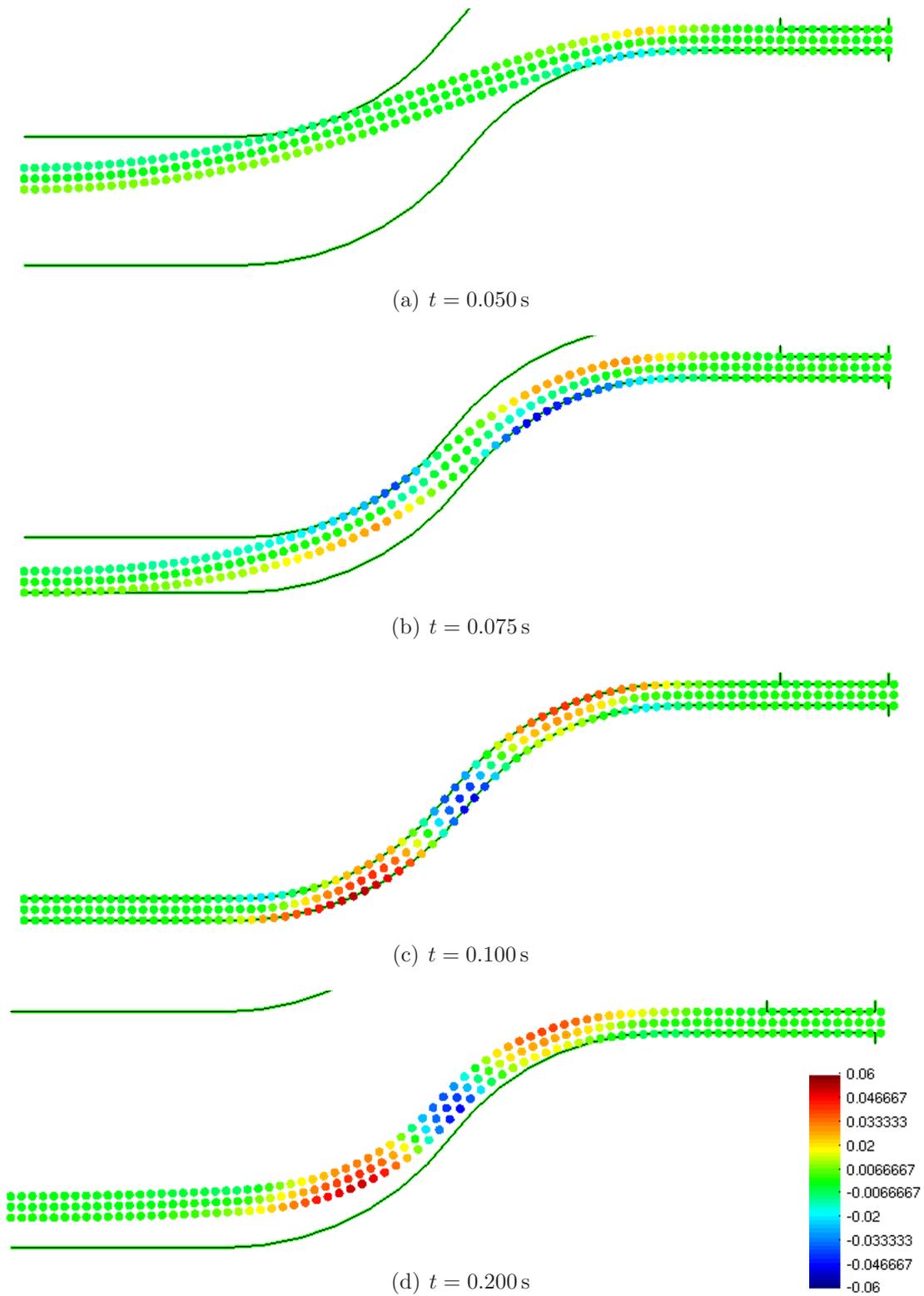


Abbildung 9.21: Beispiel Tiefziehen, verformte Netze und Dehnung ε_1 , berechnet mit FPM, Knotenanzahl $N_K = 243$

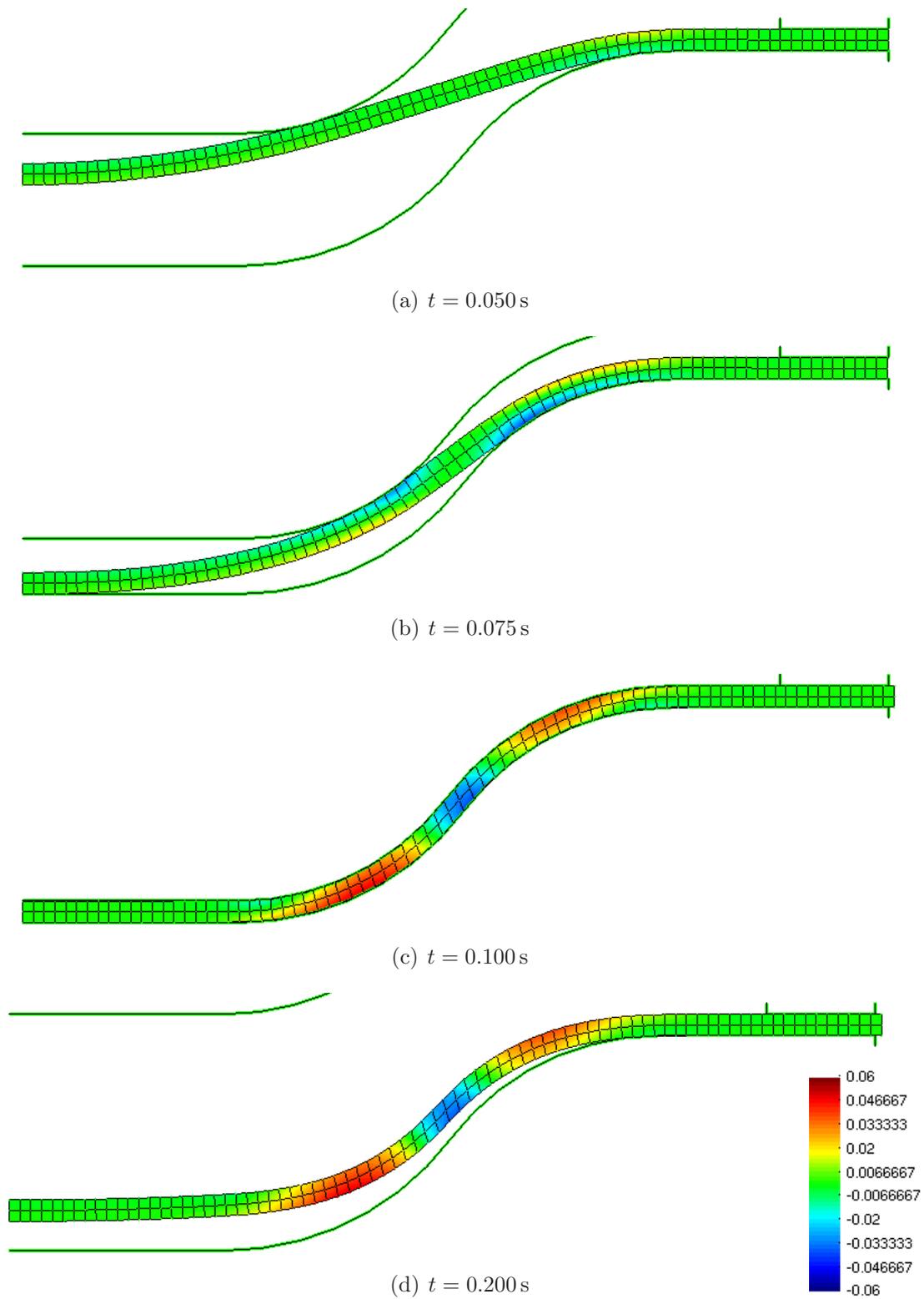


Abbildung 9.22: Beispiel Tiefziehen, verformte Netze und Dehnung ε_1 , berechnet mit FEM, Knotenanzahl $N_K = 243$

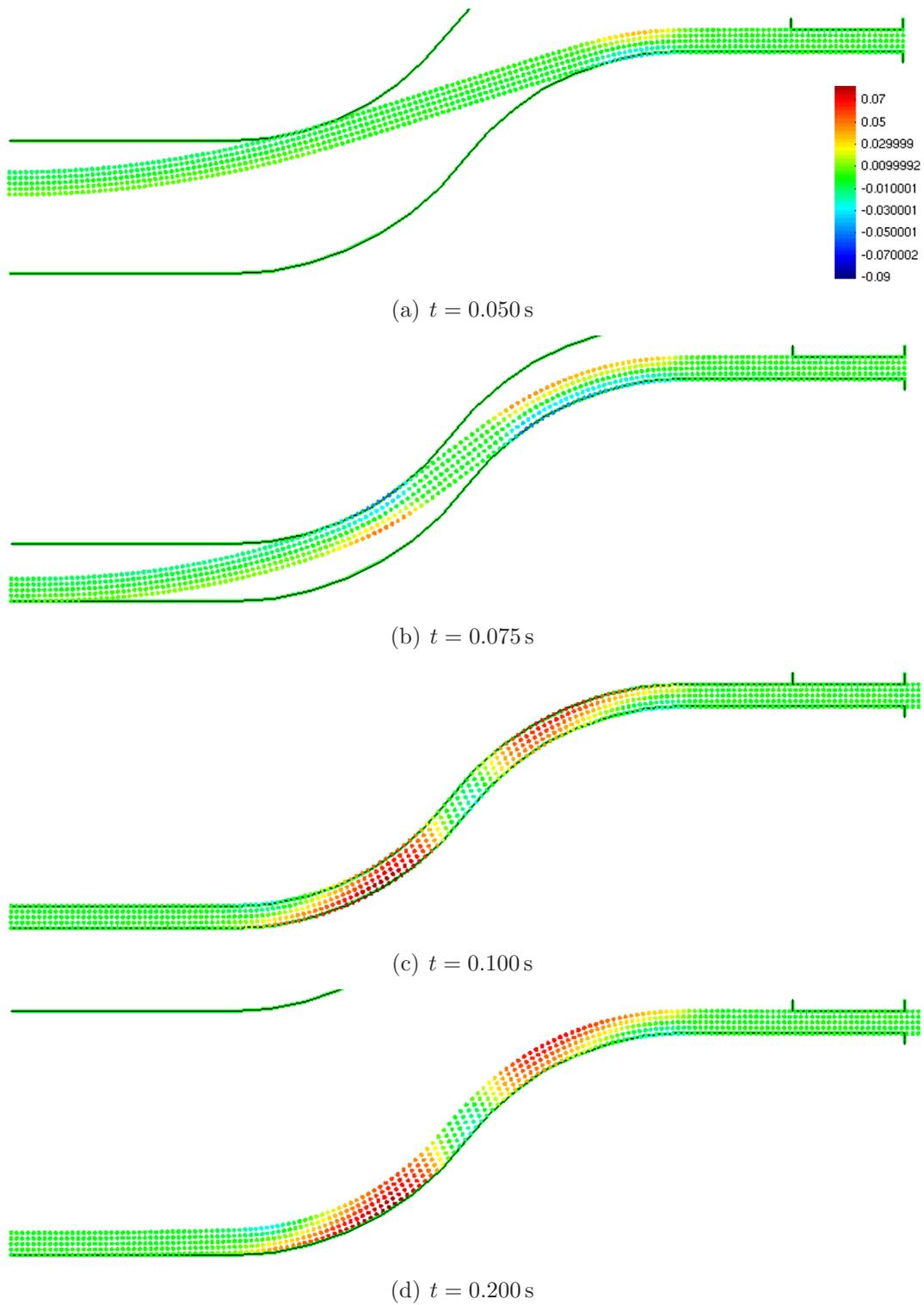


Abbildung 9.23: Beispiel Tiefziehen, verformte Netze und Dehnung ε_1 , berechnet mit FPM, Knotenanzahl $N_K = 805$

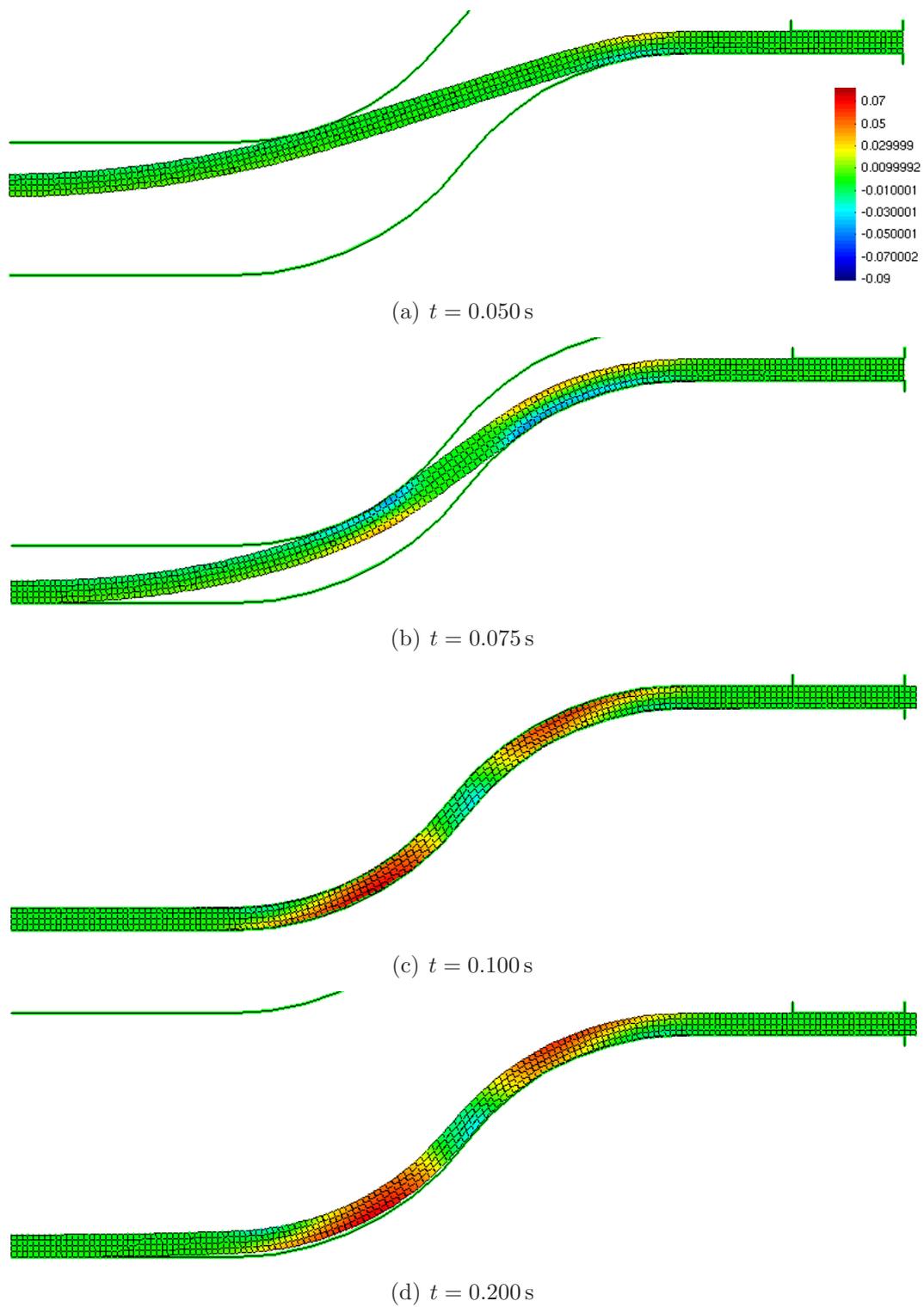


Abbildung 9.24: Beispiel Tiefziehen, verformte Netze und Dehnung ε_1 , berechnet mit FEM, Knotenanzahl $N_K = 805$

9.6 Aufprall

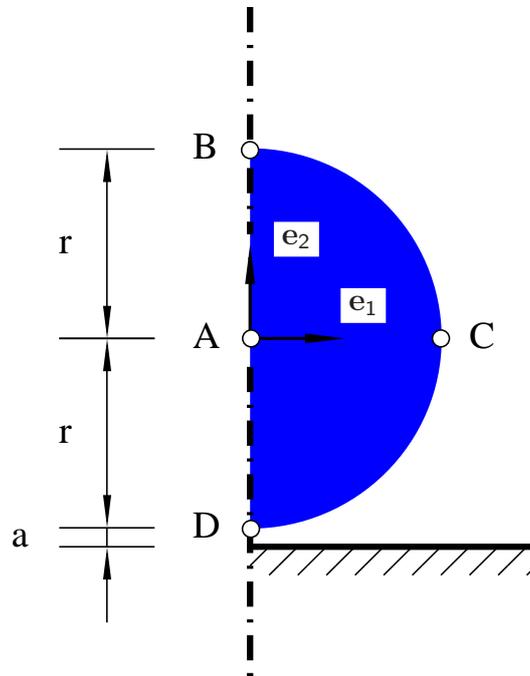


Abbildung 9.25: Beispiel Aufprall, System

Ein häufiges Einsatzgebiet elementnetzfreier Methoden ist die Untersuchung von Aufprallproblemen. Früh wurden hierfür die *SPH* (JOHNSON & BEISSEL (1996)) oder die *MPM* (SULSKY & SCHREYER (1996)) verwendet.

Eine Aluminiumscheibe (Radius 20 mm, Materialeigenschaften wie beim Extrusionsbeispiel) wird mit 300 m/s auf eine starre Barriere geschossen. Es wird ein ebener Spannungszustand angenommen. Beim Berechnungsbeginn hat der Körper einen Abstand von 2 mm von der Barriere.

Neben der räumlichen Diskretisierung wird bei diesem Beispiel auch die zeitliche Diskretisierung variiert. Die Berechnungsergebnisse für die Zeitschrittlängen $dt = 1 \cdot 10^{-7}$ s und $dt = 1 \cdot 10^{-8}$ s unterscheiden sich nur unwesentlich.

Die Zeitschrittlänge $dt = 1 \cdot 10^{-6}$ s ist zu groß gewählt. Hier zeigen sich bei den zeitlichen Verläufen der Geschwindigkeit (Abbildung 9.31 oben) und der Spannung (Abbildung 9.30 unten) unphysikalisch starke Oszillationen, die bei der FEM-Rechnung stärker ausgeprägt sind. Die FEM-Berechnung bricht für diese Zeitschrittlänge mit der feineren räumlichen Diskretisierung nach ca. $3.5 \cdot 10^{-5}$ s ab, wohingegen die FPM-Berechnung durchläuft. Mit der größeren räumlichen Diskretisierung konvergiert die FEM schlechter als die FPM.

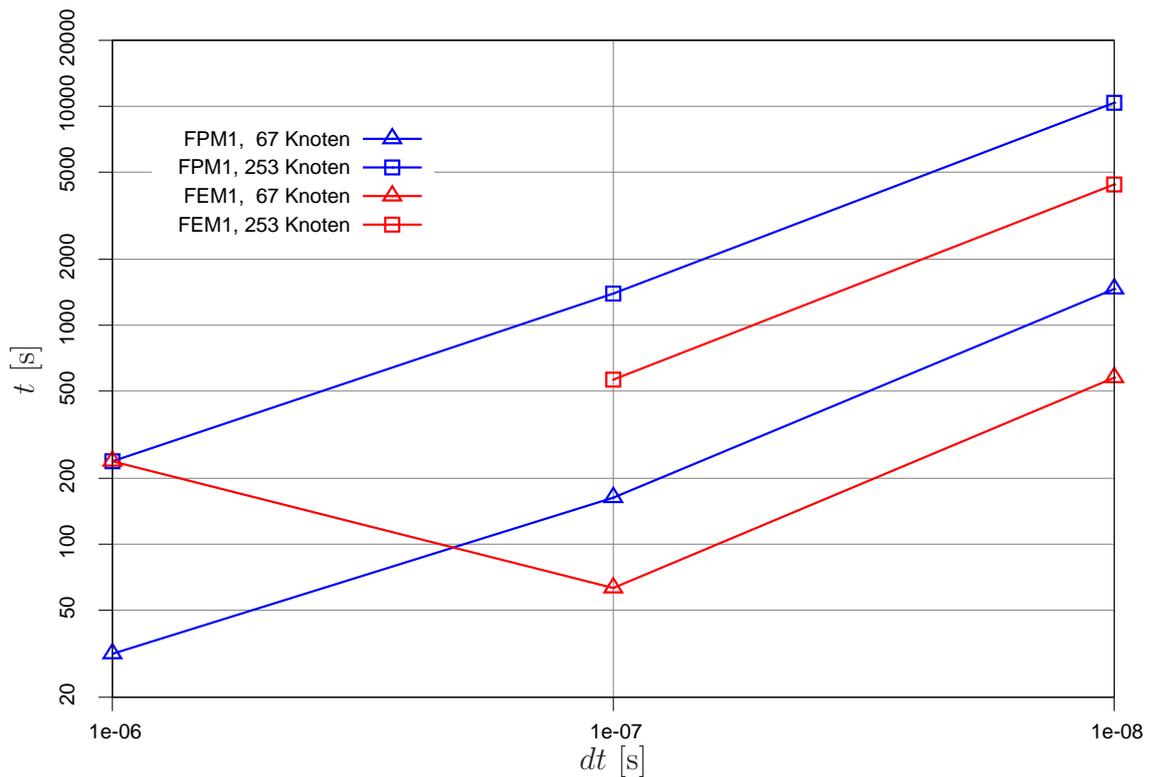


Abbildung 9.26: Beispiel Aufprall, Zeitmessungen, Berechnungen mit FPM und FEM, Variation der Zeitschrittlänge und der räumlichen Diskretisierung

Um den Aufprall realer Projektile zu berechnen, sollte entweder ein rotationsymmetrisches Modell (wenn die Projektile rotationssymmetrisch sind und keine die Rotationssymmetrie störenden Imperfektionen berücksichtigt werden sollen) oder besser ein dreidimensionales Modell verwendet werden.

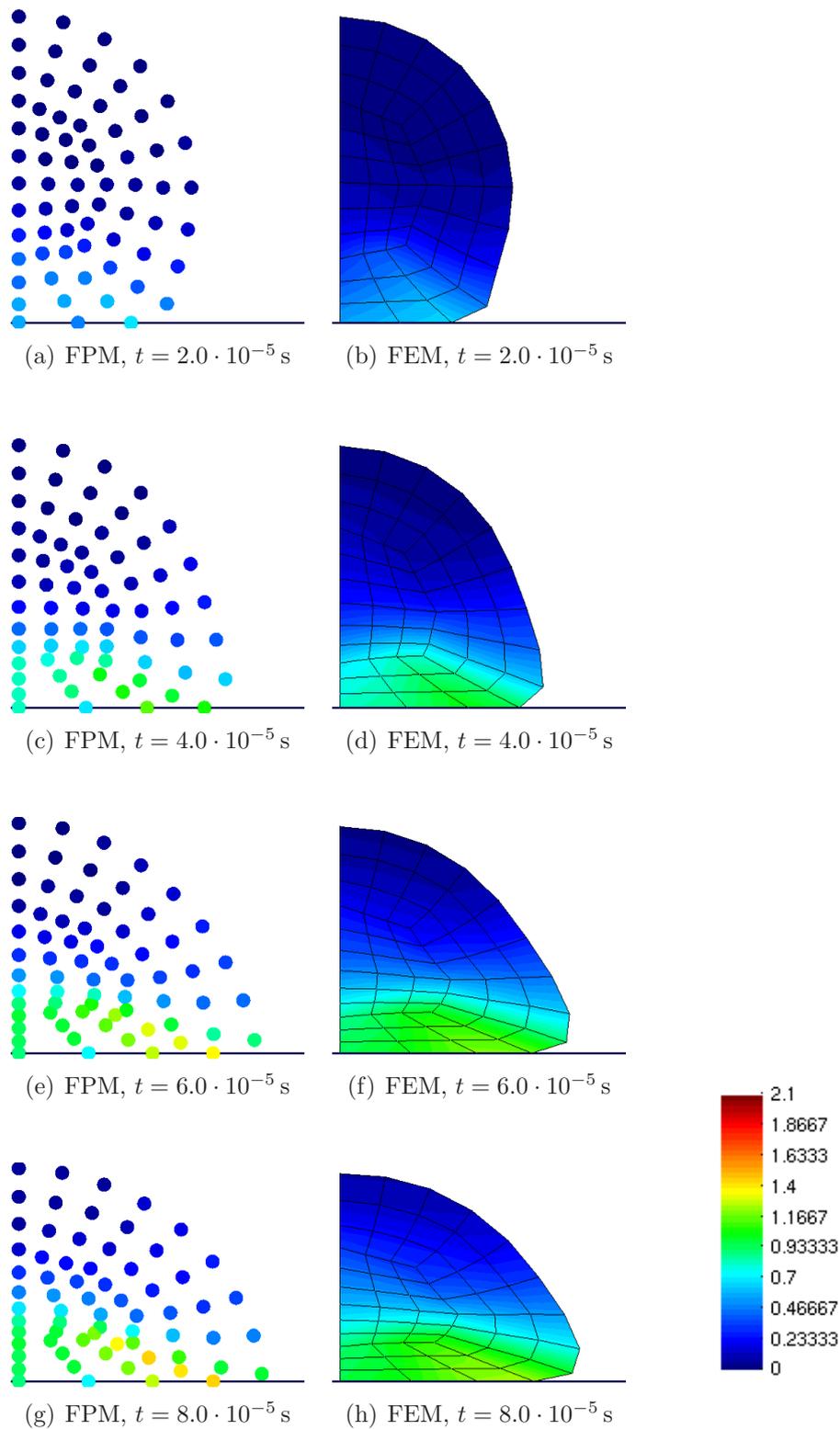


Abbildung 9.27: Beispiel Aufprall, verformte Netze und akkumulierte plastische Dehnung λ , berechnet mit FPM und FEM, Knotenanzahl $N_K = 67$, Zeitschrittlänge $dt = 1 \cdot 10^{-7}$

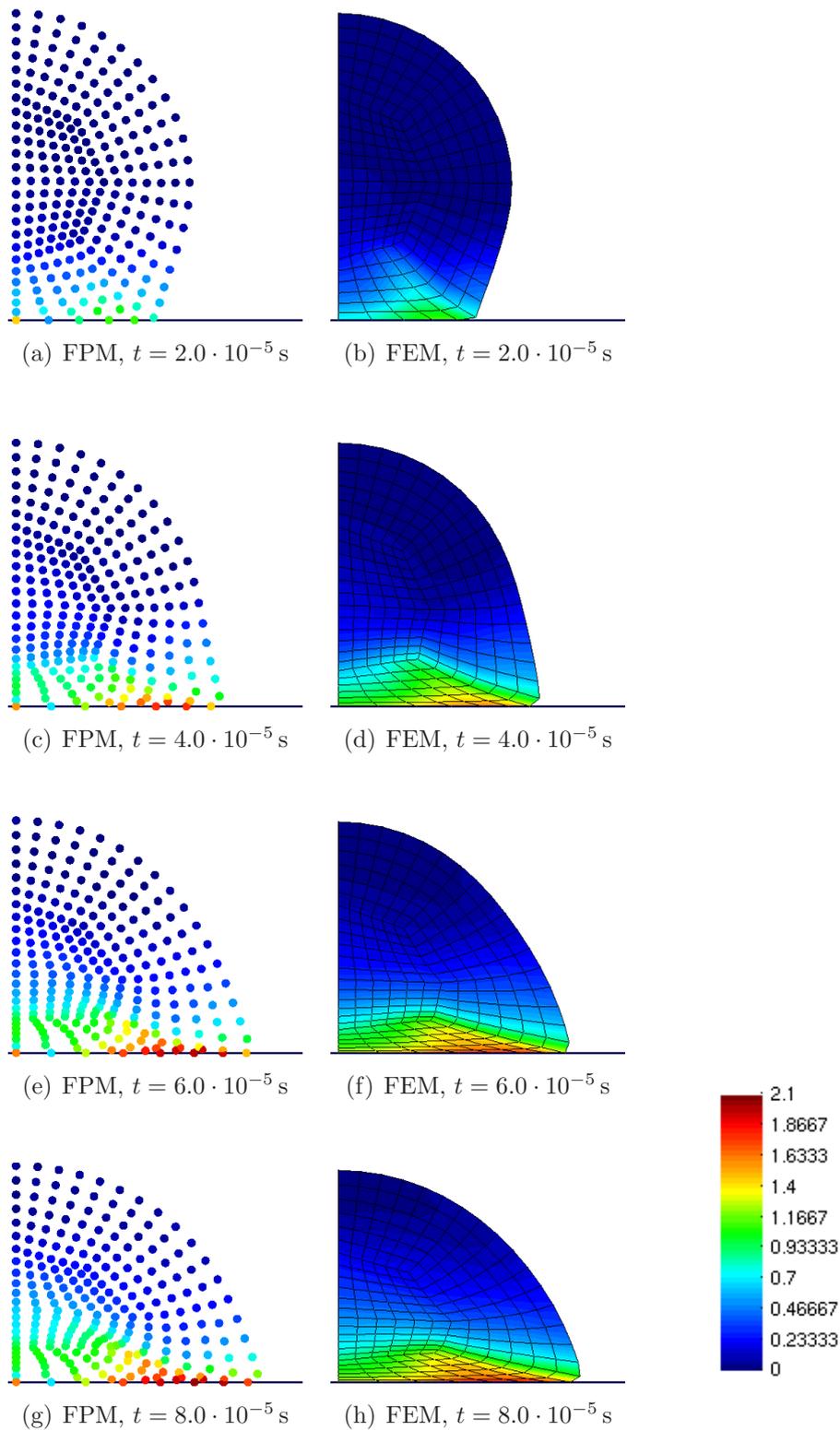


Abbildung 9.28: Beispiel Aufprall, verformte Netze und akkumulierte plastische Dehnung λ , berechnet mit FPM und FEM, Knotenanzahl $N_K = 253$, Zeitschrittlänge $dt = 1 \cdot 10^{-7}$

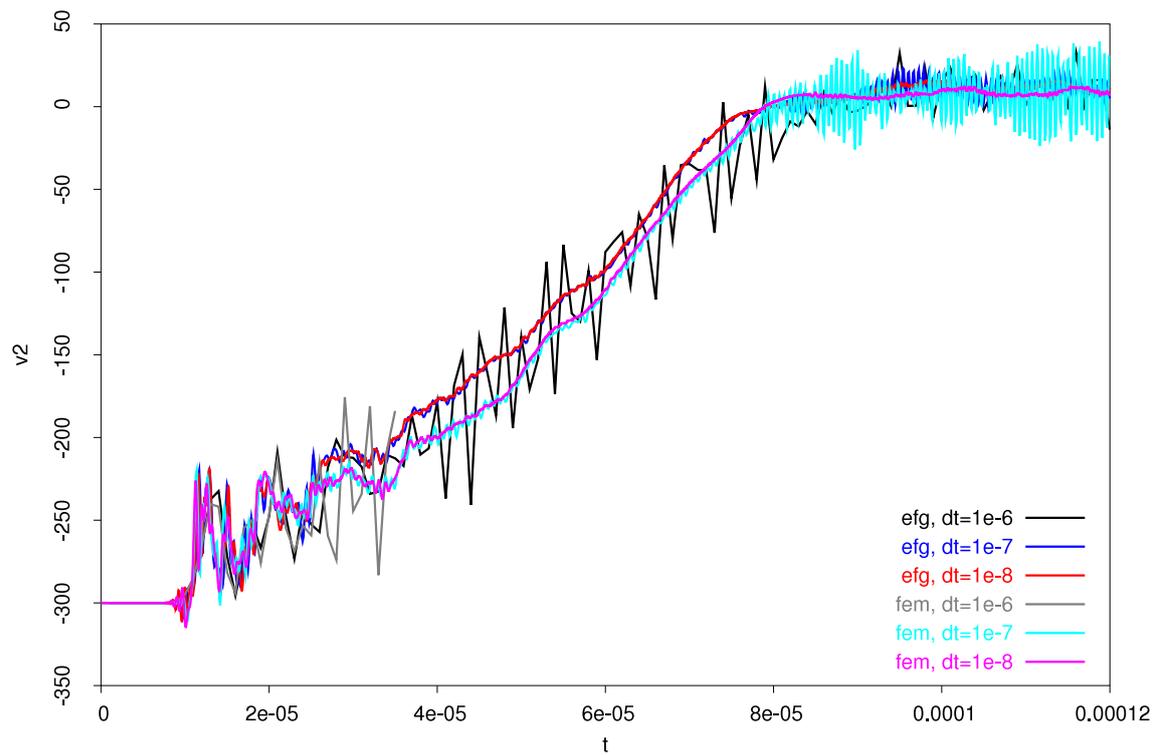
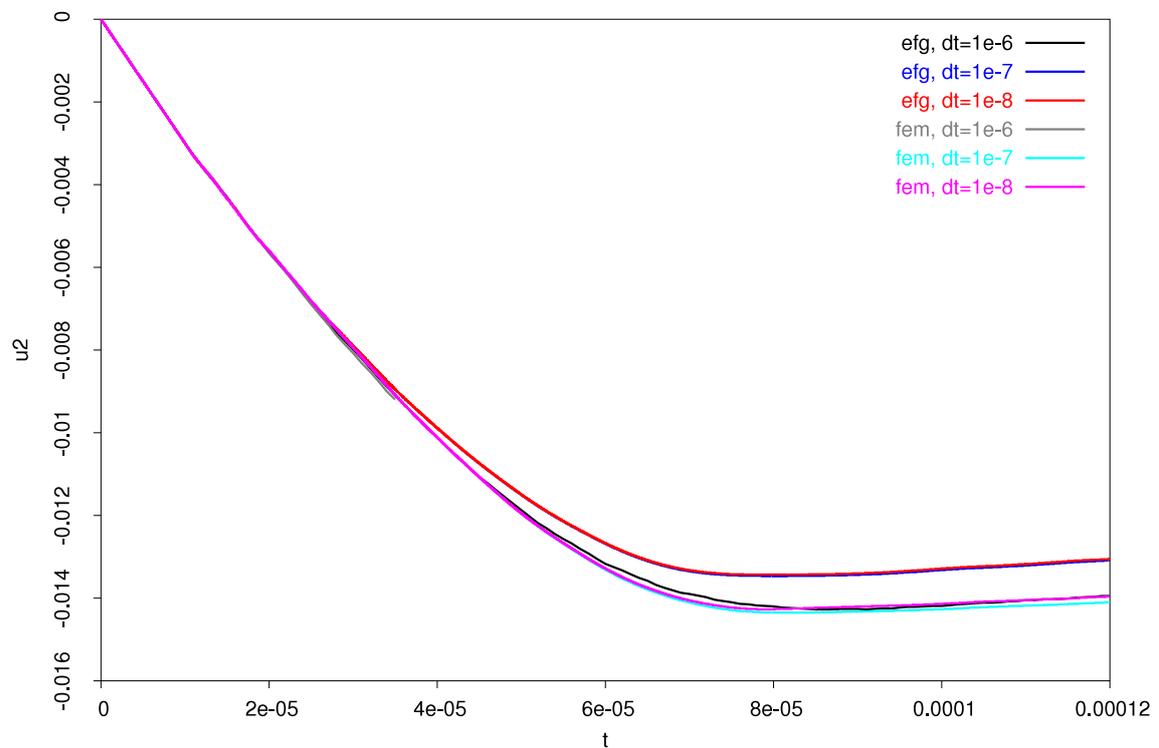
(a) Geschwindigkeit $v_{2,\mathcal{A}}(t)$ (b) Verschiebung $u_{2,\mathcal{A}}(t)$

Abbildung 9.29: Beispiel Aufprall, Geschwindigkeit und Verschiebung des Punktes \mathcal{A} in 2-Richtung, berechnet mit FPM und FEM, Knotenanzahl $N_K = 253$, Variation der Zeitschrittlänge

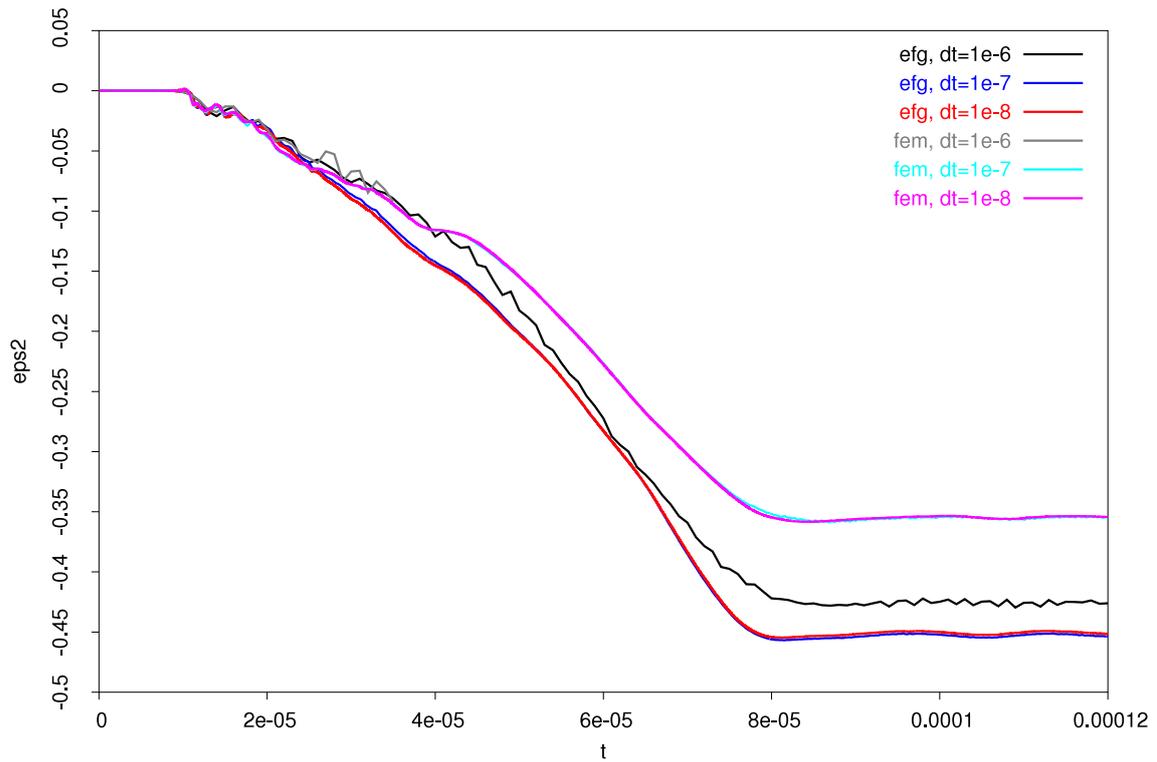
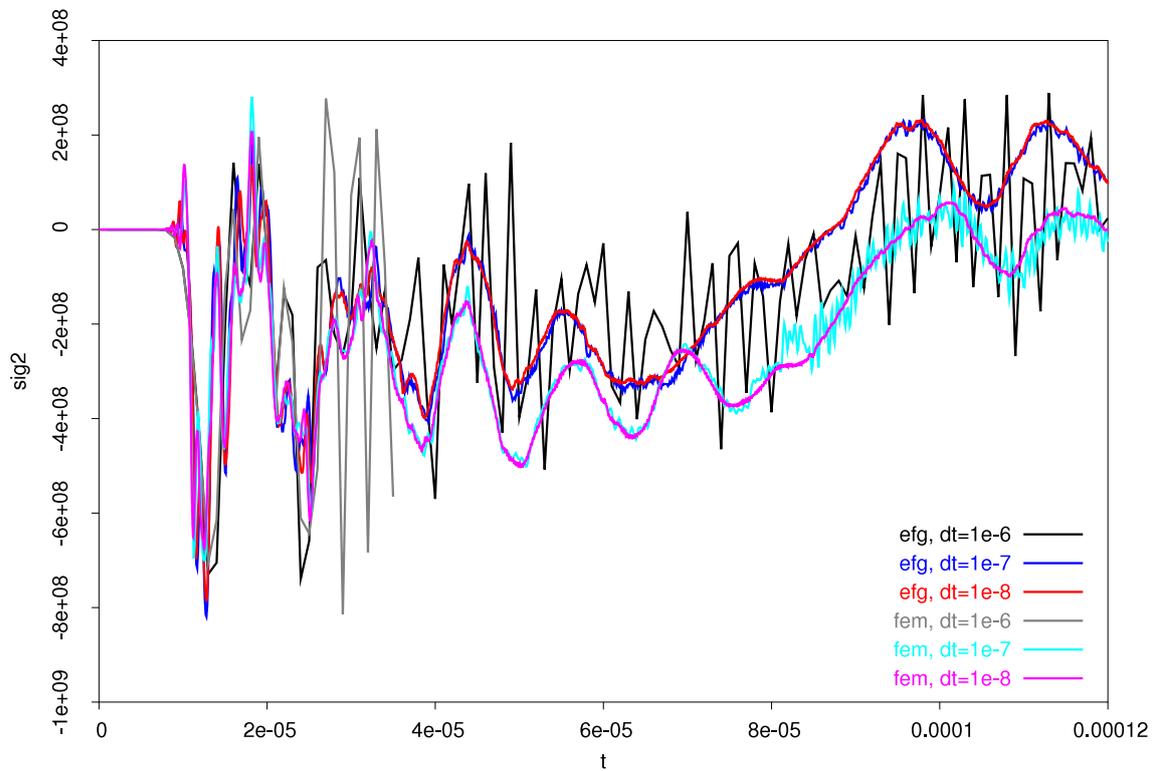
(a) Verzerrung $\varepsilon_{2, \mathcal{A}}(t)$ (b) Spannung $\sigma_{2, \mathcal{A}}(t)$

Abbildung 9.30: Beispiel Aufprall, Verzerrung und Spannung des Punktes \mathcal{A} in 2-Richtung, berechnet mit FPM und FEM, Knotenanzahl $N_K = 253$, Variation der Zeitschrittlänge

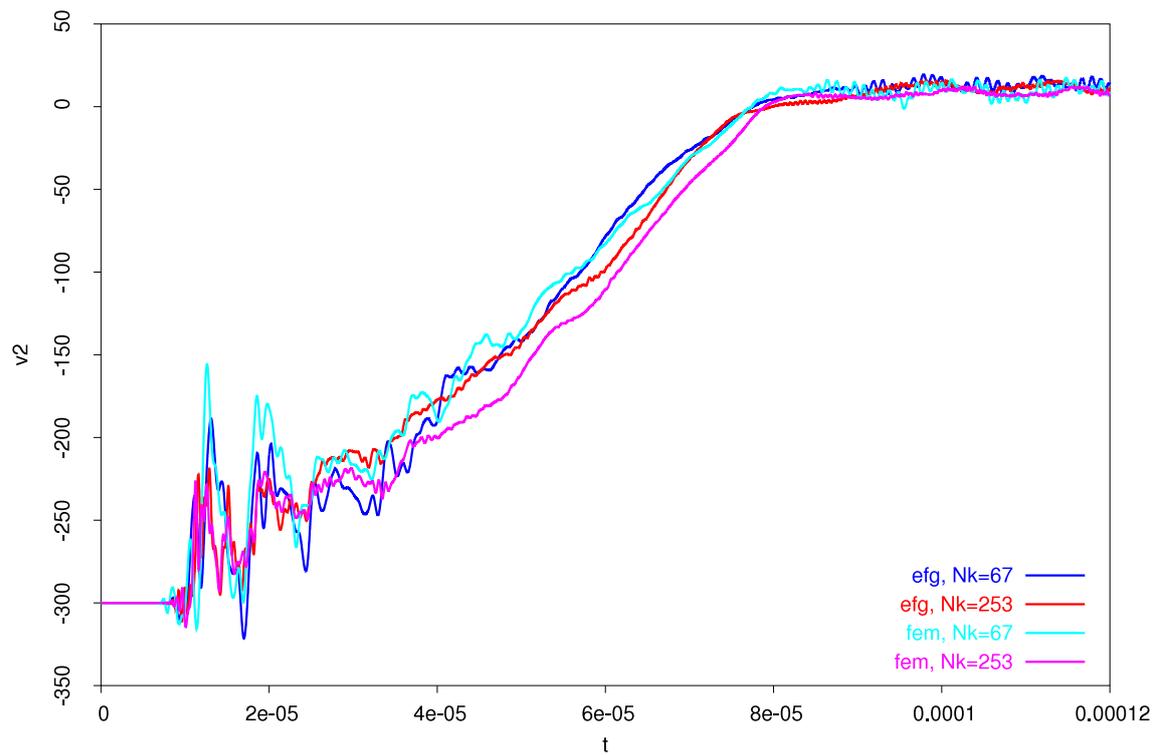
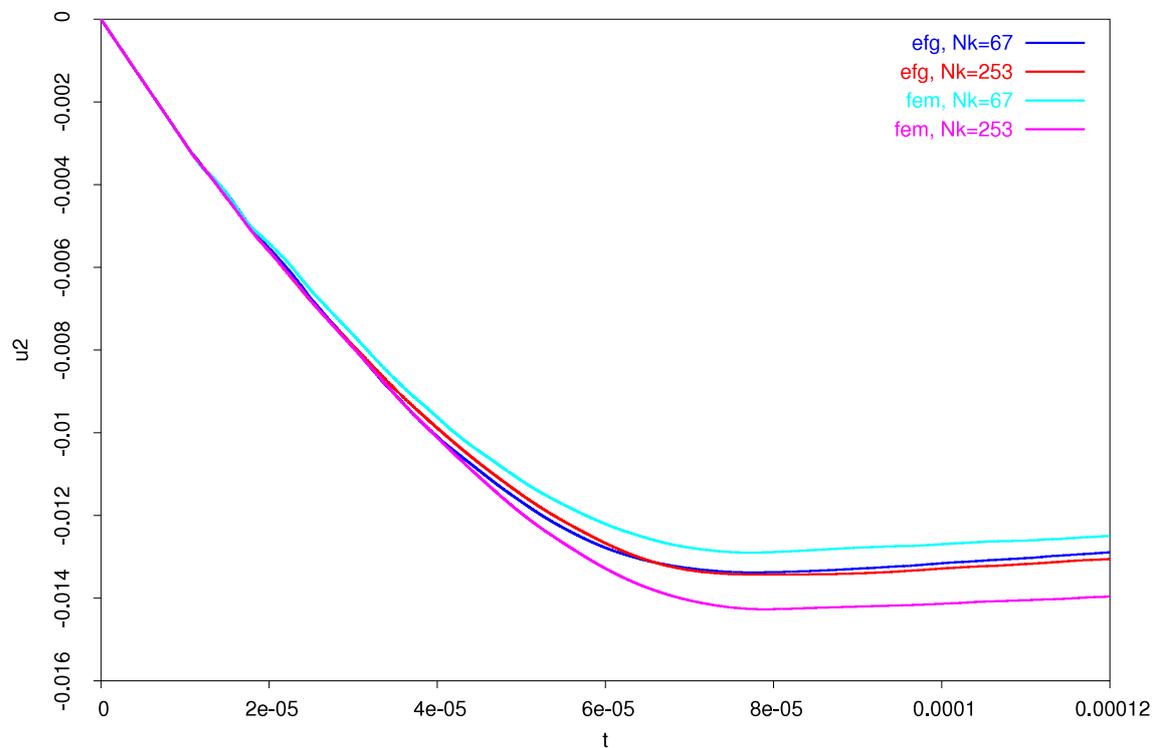
(a) Geschwindigkeit $v_{2,\mathcal{A}}(t)$ (b) Verschiebung $u_{2,\mathcal{A}}(t)$

Abbildung 9.31: Beispiel Aufprall, Geschwindigkeit und Verschiebung des Punktes \mathcal{A} in 2-Richtung, berechnet mit FPM und FEM, Zeitschrittlänge $dt = 1 \cdot 10^{-8}$, Variation der räumlichen Diskretisierung

Kapitel 10

Zusammenfassung

Netzfremde Methoden bieten gegenüber der Finite-Elemente-Methode den Vorteil einer einfacheren Diskretisierung einschließlich leicht zu realisierender h- und p-Adaptivität. Außerdem neigen Punktnetze bei ihrer Deformation weniger zur Entartung als Elementnetze.

Durch Abstraktionen war es möglich, eine Gruppe netzfreier Methoden gemeinsam mit der Finite-Elemente-Methode in einem Programmsystem zu implementieren. Da der überwiegende Teil des Quellcodes von allen Methoden gemeinsam genutzt wird, konnte dieser Teil des Quellcodes mit der in der Literatur besser dokumentierten Finite-Elemente-Methode überprüft werden.

Um den Programmieraufwand in einem vertretbaren Rahmen zu halten, wurde das Programmsystem auf kleine Verformungen und ebene Probleme beschränkt und auf eine grafische Benutzeroberfläche verzichtet. Ansonsten wird der Funktionsumfang eines vollwertigen FEM-Programmes geboten (dynamische Berechnungen, nichtlineares Materialverhalten, Kontakt, zeitabhängige Belastungen, übersichtliche Syntax der Eingabedatensätze, u.s.w).

Für die elementfreien Methoden wird eine Moving Least Square Approximation mit wählbarer Ansatzfunktionsordnung eingesetzt. Bei der Berechnung jedes Ansatzfunktionswertes ist eine Matrix zu invertieren, deren Dimension von der Anzahl der Basisfunktionen abhängt. Bereits im zweidimensionalen Fall wächst die Anzahl der Basisfunktionen und damit der numerische Aufwand bei der Berechnung der Ansatzfunktionen sehr schnell mit der Ansatzfunktionsordnung an.

Mit steigender Ansatzfunktionsordnung muss die Überlappung der Wichtungsfunktionen vergrößert werden, da ansonsten die zu invertierende Matrix singular wird. Es wurden untere und obere Grenzen und optimale Werte für die Wichtungsfunktionsüberlappungen in Abhängigkeit von der Ansatzfunktionsordnung ermittelt.

Die Wichtungsfunktionsüberlappungen bestimmen, welche Knoten über Steifigkeitsbeiträge miteinander gekoppelt sind. Die sich hieraus ergebene Bandbreite des globalen Koordinatensystems ist bei gleicher Ansatzfunktionsordnung größer als die Bandbreite in einer FEM-Berechnung.

Die MLS-Approximation 0-ter Ordnung ist im Rahmen netzfreier Methoden unbrauchbar, da sie das Strukturverhalten deutlich zu steif wiedergibt. Wird nahezu inkompressibles Materialverhalten verwendet, so ist die netzfreie Methode mit linearen MLS-Ansatzfunktionen der Finite-Elemente-Methode mit bilinearen 4-Knoten-Elementen in Bezug auf das Konvergenzverhalten überlegen.

Ein Nachteil der meisten netzfreien Methoden besteht darin, dass Knotenwerte approximiert und nicht interpoliert werden. Wesentliche Randbedingungen und Kontaktbedingungen können daher nicht auf direktem Wege berücksichtigt werden. Im Rahmen der vorliegenden Arbeit werden die wesentlichen Randbedingungen und Kontaktbedingungen durch ein Strafverfahren erzwungen. Das Strafverfahren lässt sich physikalisch einfach interpretieren, übersichtlich implementieren und ist robust. Bei Berechnungen mit doppelter Genauigkeit besteht ein großer Spielraum für die Wahl des Strafparameters.

C^1 -stetige Hermite-Elemente werden eingesetzt, um gekrümmte Oberflächen mit wenigen Kontaktelementen zutreffend („glatt“) diskretisieren zu können.

Die Funktionsfähigkeit des Programmcodes wurde auf der Basis von numerischen Beispielen nachgewiesen.

Das Programmsystem ist flexibel gestaltet, sodass es einfache Möglichkeiten bietet, die implementierten Methoden zu verändern und mit netzfreien Methoden zu experimentieren.

Für die Zukunft wäre es sinnvoll, die Theorie großer Verformungen und eine Erweiterung auf dreidimensionale Probleme zu implementieren.

Anhang A

Eingabedatei

Anhand der Eingabebefehle werden die Berechnungsmöglichkeiten des Programmsystems dargestellt.

Bevor im Abschnitt A.4 eine Beschreibungen aller in der Eingabedatei möglichen Befehle erfolgt, werden in den Abschnitten A.1, A.2 und A.3 das Format der Eingabedatei angegeben, die Befehle in Gruppen eingeteilt und die Syntax der Befehlsbeschreibungen erklärt.

A.1 Dateiformat

Das Format der Eingabedatei ist zeilenorientiert. Ein Befehl nimmt genau eine Zeile ein und setzt sich aus einem Befehlsnamen und Argumenten zusammen. Befehlsnamen bestehen aus einer Folge von Buchstaben und gegebenenfalls dem Unterstrich. Zwischen Groß- und Kleinschreibung wird bei den Befehlsnamen nicht unterschieden. Die Befehlsargumente haben die in Tabelle A.1 angegebenen Formate. Als Trennzeichen dienen Kommata. Leerzeichen und Tabulatorzeichen (im Folgenden werden beide als Leerzeichen bezeichnet) werden überlesen. Argumente, bei denen von einer vorhandenen Voreinstellung nicht abgewichen werden soll, dürfen ausgelassen werden, die Kommata sind aber vollständig anzugeben.

Kommentare werden mit dem Zeichen „!“ eingeleitet. Ab einem Kommentarzeichen wird der Rest der Zeile überlesen. Zeilen, die nur einen Kommentar enthalten, und Leerzeilen sind zulässig.

Die Reihenfolge der Befehlszeilen ist beliebig, da die Eingaberoutine die Befehlszeilen vor der Auswertung sortiert.

Format	Kürzel	Beschreibung	Beispiele
Zeichenkette	s	durch Anführungszeichen eingeschlossene Zeichenfolge, darf Leerzeichen und Kommata enthalten	"Kragbalken, 10 Lastschritte"
Wort	w	aus alphanumerischen Zeichen gebildet, darf keine Leerzeichen oder Kommata enthalten, Groß- und Kleinschreibung wird nicht unterschieden	emi113 $\hat{=}$ Emi113 $\hat{=}$ EMIL13
Logische Variable	b	Wert der Variablen ist <i>wahr</i> bei Eingabe von W, w, T, t oder 1 oder <i>falsch</i> bei Eingabe von F, f oder 0	W
Natürliche Zahl	u	positive ganze Zahl, einschließlich 0	2836 $\hat{=}$ +2836
Ganze Zahl	i	positive oder negative ganze Zahl, einschließlich 0	-845
Gleitkommazahl	f	besteht aus einem optionalen Vorzeichen, einer Ziffernfolge, die auch einen Dezimalpunkt enthalten darf und einem optionalen Exponenten (zur Basis 10), bestehend aus e oder E und einer ganzen Zahl mit optionalem Vorzeichen	3456 $\hat{=}$ +3456.0 $\hat{=}$ 3.456e3 $\hat{=}$ +3.456e+03
Liste logischer Variablen	B	durch Klammern eingeschlossene und durch Kommata getrennte Folge logischer Variablen	(1, 1, 0, 1) $\hat{=}$ (W, w, F, W)
Liste natürlicher Zahlen	U	durch Klammern eingeschlossene und durch Kommata getrennte Folge natürlicher Zahlen, kann auch über eine Schleifenkonstruktion in der Form (s, Startwert, Stopwert, Inkrement) eingegeben werden	(18, 23, 28, 33, 38) $\hat{=}$ (s, 18, 38, 5)
Liste ganzer Zahlen	I	durch Klammern eingeschlossene und durch Kommata getrennte Folge ganzer Zahlen, kann auch über eine Schleifenkonstruktion in der Form (s, Startwert, Stopwert, Inkrement) eingegeben werden	(13, 8, 3, -2, -7) $\hat{=}$ (s, 13, -7, -5)
Liste von Gleitkommazahlen	F	durch Klammern eingeschlossene und durch Kommata getrennte Folge von Gleitkommazahlen	(1.66667e+00, 9.44444e-01)

Tabelle A.1: Formate der Argumente von Eingabebefehlen

A.2 Einteilung der Befehle

Die Eingabebefehle lassen sich nach ihrem Verwendungszweck in Gruppen aufteilen:

- Befehl zur Festlegung des Berechnungstitels
- Befehle zur Auswahl der Ausgaben
- Befehle zur Festlegung der Steuerwerte
- Befehle zur Definition der Berechnungsobjekte (Materialien, Materialpunkte, Knoten, Elementtypen, Elemente, Punktlasten und Kontaktlinien)

In dieser Reihenfolge werden die Befehle weiter unten dargestellt.

Die Befehle zur Festlegung des Berechnungstitels und der Steuerwerte sollten in jeder Eingabedatei nur jeweils einmal auftreten. Werden sie mehrfach verwendet, so werden sie nur an der Stelle ihres ersten Auftretens eingelesen.

Die Befehle zur Definition der Berechnungsobjekte dürfen beliebig häufig verwendet werden. Die den Objekten gegebenen Nummern müssen aber je Objekttyp eindeutig sein. (Es dürfen z.B. nicht zwei Elemente dieselbe Nummer haben, aber ein Element darf dieselbe Nummer haben wie ein Knoten.) Die Nummerierung der Objekte muß nicht notwendigerweise fortlaufend erfolgen.¹

A.3 Syntax der Befehlsbeschreibungen

Zu jedem Befehl werden in einer Übersicht der Befehlsname, symbolische Namen für die Befehlsargumente und eine Kurzbeschreibung des Befehls angegeben:

Übersicht

`befehlsname`, *argumentname1*, *argumentname2*, ..., *argumentnameN* — Kurzbeschreibung des Befehls.

Der Befehlsname sollte in der angegebenen Schreibweise in der Eingabedatei erscheinen. Die Argumentnamen sind durch passende Argumente zu ersetzen, wobei eine Beschreibung der Befehlsargumente Hilfestellung leistet:

¹Lücken in der Numerierung haben keinen Einfluss auf den Speicherbedarf des Programmes, da die Nummern nur als Namen der Objekte dienen. Programmintern wird die Zuordnung zwischen den Objektnummern und den Speicheradressen der Objekte mit Hilfe einer von der STL-Container-Klasse *map* abgeleiteten Klasse vorgenommen.

Argumente

argumentname1 [*Formatkürzel*, *Voreinstellung*]

Beschreibung des ersten Argumentes

argumentname2 [*Formatkürzel*, *Voreinstellung*]

Beschreibung des zweiten Argumentes

⋮

argumentnameN [*Formatkürzel*, *Voreinstellung*]

Beschreibung des N-ten Argumentes

Den symbolischen Argumentnamen sind jeweils ein Formatkürzel (siehe Tabelle A.1), die Voreinstellung und eine Beschreibung des Argumentes zugeordnet.

Teilweise folgen noch Anmerkungen:

Anmerkungen

Weitere Informationen zum Befehl.

A.4 Befehlsbeschreibungen

A.4.1 Befehl `titel`

Übersicht

`titel`, *strtitel* — Legt den Berechnungstitel fest.

Argument

strtitel [*s*, ""]

Zeichenkette zur Beschreibung der Berechnung

A.4.2 Befehl `aus_dat_gid`

Übersicht

`aus_dat_gid`, *baus* — Legt fest, ob Daten zur Nachbearbeitung mit GID ausgegeben werden sollen.

Argument

baus [*b*, **F**]

Ausgabe im GID-Format?

Anmerkungen

Zur Visualisierung können Ergebnisse in einem vom Pre- und Postprozessor *GiD* lesbaren Format herausgeschrieben werden. Mit der unter <http://gid.cimne.upc.es/> kostenlos erhältlichen akademischen Version lassen sich kleinere Modelle (bis 1000 Knoten) visualisieren.

A.4.3 Befehl `aus_dat_gnupl`

Übersicht

`aus_dat_gnup`, *baus*, *bfarbe* — Legt fest, ob eine Steuerdatei zur Erzeugung von Gnuplot-Diagrammen generiert werden soll.

Argumente

baus [*b*, **F**]

Ausgabe einer Steuerdatei?

bfarbe [*b*, **W**]

Gnuplot-Einstellungen in Steuerdatei für farbige Diagramme?

Anmerkungen

Mit dem kostenlos unter <http://www.gnuplot.info/> erhältlichen Programm *Gnuplot* lassen sich Darstellungen von Daten und Funktionen in verschiedenen Grafikformaten über die Kommandozeile erzeugen. Vom vorliegenden Programm wird Gnuplot speziell genutzt, um Bilder von zeitlichen Zustandsgrößenverläufen von Knoten oder Materialpunkten im Encapsulated Postscript Format zu erzeugen.

Hierzu werden alle Einstellungen für Gnuplot in einer Steuerdatei abgelegt. Diese erhält den Namen *berechnungsname.gpl*. Die Objekte, deren Zustandsgrößenverläufe visualisiert werden sollen, werden mit dem Befehl `obj_ausdat` festgelegt. Die Daten werden während der Berechnung in die Dateien *berechnungsname.dat.kno.txt* und *berechnungsname.dat.mpu.txt* geschrieben.

Nach Abschluß der Berechnung werden die Grafikdateien erzeugt, indem auf der Kommandozeile im Berechnungsverzeichnis der Befehl `gnuplot berechnungsname.gpl` eingegeben wird.

A.4.4 Befehl `obj_ausdat`

Übersicht

`obj_ausdat`, `objtyp`, `objnumlist` — Legt die Objekte fest, deren zeitliche Zustandsgrößenverläufe, z.B. mit Gnuplot, dargestellt werden sollen.

Argumente

`objtyp` [*w*, **kno**]

Objektauswahl (**kno** für Knoten und **mpu** für Materialpunkte)

`objnumlist` [*U*, ()]

Liste mit Objektnummern

A.4.5 Befehl `stw_berechverf`

Übersicht

`stw_berechverf` *bknokoerperfest*, *bmpukoerperfest*, *bipukoerperfest*, *bmpuaufkno*, *bipuaufmpu*, *bipuaufkno*, *ansatz_kno_zu_pu*, *ansatz_mpu_zu_ipu*, *ansatz_pu_zu_kno*, *konfigur*, *verzerr* — Definition des Berechnungsverfahrens.

Argumente

bknokoerperfest [*b*, **W**]

Sind die Knoten körperfest?

bmpukoerperfest [*b*, **W**]

Sind die Materialpunkte körperfest? (Als Einstellung ist nur **W** zulässig.)

bipukoerperfest [*b*, **W**]

Sind die Integrationspunkte körperfest?

bmpuaufkno [*b*, **F**]

Liegen die Materialpunkte auf den Knoten?

bipuaufmpu [*b*, *W*]

Liegen die Integrationspunkte auf den Materialpunkten?

bipuaufkno [*b*, *F*]

Liegen die Integrationspunkte auf den Knoten?

ansatz_kno_zu_pu [*w*, *ele*]

Auswahl des Ansatzes mit Knoten als Stützpunkte und Materialpunkten oder Integrationspunkten als Auswertungsorte.

ansatz_mpu_zu_ipu [*w*, *kopieren*]

Auswahl des Ansatzes mit Materialpunkten als Stützpunkte und Integrationspunkten als Auswertungsorte.

ansatz_pu_zu_kno [*w*, *eleinv*]

Auswahl des Ansatzes mit Materialpunkten oder Integrationspunkten als Stützpunkte und Knoten als Auswertungsorte.

konfigur [*w*, *refkonf*]

Konfiguration bzgl. der die schwache Form aufgestellt wird, *refkonf* für Referenzkonfiguration und *momkonf* für Momentankonfiguration

verzerr [*w*, *klverz*]

Art der Verzerrungsberechnung, *klverz* für Theorie kleine Verformungen, *groverz* für Theorie großer Verformungen (nicht implementiert)

A.4.6 Befehl *stw_dimension*

Übersicht

stw_dimension, *dim*, *sym* — Festlegung der Dimension.

Argument

dim [*w*, *2d*]

Räumliche Dimension, nur die Auswahl *2d* für zweidimensionale Berechnungen ist implementiert

sym [*w*, *sp0*]

Ergänzende Angabe zur räumlichen Dimension

Anmerkungen

Der ebene Spannungszustand wird mit *dim=2d* und *sym=sp0* ausgewählt, der ebene Verzerrungszustand mit *dim=2d* und *sym=vz0*.

A.4.7 Befehl `stw_zeitschrittverf`

Übersicht

`stw_zeitschrittverf`, *art*, *param1*, *param2* — Festlegung des Zeitschrittverfahrens.

Argument

art [*w*, `impldyn`]

Auswahl eines impliziten Zeitschrittverfahrens mit `impldyn` und eines expliziten Zeitschrittverfahrens mit `expldyn` (nicht implementiert)

param1 [*f*, 0.25]

Erster Parameter des Newmark-Zeitschrittverfahrens (β)

param2 [*f*, 0.5]

Zweiter Parameter des Newmark-Zeitschrittverfahrens (γ)

A.4.8 Befehl `stw_m`

Übersicht

`stw_m`, *bmasse*, *konzfak* — Einstellungen zur Berücksichtigung der Massenträgheit.

Argument

bmasse [*b*, `W`]

Masse berücksichtigen?

konzfak [*f*, 0.]

Konzentrationsfaktor der Massenmatrix, 1. für vollständig auf Hauptdiagonale konzentrierte Massenmatrix, 0. für konsistente Massenmatrix

Anmerkungen

Bei Wahl von *bmasse*=F werden weder Massenmatrizen noch Massenkräfte berücksichtigt.

A.4.9 Befehl `stw_strafverf`

Übersicht

`stw_strafverf`, `bstrafverf`, `param` — Einstellungen zum Strafverfahren.

Argument

`bstrafverf` [*b*, **W**]

Alle wesentlichen RBs über Strafverfahren berücksichtigen?

`param` [*f*, **1.E6**]

Strafparameter

Anmerkungen

Je nach verwendetem Berechnungsverfahren und der Art der Verschiebungsrandbedingungen können diese zum Teil auch ohne Strafverfahren berücksichtigt werden. Bei Wahl von `bstrafverf=W` wird das Strafverfahren stets für alle Verschiebungsrandbedingungen verwendet.

A.4.10 Befehl `stw_kontakt`

Übersicht

`stw_kontakt`, `bkont`, `ggrenz`, `straffunkparam` — Einstellungen zum Kontakt.

Argument

`bkont` [*b*, **F**]

Kontakt verwenden?

`ggrenz` [*f*, **1.**]

Grenze für Eindringung

`straffunkparam` [*F*, (**1.E6**)]

Strafparameter

Anmerkungen

Bei einer sinnvollen Wahl des Strafparameters dringen Knoten nur sehr wenig in Kontaktoberflächen ein. Überschreiten Knoten die mit dem Parameter *ggrenz* vorgegebene Grenze für die Eindringung, so wird davon ausgegangen, dass die Knoten nicht in die Kontaktoberfläche eingedrungen sind, sondern sich aus anderen Gründen (z.B. komplexe Geometrie) hinter der Kontaktoberfläche befinden. Diese Knoten werden daher beim Kontakt übergangen.

A.4.11 Befehl `stw_nummattangente`

Übersicht

`stw_nummattangente`, *bnummattang* — Einstellungen zur numerischen Materialtangente.

Argument

bnummattang [*b*, **F**]

Numerische Materialtangente verwenden?

dz [*f*, **1.E-8**]

Parameter zur Berechnung der numerischen Materialtangente

Anmerkungen

Die Berechnung der numerischen Tangente wird im Abschnitt 7.2.4 im Zusammenhang mit dem diskretisierten Newton-Verfahren beschrieben.

A.4.12 Befehl `stw_wolken`

Übersicht

`stw_wolken`, *ng*, *bvollst*, *a_mpu*, *a_ipu*, *a_kno* — Einstellungen zu elementfreien Ansätzen (MLS-Approximation).

Argumente

ng [*u*, 1]

Polynomgrad der Funktionsbasis, zulässig sind die Polynomgrade 0 bis 4

bvollst [*b*, F]

Funktionsbasis vollständig bis zum gewählten Polynomgrad?

a_mpu [*f*, 3.25]

Wichtungsfunktionsüberlappungsparameter für Ansätze mit Materialpunkten als Stützpunkte

a_ipu [*f*, 3.25]

Wichtungsfunktionsüberlappungsparameter für Ansätze mit Integrationspunkten als Stützpunkte

a_kno [*f*, 3.25]

Wichtungsfunktionsüberlappungsparameter für Ansätze mit Knoten als Stützpunkte

Anmerkungen

Als Basisfunktionen dienen Monome (siehe Abschnitt 5.5.2), die sich im zweidimensionalen Fall in Form eines Pascalschen Dreiecks anordnen lassen. Wird *bvollst*=F gewählt, so werden die Basisfunktionen den ersten $ng+1$ Zeilen des Pascalschen Dreiecks entnommen. Mit *bvollst*=W werden zusätzlich die Monome des dreiecksförmigen Bereiches unterhalb der $(ng+1)$ -ten Zeile des Pascalschen Dreiecks der Funktionsbasis hinzugefügt. Letztere Einstellung führt zu numerischen Schwierigkeiten bei der Berechnung der MLS-Ansatzfunktionen und sollte daher nicht verwendet werden.

Welche der Parameter *a_mpu*, *a_ipu* und *a_kno* benötigt werden, hängt vom Berechnungsverfahren ab.

A.4.13 Befehl `stw_bneuberech`

Übersicht

`stw_bneuberech`, *bneuberech* — Einstellung, ob Neuberechnung oder Aufsetzen auf Restart-Datei erfolgen soll.

Argumente

bneuberech [*b*, *w*]

Neuberechnung?

Anmerkungen

Aufsetzen auf Restart-Datei ist nicht implementiert.

A.4.14 Befehl *stw_uebtragrichtstart*

Übersicht

stw_uebtragrichtstart, *uebtragrichtstart* — Einstellung zur Datenübertragungsrichtung beim Berechnungsbeginn.

Argumente

uebtragrichtstart [*w*, *kno_zu_mpu*]

Bei *kno_zu_mpu* Datenübertragung von Knoten zu Materialpunkten, bei *mpu_zu_kno* Datenübertragung von Materialpunkten zu Knoten

Anmerkungen

Einstellung hat nur Auswirkung auf Methoden, bei denen die Knoten nicht körperfest sind.

A.4.15 Befehl *stw_numplus*

Übersicht

stw_numplus, *dnum_mpu*, *dnum_ipu*, *dnum_kno*, *dnum_ele* — Erhöht Objektnummern.

Argumente

dnum_mpu [*u*, 0]

Zahl, um die alle Materialpunktnummern erhöht werden

dnum_ipu [*u*, 0]

Zahl, um die alle Integrationspunktnummern erhöht werden

dnum_kno [*u*, 0]

Zahl, um die alle Knotennummern erhöht werden

dnum_ele [*u*, 0]

Zahl, um die alle Elementnummern erhöht werden

Anmerkungen

Sollen die Ausgabedaten mehrerer Berechnungen kombiniert werden, um sie gemeinsam mit einem Postprocessing-Tool zu visualisieren, so kann mit Hilfe des Befehls `stw_numplus` ohne größere Änderungen an den Eingabedateien sichergestellt werden, daß die Objektnummern in der zusammengeführten Ausgabedatei eindeutig sind.

A.4.16 Befehl `stw_t`

Übersicht

`stw_t`, *ta*, *te*, *dt*, *nt*, *tta*, *dtkontakt* — Einstellungen zur zeitlichen Diskretisierung.

Argumente

ta [*f*, 0.]

Zeit am Berechnungsbeginn

te [*f*, 0.]

Zeit am Berechnungsende

dt [*f*, 0.]

Zeitschrittlänge

nt [*u*, 1]

Anzahl Zeitschritte

tta [*u*, 0]

Zeitschrittnummer am Berechnungsanfang

dtkontakt [*f*, 0.]

Zeitschrittlänge bei Kontakt, ohne Angabe gilt *dtkontakt=dt*

Anmerkungen

Es ist nur die Eingabe eines Teiles der Argumente notwendig, da die Argumente voneinander abhängen. Fehlende Angaben werden vom Programm ergänzt.

A.4.17 Befehl *stw_zstrg*

Übersicht

stw_zstrg, *bzstrg*, *unterteilung*, *tiefe* — Einstellungen zur Zeitschrittsteuerung.

Argumente

bzstrg [*b*, W]

Zeitschrittsteuerung verwenden?

unterteilung [*u*, 2]

Zeitschrittunterteilung bei Zeitschrittverfeinerung

tiefe [*u*, 20]

Maximale Anzahl Zeitschrittverfeinerungsstufen

Anmerkungen

Die Zeitschrittsteuerung wird in Abschnitt 7.4 beschrieben.

A.4.18 Befehl *stw_t_aus_dat*

Übersicht

stw_t_aus_dat, *bimmer*, *dt*, *tlist*, *dnt*, *ttlist* — Einstellungen zu Datenausgaben.

Argumente

bimmer [*b*, F]

Stets ausgeben? *bimmer*=W bewirkt Ausgaben in jedem erfolgreich abgeschlossenen Zeitschritt

dt [*f*, 0.]

Zeitdifferenz zwischen Ausgaben

tlist [*F*, ()]

Zeitpunkte für Ausgaben

dnt [*u*, 0]

Zeitschrittdifferenz zwischen Ausgaben

ttlist [*U*, ()]

Zeitschritte für Ausgaben

A.4.19 Befehl `stw_t_aus_ktr`

Übersicht

`stw_t_aus_ktr`, *bimmer*, *dt*, *tlist*, *dnt*, *ttlist* — Einstellungen zu Kontrollausgaben.

Argumente

bimmer [*b*, F]

Stets ausgeben? *bimmer*=W bewirkt Ausgaben in jedem erfolgreich abgeschlossenen Zeitschritt

dt [*f*, 0.]

Zeitdifferenz zwischen Ausgaben

tlist [*F*, ()]

Zeitpunkte für Ausgaben

dnt [*u*, 0]

Zeitschrittdifferenz zwischen Ausgaben

ttlist [*U*, ()]

Zeitschritte für Ausgaben

A.4.20 Befehl `stw_autodtparam`

Übersicht

`stw_autodtparam`, *paramlist* — Angaben zur Zeitschrittlängenabschätzung.

Argumente

paramlist [*F*, (1., 1., 1., 1., 1., 1., 0.)]

Länge, Breite, Dicke, kleinste Elementabmessung, Dichte, E-Modul, Querkontraktionszahl

Anmerkungen

Werden keine hinreichenden Angaben zur zeitlichen Diskretisierung über den Befehl `stw_t` gemacht, so kann eine sinnvolle Zeitschrittlänge vom Programm geschätzt werden, wenn über den Befehl `stw_autodtparam` für das Problem charakteristische Werte angegeben werden.

Als sinnvolle Zeitschrittlänge wird ein Viertel der Periodendauer der Kompressionswelle im kleinsten Element angenommen. Hierzu werden als Angaben die kleinste Elementabmessung, die Dichte, der E-Modul und die Querkontraktionszahl benötigt. Zusätzlich werden zum Vergleich die Periodendauern der freien Längsschwingung und der freien Querschwingung (Biegeschwingung) berechnet und in die Kontrollausgabedatei geschrieben. Hierzu sind (wenn dies beim zu berechnenden Problem sinnvoll ist) zusätzlich charakteristische Werte für Länge, Breite und Dicke anzugeben.

A.4.21 Befehl `stw_gleichloes_bil`

Übersicht

`stw_gleichloes_bil`, *loeser* — Auswahl des Gleichungslösers zur Lösung des linearisierten globalen Gleichungssystems innerhalb der Bilanziteration.

Argument

loeser [*w*, *profil*]

Löser

Argument <i>loeser</i>	Verfahren	Erforderliche Matrixeigenschaften
<code>gauss</code>	Gauß-Algorithmus	regulär
<code>lu</code>	LU-Zerlegung	regulär
<code>profil</code>	Cholesky-Zerlegung, Profilspeichertechnik	regulär, symmetrisch, positiv definit
<code>svd</code>	Singulärwertzerlegung	–

Tabelle A.2: Gleichungslöser

Anmerkungen

Die Auswahlmöglichkeiten für das Befehlsargument *loeser* zeigt Tabelle A.2.

A.4.22 Befehl `stw_gleichloes_mat`

Übersicht

`stw_gleichloes_mat`, *loeser* — Auswahl des Gleichungslösers zur Lösung des linearisierten Materialgleichungssystems innerhalb der Materialiterationen.

Argumente

loeser [*w*, `gauss`]
 Löser

Anmerkungen

Von den Auswahlmöglichkeiten für das Befehlsargument *loeser* nach Tabelle A.2 darf der Profillöser nicht verwendet werden, da er symmetrische Matrizen voraussetzt.

A.4.23 Befehl `stw_gleichloes_ans`

Übersicht

`stw_gleichloes_ans`, *loeser* — Auswahl des Gleichungslösers zur Berechnung der elementfreien Ansatzfunktionen.

Argumente

loeser [*w*, *gauss*]
 Löser

Anmerkungen

Die Auswahlmöglichkeiten für das Befehlsargument *loeser* zeigt Tabelle A.2.

A.4.24 Befehl *stw_bilanzit***Übersicht**

stw_bilanzit, *maxnit*, *schrinke_normdU*, *schrinke_normG*, *bkonvkontr*, *minkonv* — Einstellungen für die globale Iteration.

Argumente

maxnit [*u*, 10]
 Maximale Anzahl Iterationsschritte im Zeitschritt

schrinke_normdU [*f*, 1.E-6]
 Abbruchschranke der Iteration, Norm der Verbesserung des Unbekanntenvektors im Iterationsschritt

schrinke_normG [*f*, 1.E-6]
 Abbruchschranke der Iteration, Norm des Residualvektors im Iterationsschritt (nicht verwendet)

bkonvkontr [*b*, F]
 Konvergenzrate kontrollieren?

minkonv [*f*, 1.]
 Minimale Konvergenzrate

Anmerkungen

Ist die Konvergenzkontrolle eingeschaltet (*bkonvkontr*=W), so wird ab dem zweiten Iterationsschritt kontrolliert, ob die geforderte Konvergenzrate eingehalten wird. Andernfalls wird die Iteration abgebrochen und die Berechnung mit reduzierter Zeitschrittlänge fortgesetzt.

A.4.25 Befehl `stw_matit`

Übersicht

`stw_matit`, `maxnit`, `schrinke_normdz`, `schrinke_normr`, `bkonvkontr`, `minkonv` — Einstellungen für die Materialiteration.

Argumente

`maxnit` [*u*, 10]

Maximale Anzahl Iterationsschritte

`schrinke_normdz` [*f*, 1.E-6]

Abbruchschranke der Iteration, Norm der Verbesserung des Vektors der zusätzlichen (inneren) Variablen im Iterationsschritt

`schrinke_normr` [*f*, 1.E-6]

Abbruchschranke der Iteration, Norm des Residualvektors im Iterationsschritt

`bkonvkontr` [*b*, F]

Konvergenzrate kontrollieren?

`minkonv` [*f*, 1.]

Minimale Konvergenzrate

Anmerkungen

Die Iteration wird beendet, wenn beide Abbruchschranken erfüllt sind.

Ist die Konvergenzkontrolle eingeschaltet (`bkonvkontr=W`), so wird ab dem zweiten Iterationsschritt kontrolliert, ob die geforderte Konvergenzrate eingehalten wird. Andernfalls werden die Materialiteration und die globale Iteration abgebrochen und die Berechnung wird mit reduzierter Zeitschrittlänge fortgesetzt.

A.4.26 Befehl `stw_schrinkefliess`

Übersicht

`stw_schrinkefliess`, `schrinke` — Einstellung zur Fließbedingung.

Argument

schranke [*f*, -1.E-12]

Schranke fuer Fließbedingung

Anmerkungen

Die Materialiteration bei elastoplastischem Materialverhalten zeigt ein besseres Konvergenzverhalten, wenn in der Fließbedingung nicht mit 0, sondern mit einem sehr kleinen negativen Wert verglichen wird.

A.4.27 Befehl `mat`

Übersicht

`mat`, *num*, *mattyp*, *matkennwerte*, *farbe* — Erzeugt Material-Objekt.

Argumente

num [*u*, 0]

Nummer des Materials

mattyp [*u*, 1]

Materialtyp

matkennwerte [*F*, (1., 1., 0., 1., 0.)]

Materialkennwerte

farbe [*u*, 1]

Kennzeichnung

Anmerkungen

Tabelle A.3 zeigt die zur Verfügung stehenden Materialmodelle und die dazugehörigen Materialparameter. Mit dem Argument *num* wird dem Material eine Zahl zugeordnet, über die in anderen Befehlen Bezug auf das Material-Objekt genommen werden kann.

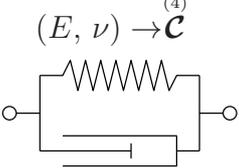
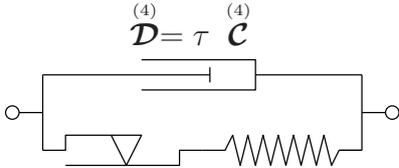
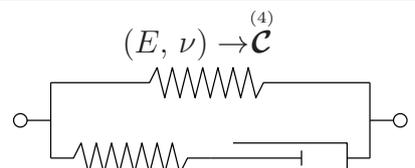
Befehlsargument <i>mattyp</i> und Benennung des Materialtypes	Rheologisches Materialmodell und zugeordnete Materialparameter	Im Befehlsargument <i>matkennwerte</i> übergebene Materialparameter
1, Kelvin-Körper	 $(E, \nu) \rightarrow \overset{(4)}{\mathcal{C}}$ $\overset{(4)}{\mathcal{D}} = \tau \overset{(4)}{\mathcal{C}}$	ρ, E, ν, τ
2, elastoplastisch- viskoser Körper	 $\overset{(4)}{\mathcal{D}} = \tau \overset{(4)}{\mathcal{C}}$ $\sigma_{y0}, H_{iso} \quad (E, \nu) \rightarrow \overset{(4)}{\mathcal{C}}$	$\rho, E, \nu, \tau, \sigma_{y0}, H_{iso}$
3, Zener-M-Körper	 $(E, \nu) \rightarrow \overset{(4)}{\mathcal{C}}$ $(E_2, \nu_2) \rightarrow \overset{(4)}{\mathcal{C}}_2 \quad \overset{(4)}{\mathcal{D}}_2 = \tau \overset{(4)}{\mathcal{C}}_2$	$\rho, E, \nu, \tau, E_2, \nu_2$

Tabelle A.3: Materialmodelle und Materialparameter

A.4.28 Befehl `mpu`

Übersicht

`mpu`, `num`, `matnum`, `vol`, `x0`, `u`, `ut`, `utt`, `z`, `farbe` — Erzeugt Materialpunkt.

Argumente

`num` [`u`, 0]

Nummer des Materialpunktes

`matnum` [`u`, 1]

Nummer des zugeordneten Materials

`vol` [`f`, 1.]

Volumen

`x0` [`F`, ()]

Materialpunktposition in Referenzkonfiguration

$u [F , ()]$
Verschiebung am Berechnungsanfang

$ut [F , ()]$
Geschwindigkeit am Berechnungsanfang

$utt [F , ()]$
Beschleunigung am Berechnungsanfang

$z [F , ()]$
zusätzliche Variablen am Berechnungsanfang

$farbe [u , 1]$
Kennzeichnung

A.4.29 Befehl `kno`

Übersicht

`kno`, `num`, `res`, `vol`, `x0`, `u`, `ut`, `utt`, `z`, `farbe` — Erzeugt Knoten.

Argumente

$num [u , 0]$
Nummer

$res [u , 1]$
(reserviert)

$vol [f , 1 .]$
Volumen

$x0 [F , ()]$
Knotenposition in Referenzkonfiguration

$u [F , ()]$
Verschiebung am Berechnungsanfang

$ut [F , ()]$
Geschwindigkeit am Berechnungsanfang

$utt [F , ()]$
Beschleunigung am Berechnungsanfang

$z [F , ()]$
Zusätzliche Variablen am Berechnungsanfang

$farbe [u , 1]$
Kennzeichnung

A.4.30 Befehl `eletyp`

Argument <i>form</i>	Elementgestalt
<code>dreieck</code>	Dreieck
<code>viereck</code>	Viereck

Tabelle A.4: Geometrische Elementformen

Argument <i>ansfunkx</i> oder <i>ansfunku</i>	Ansatzfunktionstyp
<code>lagro1</code>	Lagrange, linear
<code>lagro2</code>	Lagrange, quadratisch
<code>lagro3</code>	Lagrange, kubisch
<code>sereo1</code>	Serendipity, linear ¹
<code>sereo2</code>	Serendipity, quadratisch
<code>sereo3</code>	Serendipity, kubisch

¹ identisch mit Lagrange, linear

Tabelle A.5: Elementansatzfunktionen

Argument <i>form</i>	Argument <i>ansfunkx</i>	Argument <i>ansfunku</i>	Elementart
<code>dreieck</code>	<code>lagro1</code>	<code>lagro1</code>	isoparametrisch
	<code>viereck</code>	<code>lagro1</code>	
	<code>lagro2</code>	<code>lagro2</code>	
	<code>lagro3</code>	<code>lagro3</code>	
	<code>sereo2</code>	<code>sereo2</code>	
	<code>sereo3</code>	<code>sereo3</code>	
	<code>lagro1</code>	<code>lagro2</code>	
	<code>lagro1</code>	<code>lagro3</code>	superparametrisch
	<code>lagro2</code>	<code>lagr01</code>	
	<code>lagro3</code>	<code>lagr01</code>	

Tabelle A.6: Kombinationen der Elementansätze für Geometrie und Unbekannte

Übersicht

`eletyp`, `num`, `form`, `ansfunkx`, `ansfunku`, `iputyp`, `nipu`, `farbe` — Definition eines Elementtypes.

Argument <i>iputyp</i>	Integrationsart	zul. Werte für Argument <i>nipu</i>	
		<i>form=dreieck</i>	<i>form=viereck</i>
leer	(Punkte werden direkt definiert)	0	0
gauss	Gauß	1, 3, 4, 6, 7, 9, 12, 13	1, 3, 4, 7, 9, 16
newcotgeschl	Newton-Cotes geschlossen	3, 6, 10	4, 9, 16
rasteroffen	regelmäßiges Raster offen	–	1, 4, 9, 16
rastergeschl	regelmäßiges Raster geschlossen	3, 6, 10	4, 9, 16

Tabelle A.7: Arten der numerischen Integration

Argumente*num* [*u* , 0]

Nummer des Elementtypes

form [*w* , viereck]

Geometrische Form

ansfunkx [*w* , *lagr01*]

Ansatzfunktionen zur Beschreibung der Geometrie

ansfunku [*w* , *lagr01*]

Ansatzfunktionen zur Beschreibung der Unbekannten

iputyp [*w* , gauss]

Art der numerischen Integration

nipu [*u* , 4]

Anzahl Integrationspunkte

farbe [*u* , 1]

Kennzeichnung

Anmerkungen

Die Auswahlmöglichkeiten für die Befehlsargumente *form*, *ansfunkx*, *ansfunku*, *iputyp* und *nipu* zeigen die Tabellen A.4 bis A.7.

A.4.31 Befehl `ele`

Übersicht

`ele`, `num`, `matnum`, `eletypnum`, `knonumlist`, `farbe` — Definition eines Elementes.

Argumente

`num` [u , 0]

Nummer des Elementes

`matnum` [u , 1]

Nummer des zugeordneten Materials

`eletypnum` [u , 1]

Nummer des zugeordneten Elementtypes

`knonumlist` [U , ()]

Knotennummern

`farbe` [u , 1]

Kennzeichnung

A.4.32 Befehl `ele_c`

Übersicht

`ele_c`, `elenumlist`, `farbe` — Kennzeichnung von Elementen.

Argumente

`elenumlist` [U , ()]

Elementnummern

`farbe` [u , 1]

Kennzeichnung

A.4.33 Befehl `pu_u`

Übersicht

`pu_u`, `puobjtyp`, `punumlist`, `ulist` — Festlegung der Verschiebung von Punkten am Berechnungsanfang.

Argumente

puobjtyp [*w* , **kno**]

Objekttyp, **kno** für Knoten oder **mpu** für Materialpunkt

punumlist [*U* , ()]

Objektnummern

u [*F* , ()]

Anfangsverschiebung

A.4.34 Befehl *pu_ut*

Übersicht

pu_ut, *puobjtyp*, *punumlist*, *utlist* — Festlegung der Geschwindigkeit von Punkten am Berechnungsanfang.

Argumente

puobjtyp [*w* , **kno**]

Objekttyp, **kno** für Knoten oder **mpu** für Materialpunkt

punumlist [*U* , ()]

Objektnummern

ut [*F* , ()]

Anfangsgeschwindigkeit

A.4.35 Befehl *pu_utt*

Übersicht

pu_utt, *puobjtyp*, *punumlist*, *uttlist* — Festlegung der Beschleunigung von Punkten am Berechnungsanfang.

Argumente

puobjtyp [*w* , **kno**]

Objekttyp, **kno** für Knoten oder **mpu** für Materialpunkt

punumlist [*U* , ()]

Objektnummern

utt [*F*, ()]
Anfangsbeschleunigung

A.4.36 Befehl *pu_z*

Übersicht

pu_z, *puobjtyp*, *punumlist*, *zustvarlist* — Festlegung der zusätzlichen Variablen von Punkten am Berechnungsanfang.

Argumente

puobjtyp [*w*, *kno*]
Objekttyp, *kno* für Knoten oder *mpu* für Materialpunkt

punumlist [*U*, ()]
Objektnummern

z [*F*, ()]
zusätzliche (innere) Variablen

A.4.37 Befehl *pu_c*

Übersicht

pu_c, *puobjtyp*, *punumlist*, *farbe* — Kennzeichnung von Punkten.

Argumente

puobjtyp [*w*, *kno*]
Objekttyp, *kno* für Knoten, *mpu* für Materialpunkt und *ipu* für Integrationspunkt

punumlist [*U*, ()]
Objektnummern

farbe [*u*, 1]
Kennzeichnung

Argument <i>puobjtyp</i>	Last wirkt auf
kno	Knoten
mpu	Materialpunkt

Tabelle A.8: Punktlastziele

Argument <i>lastart</i>	Lastart
beschleunigung	massenbezogene Kraft
kraft	Kraft
verschiebung	Verschiebung

Tabelle A.9: Punktlastarten

Argument <i>funkttyp</i>	Funktion	Argument <i>funkparamlist</i>
kon	Polynom Grad 0	(Konstante)
lin	Polynom Grad 1	(linker Randwert, rechter Randwert)
qua	Polynom Grad 2	(linker Randwert, rechter Randwert, Wert in Zeitintervallmitte)
kub	Polynom Grad 3	(linker Randwert, rechter Randwert, linker Randwert der Ableitung, rechter Randwert der Ableitung)
sin	Sinusfunktion	(Amplitude, Periode, zeitliche Verschiebung, additive Konstante)
cos	Kosinusfunktion	(Amplitude, Periode, zeitliche Verschiebung, additive Konstante)

Tabelle A.10: Punktlastfunktionen

A.4.38 Befehl punktlast

Übersicht

`punktlast`, *num*, *puobjtyp*, *lastart*, *punumlist*, *richtung*, *funkttyp*, *zeitlist*, *funkparamlist* — Definition einer Punktlast.

Argumente*num* [*u*, 0]

Nummer der Punktlast

puobjtyp [*w*, **kno**]Objekttyp, **kno** für Knoten oder **mpu** für Materialpunkt*lastart* [*w*, **kraft**]

Lastart

punumlist [*U*, ()]

Punktnummern

richtung [*f*, 0.]Richtung in Grad bezüglich der globalen *x*-Achse*funkttyp* [*w*, **kon**]

Funktionstyp

zeitlist [*F*, (0., 0.)]

Anfangszeit und Endzeit

funkparamlist [*F*, (0.)]

Funktionsparameter

Anmerkungen

Die Auswahlmöglichkeiten für die Befehlsargumente *puobjtyp*, *lastart* und *funkttyp* zeigen die Tabellen A.4.38 bis A.4.38.

A.4.39 Befehl konlin**Übersicht**

konlin, *num*, *puobjtyp*, *konlintyp*, *punumlist* — Definition einer Kontaktlinie.

Argumente*num* [*u*, 0]

Nummer der Kontaktlinie

puobjtyp [*w*, **kno**]

Punktobjekttyp

konlintyp [*w*, **sp1e1e**]
Kontaktlinientyp

punumlist [*U*, ()]
Punktnummern

Anmerkungen

Das Argument *puobjtyp* bestimmt, welche Objekte (**kno** für Knoten oder **mpu** für Materialpunkte) mit der Kontaktlinie in Kontakt treten können.

Die Kontaktlinien setzen sich aus Hermite-Elementen zusammen, deren Knoten entweder auf den Mitten linearer Stützelemente (*konlintyp=sp1e1e*) oder auf den Knoten dieser Stützelemente (*konlintyp=sp1kno*) liegen.

Die im Argument *punumlist* übergebenen Punktobjektnummern definieren einen polygonalen Linienzug aus linearen Stützelementen. Die Punktobjekte, in der Regel handelt es sich um Knoten, müssen bereits definiert sein.

Eine ausführliche Beschreibung der Kontaktlinien findet sich in Abschnitt 6.3.2.

Abbildungsverzeichnis

1.1	Beispiele für Umformprozesse	2
3.1	Netzausschnitte zu Methoden 1a (FEM) und 1b (FPM)	18
3.2	Netzausschnitte zu Methoden 2a (MPM) und 2b	18
3.3	Netzausschnitte zu Methoden 3a und 2b	19
4.1	Viskoelastische 2-Element-Körper	27
4.2	Viskoelastische 3- und 4-Element-Körper	27
4.3	Zener-M-Körper	28
4.4	Elastoplastischer Körper	29
4.5	Elastoplastisch-viskoser Körper	30
4.6	Wesentliche Randbedingungen und Kontaktbedingungen	33
4.7	Beispiele für Umformprozesse	33
4.8	Strafverfahren, Geometrie einer Verschiebungsrandbedingung	34
4.9	Strafverfahren, Geometrie einer Kontaktbedingung	35
4.10	Strafverfahren, Federkraftgesetz für wesentliche Randbedingung	37
4.11	Strafverfahren, Federkraftgesetz für Kontaktbedingung	37
5.1	Krummlinig berandetes Viereckselement	43
5.2	Krummlinig berandetes Dreieckselement	43
5.3	Punkt in konvex polygonal begrenztem Element	46
5.4	Fehlerfunktion und Gammafunktion	54

5.5	Normierte Wichtungsfunktionen (Exponentialfunktion und Polynom $2n$ -ten Grades), 1D	55
5.6	Normierte Wichtungsfunktion (Exponentialfunktion), 1D	56
5.7	Normierte Wichtungsfunktion (Polynom $2n$ -ten Grades), 1D	57
5.8	Normierte Wichtungsfunktion (Exponentialfunktion), 2D	58
5.9	Normierte Wichtungsfunktion (Polynom $2n$ -ten Grades), 2D	59
5.10	Netzausschnitte zur elementnetzfreien FPM und zur FEM	65
6.1	Hermite-Element	74
6.2	Hermite-Ansatzfunktionen und ihre Ableitungen	75
6.3	Hermite-Element, Einfluss des Tangentenwichtungsfaktors	77
6.4	Kontaktliniendiskretisierung Variante 1 und 2	78
6.5	Hermite-Interpolation eines Kreises	79
6.6	Hermite-Interpolation eines Quadrates mit abgerundeten Ecken	80
6.7	Hermite-Element, Geometrie einer Kontaktbedingung	82
7.1	Zeitschrittsteuerung, Zeitunterteilung	105
9.1	Berechnungsbeispiele	111
9.2	Beispiel Quadratscheibe unter Teillast, System	113
9.3	Diskretisierung mit finiten Punkten	114
9.4	Beispiel Quadratscheibe, MLS-Ansatzfunktionsordnung 0, Variation des Wichtungsfunktionsüberlappungsparameters	117
9.5	Beispiel Quadratscheibe, MLS-Ansatzfunktionsordnung 1, Variation des Wichtungsfunktionsüberlappungsparameters	118
9.6	Beispiel Quadratscheibe, MLS-Ansatzfunktionsordnung 2, Variation des Wichtungsfunktionsüberlappungsparameters	119
9.7	Beispiel Quadratscheibe, MLS-Ansatzfunktionsordnung 3, Variation des Wichtungsfunktionsüberlappungsparameters	120
9.8	Beispiel Quadratscheibe, MLS-Ansatzfunktionsordnung 4, Variation des Wichtungsfunktionsüberlappungsparameters	121

9.9	Beispiel Quadratscheibe unter Teillast, Konvergenzvergleich FEM, FPM	123
9.10	Beispiel Quadratscheibe, Zeitmessungen, FPM und FEM, Variation der räumlichen Diskretisierung	125
9.11	Beispiel Quadratscheibe, Zeitmessungen, FPM und FEM, Variation der räumlichen Diskretisierung	127
9.12	Beispiel Quadratscheibe, Zeitmessungen, FPM, Variation des Wichtungsfunktionsüberlappungsparameters und der räumlichen Diskretisierung	128
9.13	Beispiel Extrusion, System	129
9.14	Beispiel Extrusion, verformte Netze und Dehnung, FPM	130
9.15	Beispiel Extrusion, verformte Netze und Dehnung, FEM	131
9.16	Beispiel Extrusion, Zeitmessungen, FPM und FEM, Variation der räumlichen Diskretisierung	132
9.17	Beispiel Extrusion, Verzerrung und Spannung, FPM und FEM, Variation der räumlichen Diskretisierung	133
9.18	Beispiel Extrusion, Verzerrung und Spannung, FPM und FEM, Variation der räumlichen Diskretisierung	134
9.19	Beispiel Extrusion, Verschiebung, FPM und FEM, Variation räumlichen Diskretisierung	135
9.20	Beispiel Tiefziehen, System	136
9.21	Beispiel Tiefziehen, verformte Netze und Dehnung, FPM	138
9.22	Beispiel Tiefziehen, verformte Netze und Dehnung, FEM	139
9.23	Beispiel Tiefziehen, verformte Netze und Dehnung, FPM	140
9.24	Beispiel Tiefziehen, verformte Netze und Dehnung, FEM	141
9.25	Beispiel Aufprall, System	142
9.26	Beispiel Aufprall, Zeitmessungen, FPM und FEM, Variation der Zeitschrittlänge und der räumlichen Diskretisierung	143
9.27	Beispiel Aufprall, verformte Netze und akkumulierte plastische Dehnung, FPM und FEM, grobe Diskretisierung	144
9.28	Beispiel Aufprall, verformte Netze und akkumulierte plastische Dehnung, FPM und FEM, feine Diskretisierung	145

9.29 Beispiel Aufprall, Geschwindigkeit und Verschiebung, FPM und FEM, Variation der Zeitschrittlänge	146
9.30 Beispiel Aufprall, Verzerrung und Spannung, FPM und FEM, Variation der Zeitschrittlänge	147
9.31 Beispiel Aufprall, Geschwindigkeit und Verschiebung, FPM und FEM, Variation räumlichen Diskretisierung	148

Tabellenverzeichnis

3.1	Methoden und Ansatzfunktionen	16
3.2	Symbole	17
4.1	Rheologische Grundelemente	26
5.1	Fallunterscheidung bei der Bestimmung der (pseudo)inversen Elementansatzfunktionsmatrix	48
5.2	Eigenschaften der Approximation von Elementknotenwerten über die (pseudo)inverse Elementansatzfunktionsmatrix	50
5.3	Wichtungsfunktionen und Faktoren zur Normierung	53
5.4	Grundwerte der Wichtungsfunktionsbereichsbereiche für regelmäßige Punktgitter	60
5.5	Grundwerte der Wichtungsfunktionsbereichsbereiche für unregelmäßige Punktgitter	61
5.6	Funktionsbasen im Zweidimensionalen	62
5.7	Basisfunktionsanzahl	62
5.8	Vergleich der Ansätze zur räumlichen Diskretisierung in FPM und FEM	65
8.1	Einheitensysteme	110
9.1	Beispiel Quadratscheibe unter Teillast, Verschiebungen in Abhängigkeit vom Wichtungsfunktionsüberlappungsparameter	116
9.2	Optimaler Wichtungsfunktionsüberlappungsparameter in Abhängigkeit von der Ansatzfunktionsordnung	116
9.3	Beispiel Quadratscheibe unter Teillast, Diskretisierungen	122

A.1	Formate der Argumente von Eingabebefehlen	152
A.2	Gleichungslöser	167
A.3	Materialmodelle und Materialparameter	171
A.4	Geometrische Elementformen	173
A.5	Elementansatzfunktionen	173
A.6	Kombinationen der Elementansätze für Geometrie und Unbekannte .	173
A.7	Arten der numerischen Integration	174
A.8	Punktlastziele	178
A.9	Punktlastarten	178
A.10	Punktlastfunktionen	178

Literaturverzeichnis

- Altenbach, J., H. Altenbach (1994). *Einführung in die Kontinuumsmechanik*. Teubner, Stuttgart.
- Askes, H., E. C. Aifantis (2002). Numerical modeling of size effects with gradient elasticity - Formulation, meshless discretization and examples. *International Journal of Fracture*, 117:347–358.
- Askes, H., J. Pamin, R. de Borst (2000). Dispersion analysis and element-free Galerkin solutions of second- and fourth-order gradient-enhanced damage models. *International Journal for Numerical Methods in Engineering*, 49:811–832.
- Babuska, I. (1973a). The finite element method with Lagrangian multipliers. *Numerische Mathematik*, 20:179–192.
- Babuska, I. (1973b). The finite element method with penalty. *Mathematics of Computation*, 122:221–228.
- Babuska, I., J. M. Melenk (1997). The partition of unity method. *International Journal for Numerical Methods in Engineering*, 40:727–758.
- Bardenhagen, S. G., J. U. Brackbill, D. Sulsky (2000). The material-point method for granular materials. *Computer Methods in Applied Mechanics and Engineering*, 187:529–541.
- Bathe, K.-J. (2002). *Finite-Elemente-Methoden*. Springer.
- Belytschko, T., W. J. T. Daniel, G. Ventura (2002). A monolithic smoothing-gap algorithm for contact-impact based on the signed distance function.
- Belytschko, T., Fleming (1999). Smoothing, enrichment and contact in the element-free Galerkin method. *Computers and Structures*, 71:173–195.
- Belytschko, T., Y. Krongauz, J. Dolbow, C. Gerlach (1998). On the completeness of meshfree particle methods. *International Journal for Numerical Methods in Engineering*, 43:785–819.

- Belytschko, T., Y. Krongauz, D. Organ, Fleming, Krysl (1996). Meshless methods: An overview and recent developments. *Computer Methods in Applied Mechanics and Engineering*, 139:3–47.
- Belytschko, T., Y. Y. Lu, L. Gu (1994). Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering*, 37:229–256.
- Belytschko, T., Y. Y. Lu, L. Gu (1995a). Crack propagation by element-free Galerkin methods. *Engineering Fracture Mechanics*, 51,2:295–315.
- Belytschko, T., Y. Y. Lu, L. Gu, M. Tabarra (1995b). Element-free Galerkin methods for static and dynamic fracture. *International Journal of Solids and Structures*, 32,17/18:2547–2570.
- Belytschko, T., D. Organ, C. Gerlach (2000). Element-free Galerkin methods for dynamic fracture in concrete. *Computer Methods in Applied Mechanics and Engineering*.
- Bonet, J., S. Kulasegaram (2000). Correction and stabilization of smooth particle hydrodynamics methods with applications in metal forming simulations. *International Journal for Numerical Methods in Engineering*, 47:1189–1214.
- Bonet, J., S. Kulasegaram, M. X. Rodriguez-Paz, M. Profit (2004). Variational formulation for the smooth particle hydrodynamics (SPH) simulation of fluid and solid problems. *Computational Mechanics: Theory and Applications*, 193:1245–1256.
- Brackbill, J. U. (1987). On modeling angular and vorticity in compressible fluid flow. *Computer Physics Communications*, 47:1–.
- Brackbill, J. U. (1988). The ringing instability in particle-in-cell calculations of low speed flow. *Journal of Computational Physics*, 75:469–.
- Brackbill, J. U. (1991). FLIP-MHD: A particle-in-cell method for magnetohydrodynamics. *Journal of Computational Physics*, 96:163–192.
- Brackbill, J. U., D. B. Kothe, H. M. Ruppel (1988). FLIP: A low-dissipation, particle-in-cell method for fluid flow. *Computer Physics Communications*, 48:25–38.
- Brackbill, J. U., H. M. Ruppel (1986). FLIP: A method for adoptively zoned, particle-in-cell calculations in two dimensions. *Journal of Computational Physics*, 65:314–343.

- Brezzi, F. (1974). On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Revue française d'automatique, informatique, recherche opérationnelle / Série rouge*, 8:129–151.
- Brokken, F. B. (2003). *C++ Annotations Version 5.2.1*. Computer Center, University of Groningen, <http://www.icce.rug.nl/documents/>.
- Chadwick, P. (1999). *Continuum Mechanics, Consise Theory and Problems*. Dover Publications, Mineola, New York.
- Chen, J.-S., W. Han, Y. You, X. Meng (2003). A reproducing kernel method with nodal interpolation property. *International Journal for Numerical Methods in Engineering*, 56:935–969.
- Chen, J.-S., C. Pan, C.-T. Wu, W. K. Liu (1996). Reproducing kernel particle methods for large deformation analysis of non-linear structures. *Computational Mechanics: Theory and Applications*, 139:195–227.
- Cleveland, W. S. (1993). *Visualizing Data*. AT&T Bell Laboratories, Murray Hill, NJ.
- Crisfield, M. A. (1991). *Non-linear Finite Element Analysis of Solids and Structures, vol 1*. J. Wiley, Chichester.
- Crisfield, M. A. (1997). *Non-linear Finite Element Analysis of Solids and Structures, vol. 2*. J. Wiley, Chichester.
- Dettmer, W., D. Peric (2003). An analysis of the time integration algorithms for the finite element solutions of incompressible Navier-Stokes equations based on a stabilised formulation. *Computational Mechanics: Theory and Applications*, 192:1177–1226.
- Dhatt, G., G. Touzot, G. Cantin (1984). *The Finite Element Method Displayed*. John Wiley and Sons.
- Duarte, C. A., J. T. Oden (1995a). A review of some meshless methods to solve partial differential equations. Techn. Ber.
- Duarte, C. A., J. T. Oden (1995b). Hp clouds - a meshless method to solve boundary-value problems. Techn. Ber.
- Duarte, C. A., J. T. Oden (1996). An h-p adaptive method using clouds. *Computer Methods in Applied Mechanics and Engineering*, 139:237–262.

- El-Abbasi, N., S. Meguid, A. C. (2001). On the modelling of smooth contact surfaces using cubic splines.
- Engeln-Müllges, G., F. Uhlig (1996). *Numerical Algorithms with C*. Springer-Verlag.
- Fasshauer, G. E. (2004). Toward approximate moving least squares approximation with irregularly spaced centers. *Computer Methods in Applied Mechanics and Engineering*, 193:1231–1243.
- Fernández-Méndez, S., A. Huerta (2004). Imposing essential boundary conditions in mesh-free methods. *Computer Methods in Applied Mechanics and Engineering*, 193:1257–1275.
- Fries, T.-P., H.-G. Matthies (2004). Classification and Overview of Meshfree Methods. Techn. Ber.
- Gingold, R. A., J. J. Monaghan (1977). Smoothed particle hydrodynamics: theory and application to non-spherical stars. *Monthly Notices of the Royal Astronomical Society*, 181:375–389.
- Girault, V. (1974). Theory of a finite difference method on irregular networks. *SIAM Journal on Numerical Analysis*, 11:260–282.
- Gordon, W. J., J. A. Wixom (1978). Shepard's method of 'metric interpolation' to bivariate and multivariate data. *Mathematics of Computation*, 32:253–264.
- Gorn, S. (1954). The Automatic Analysis and Control of Computing Errors. *Journal of the Society for Industrial and Applied Mathematics*, 2 (2):69–81.
- Griebel, M., M. A. Schweitzer (2002a). *A particle-partition of unity method. Part V: Boundary conditions*, 517–540. Springer-Verlag, Berlin.
- Griebel, M., M. A. Schweitzer, Hg. (2002b). *Meshfree Methods for Partial Differential Equations*, Bd. 26.
- Guilkey, J. E., J. A. Weiss (2003). Implicit time integration for the material point method: Quantitative and algorithmic comparisons with the finite element method. *International Journal for Numerical Methods in Engineering*, 57:1323–1338.
- Gummert, P., K.-A. Reckling (1986). *Mechanik*. Vieweg, Braunschweig, Wiesbaden.
- Hansbo, P., M. G. Larson (2002). Discontinuous Galerkin methods for incompressible and nearly incompressible elasticity by Nitschet's method. *Computer Methods in Applied Mechanics and Engineering*, 191:1895–1908.

- Harlow, F. H. (1964). The particle-in-cell computing method for fluid dynamics. *Methods in computational physics*, 3:319–343.
- Hibbitt, B. I., H. D.; Karlsson (1979). Analysis of Pipe Whip.
- Huerta, A., S. Fernández-Méndez (2000). Enrichment and coupling of the finite element and meshless methods. *International Journal for Numerical Methods in Engineering*, 48:1615–1636.
- Huerta, A., S. Fernández-Méndez, W. K. Liu (2004a). A comparison of two formulations to blend finite elements and mesh-free methods. *Computer Methods in Applied Mechanics and Engineering*, 193:1105–1117.
- Huerta, A., Y. Vidal, P. Villon (2004b). Pseudo-divergence-free Galerkin method for incompressible fluid flow. *Computer Methods in Applied Mechanics and Engineering*, 193:1119–1136.
- Johnson, G. R., S. R. Beissel (1996). Normalized smoothing functions for SPH impact computations. *International Journal for Numerical Methods in Engineering*, 39:2725–2741.
- Knothe, K., H. Wessels (1991). *Finite Elemente*. Springer-Verlag, Berlin, Heidelberg.
- Krok, O. J., J. (1989). A unified approach to the adaptive meshless FDM and FEM. In *Discretization methods in structural mechanics*, 353–362. IUTAM Symposium.
- Kuhl, D. (1996). *Stabile Zeitintegrationsalgorithmen in der nichtlinearen Elastodynamik dünnwandiger Tragwerke*. Dissertation, Universität Stuttgart, Lehrstuhl für Baustatik.
- Lagally, M. (1964). *Vorlesungen über Vektorrechnung*. Akademische Verlagsgesellschaft Geest & Portig K.-G., Leipzig.
- Lancaster, P., H. Salkauskas (1981). Surfaces generated by moving least squares methods. *Mathematics of Computation*, 37,155:141–158.
- Liszka, T. J. (1984). An interpolation method for an irregular net of nodes. *International Journal for Numerical Methods in Engineering*, 20:1599–1612.
- Liszka, T. J., C. A. M. Duarte, W. W. Tworzydło (1996). hp-Meshless cloud method. *Computer Methods in Applied Mechanics and Engineering*, 139:263–288.

- Liszka, T. J., J. Orkisz (1977). The finite difference method at arbitrary irregular meshes in non-linear problems of applied mechanics. In *Proc. 4th Int. Conference on Structural Mechanics in Reactor Technology, San Francisco, California*, Bd. M 3/8, 1–10.
- Liszka, T. J., J. Orkisz (1980). The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Computers and Structures*, 11:83–95.
- Liu, W. K., Y. Chen (1995). Wavelet and multiple scale reproducing kernel methods. *International Journal for Numerical Methods in Fluids*, 21:901–933.
- Liu, W. K., Y. Chen, R. A. Uras, C. T. Chang (1996). Generalized multiple scale reproducing kernel particle methods. *Computer Methods in Applied Mechanics and Engineering*, 139:91–157.
- Liu, W. K., S. Jun, S. Li, J. Adee, T. Belytschko (1995a). Reproducing kernel particle methods for structural dynamics. *International Journal for Numerical Methods in Engineering*, 38:1655–1679.
- Liu, W. K., S. Jun, Y. F. Zhang (1995b). Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, 20:1081–1106.
- Liu, W. K., S. Li, T. Belytschko (1997). Moving least-square reproducing kernel methods; (I) Methodology and convergence. *Computer Methods in Applied Mechanics and Engineering*, 143:113–154.
- Liu, W. K., et al. (2004). Reproducing kernel element method. Part I – Part IV. *Computer Methods in Applied Mechanics and Engineering*, 193:933–1034.
- Lu, Y. Y., T. Belytschko, L. Gu (1994). A new implementation of the element free Galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 113:397–414.
- McLain (1974). Drawing contours from arbitrary data points. *Computer Journal*, 17:318–324.
- Melenk, J. M., I. Babushka (1996). The partition of unity finite element method: Basic theory and applications. *Computer Methods in Applied Mechanics and Engineering*, 139:289–314.
- Mendez, B., A. Velazquez (2004). Finite point solver for the simulation of 2-D laminar incompressible unsteady flows. *Computer Methods in Applied Mechanics and Engineering*, 193:825–848.

- Meyers, S. (1997). *Mehr Effektiv C++ programmieren*. Addison-Wesley.
- Meyers, S. (1998). *Effektiv C++ programmieren*. Addison-Wesley.
- Monaghan, J. J. (1982). Why particle methods work. *SIAM Journal on Scientific and Statistical Computing*, 3(4):422–433.
- Monaghan, J. J. (1988). An introduction to SPH. *Computer Physics Communications*, 48:89–96.
- Monaghan, J. J., A. Kocharyan (1995). SPH simulation of multi-phase flow. *Computer Physics Communications*, 87:225–235.
- Moresi, L., F. Dufour, H.-B. Mühlhaus (2003). A Lagrangian integration point finite element method for large deformation modeling viscoelastic geomaterials. *Journal of Computational Physics*, 184:476–497.
- Nay, R. A., S. Utku (1972). An alternative for the finite element method. Bd. 3, 62–74. Variational methods in engineering : proceedings of an International Conference held at the University of Southampton 25th September, 1972.
- Nayroles, B., G. Touzot, P. Villon (1992). Generalizing the finite element method: diffuse approximation and diffuse elements. *Computational Mechanics: Theory and Applications*, 10:307–318.
- Newmark, N. M. (1959). A method of computation for structural dynamics. In *Proceedings of ASCE, Journal of Engineering Mechanics*, Bd. 85, 67–94.
- Nitsche, J. (1970). Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36:9–15.
- Oden, J. T., C. A. M. Duarte, O. C. Zienkiewicz (1998). A new cloud-based hp finite element method. *Computer Methods in Applied Mechanics and Engineering*, 153:117–126.
- Onate, E., S. Idelsohn, O. C. Zienkiewicz, R. L. Taylor (1996a). A Finite Point Method in Computational Mechanics. Applications to Convective Transport and Fluid Flow. *International Journal for Numerical Methods in Engineering*, 39:3839–3866.

- Oñate, E., S. Idelsohn, O. C. Zienkiewicz, R. L. Taylor, C. Sacco (1996b). A stabilized finite point method for analysis of fluid mechanics problems. *Computer Methods in Applied Mechanics and Engineering*, 139:315–346.
- Pamin, J., H. Askes, R. de Borst (2003). Two gradient plasticity theories discretized with the element-free Galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 192:2377–2403.
- Pavlin, V., N. Peronne (1975). Finite difference energy techniques for arbitrary meshes. *Computers and Structures*, 5:45–58.
- Perrone, N., R. Kao (1975). A general finite difference method for arbitrary meshes. *Computers and Structures*, 5:45–48.
- Prager, W. (1955). *Probleme der Plastizitätstheorie*.
- Press, W. H. (2002). *Numerical Recipes in C++, The Art of Scientific Computing, Second Edition*. Cambridge University Press.
- Puso, M. A., T. A. Laursen (2002). A 3D contact smoothing method using Gregory patches.
- Reckling, K. A. (1967). *Plastizitätstheorie und ihre Anwendung auf Festigkeitsprobleme*. Springer-Verlag, Berlin, Heidelberg.
- Reese, S. (2000). *Thermomechanische Modellierung gummiartiger Polymerstrukturen*. Habilitationsschrift, Universität Hannover, Institut für Baumechanik und Numerische Mechanik.
- Shepard, D. (1968). A two-dimensional interpolation function for irregularly-spaced data. In *ACM National Conference, Proceedings*, 517–524.
- Snell, C., D. G. Vesey, P. Mullord (1981). The application of a general finite difference method to some boundary value problems. *Computers and Structures*, 13:547–552.
- Stadler, M., G. A. Holzapfel, J. Korelc (2003). C1-continuous modelling of smooth contact surfaces using NURBS and application to 2D problems.
- Stroustrup, B. (2000). *The C++ Programming Language, Special Edition*. Addison-Wesley.
- Sulsky, D., A. Kaul (2004). Implicit dynamics in the material-point method. *Computer Methods in Applied Mechanics and Engineering*, 193:1137–1170.

- Sulsky, D., H. L. Schreyer (1996). Axisymmetric form of the material point method with applications to upsetting and Taylor impact problems . *Computer Methods in Applied Mechanics and Engineering*, 139:409–429.
- Sulsky, D., Zhou, H. L. Schreyer (1995). Application of a particle-in-cell method to solid mechanics. *Computer Physics Communications*, 87:236–252.
- Swegle, J. W., D. L. Hicks, S. W. Attaway (1995). Smoothed particle hydrodynamics stability analysis. *Journal of Computational Physics*, 116:123–134.
- Szabó, I. (1984a). *Einführung in die Technische Mechanik*. Springer-Verlag, Berlin, Heidelberg, New York, Tokyo.
- Szabó, I. (1984b). *Höhere Technische Mechanik*. Springer-Verlag, Berlin, Heidelberg, New York, Tokyo.
- Szyler, P. (1997). *Numerische Mathematik, Teil 1 bis 3*. Skript, Otto-von-Guericke-Universität Magdeburg, Fakultät für Mathematik, Institut für Analysis und Numerik.
- Tabbara, M., T. Blacker, T. Belytschko (1994). Finite element derivative recovery by moving least square interpolants. *Computer Methods in Applied Mechanics and Engineering*, 117:211–223.
- Taylor, R. L., O. C. Zienkiewicz, Onate, E. (1998). A hierarchical finite element method based on the partition of unity. *Computer Methods in Applied Mechanics and Engineering*, 152:73–84.
- Trostel, R. (1993). *Mathematische Grundlagen der Technischen Mechanik I, Vektor- und Tensoralgebra*. Vieweg, Braunschweig, Wiesbaden.
- Trostel, R. (1997). *Mathematische Grundlagen der Technischen Mechanik II, Vektor- und Tensoranalysis*. Vieweg, Braunschweig, Wiesbaden.
- Trostel, R. (1999). *Mathematische Grundlagen der Technischen Mechanik III, Materialmodelle in der Ingenieurmechanik*. Vieweg, Braunschweig, Wiesbaden.
- Voß, H. (2001). *Grundlagen der Numerischen Mathematik*. Skript, Technische Universität Hamburg-Harburg, Arbeitsbereich Mathematik, www.tu-harburg.de/mat/LEHRE/material/grnummath.pdf.
- Waller, H. (1989). *Schwingungslehre für Ingenieure*. BI-Wissenschaftsverlag.

- Wood, W. L. (1981). An alpha modification of newmark's method. *International Journal for Numerical Methods in Engineering*, 15:1562–1566.
- Wood, W. L. (1990). *Practical Time-stepping Schemes*. Clarendon Press, Oxford.
- Wriggers, P. (2001). *Nichtlineare Finite-Element-Methoden*. Springer.
- Wriggers, P., L. Krstulovic-Opara, J. Korelc (2001). Smooth C1-interpolations for two-dimensional frictional contact problems. *International Journal for Numerical Methods in Engineering*, 51:1469–1495.
- York II, A. R., D. Sulsky, H. L. Schreyer (1999). The material point method for simulation of thin membranes. *International Journal for Numerical Methods in Engineering*, 44:1429–1456.
- York II, A. R., D. Sulsky, H. L. Schreyer (2000). Fluid-membrane interaction based on the material point method. *International Journal for Numerical Methods in Engineering*, 48:901–924.
- Zavarise, G., P. Wriggers, B. A. Schrefler (1998). A method for solving contact problems. *International Journal for Numerical Methods in Engineering*, 42:473–498.
- Zhu, T., S. N. Atluri (1998). A modified collocation method and a penalty formulation for enforcing the essential boundary in the element free Galerkin method. *Computational Mechanics: Theory and Applications*, 21 (3):211–222.
- Zienkiewicz, O. C., R. L. Taylor (2000a). *The Finite Element Method, Volume 1, The Basics*. Butterworth-Heinemann.
- Zienkiewicz, O. C., R. L. Taylor (2000b). *The Finite Element Method, Volume 2, Solid Mechanics*. Butterworth-Heinemann.
- Zienkiewicz, O. C., R. L. Taylor (2000c). *The Finite Element Method, Volume 3, Fluid Mechanics*. Butterworth-Heinemann.

Lebenslauf

Olaf Schilling	Am Panorama 1, 15834 Rangsdorf
29.01.1966	geboren in Berlin
08/1972 – 07/1978	Clemens-Brentano-Grundschule in Berlin Lichterfelde
08/1978 – 12/1984	Goethe-Oberschule (Gymnasium) in Berlin-Lichterfelde
01/1985 – 03/1985	Praktikum bei der SUSPA Spannbeton GmbH in Langenfeld/Rhld.
04/1985	Beginn des Bauingenieurstudiums mit der Fachrichtung Konstruktiver Ingenieurbau an der Technischen Universität Berlin
02/1988	Vordiplom
04/1988 – 03/1992	Studentischer Mitarbeiter mit Unterrichtsaufgaben am 1. Institut für Mechanik an der Technischen Universität Berlin
05/1992 – 11/1994	Studentischer Mitarbeiter bei der INPRO in Berlin
10/1993 – 09/2000	Selbstständige Tätigkeit als Programmierer Freiberuflicher Mitarbeiter in einem Statikbüro
03/2000	Hauptdiplom
10/2000 – 10/2005	Wissenschaftlicher Mitarbeiter in der Arbeitsgruppe Numerische Mechanik und Simulationstechnik an der Ruhr-Universität Bochum
seit 11/2005	CAE Ingenieur bei der TECOSIM in Rüsselsheim

Mitteilungen aus dem Institut für Mechanik

- Nr. 1 Theodor Lehmann: Dezember 1976
Große elasto-plastische Formänderungen
- Nr. 2 Bogdan Raniecki/Klaus Thermann: Juni 1978
Infinitesimal Thermoplasticity and Kinematics of Finite Elastic-Plastic Deformations. Basic Concepts
- Nr. 3 Wolfgang Krings: Januar 1976
Beitrag zur Finiten Element Methode bei linearem, viskoelastischem Stoffverhalten
- Nr. 4 Burkhard Lücke: Januar 1976
Theoretische und experimentelle Untersuchungen der zyklischen elastoplastischen Blechbiegung bei endlichen Verzerrungen
- Nr. 5 Knut Schwarze: Februar 1976
Einfluß von Querschnittsverformungen bei dünnwandigen Stäben mit stetig gekrümmter Profilmittellinie
- Nr. 6 Hubert Sommer: Januar 1977
Ein Beitrag zur Theorie des ebenen elastischen Verzerrungszustandes bei endlichen Formänderungen
- Nr. 7 H. Stumpf/F. J. Biehl: März 1977
Die Methode der orthogonalen Projektionen und ihre Anwendungen zur Berechnung orthotroper Platten
- Nr. 8 Albert Meyers: April 1977
Ein Beitrag zum optimalen Entwurf von schnelllaufenden Zentrifugenschalen
- Nr. 9 Berend Fischer: April 1977
Zur zyklischen, elastoplastischen Beanspruchung eines dickwandigen Zylinders bei endlichen Verzerrungen
- Nr. 10 Wojciech Pietraszkiewicz: Mai 1977
Introduction to the Non-Linear Theory of Shells
- Nr. 11 Wilfried Ullenboom: Juni 1977
Optimierung von Stäben unter nichtperiodischer dynamischer Belastung
- Nr. 12 Jürgen Guldenpfennig: Juli 1977
Anwendung eines Modells der Vielkristallplastizität auf ein Problem gekoppelter elastoplastischer Wellen
- Nr. 13 Pawel Rafalski: März 1978
Minimum Principles in Plasticity
- Nr. 14 Peter Hilgers: Juli 1978
Der Einsatz eines Mikrorechners zur hybriden Optimierung und Schwingungsanalyse
- Nr. 15 Hans-Albert Lauert: August 1979
Optimierung von Stäben unter dynamischer periodischer Beanspruchung bei Beachtung von Spannungsrestriktionen
- Nr. 16 Martin Fritz: Juli 1979
Berechnung der Auflagerkräfte und der Muskelkräfte des Menschen bei ebenen Bewegungen aufgrund von kinematographischen Aufnahmen
- Nr. 17 H. Stumpf/F. J. Biehl: Dezember 1979
Approximations and Error Estimates in Eigenvalue Problems of Elastic Systems with Application to Eigenvibrations of Orthotropic Plates

- Nr. 18 Uwe Kohlberg: Juli 1979
Variational Principles and their Numerical Application to Geometrically Nonlinear v. Karman Plates
- Nr. 19 Heinz Antes: Januar 1980
Über Fehler und Möglichkeiten ihrer Abschätzung bei numerischen Berechnungen von Schalenträgwerken
- Nr. 20 Czeslaw Wozniak: März 1980
Large Deformations of Elastic and Non-Elastic Plates, Shells and Rods
- Nr. 21 Maria K. Duszek: Juni 1980
Problems of Geometrically Non-Linear Theory of Plasticity
- Nr. 22 Burkhard von Bredow: Dezember 1980
Optimierung von Stäben unter stochastischer Erregung
- Nr. 23 Jürgen Preuss: Februar 1981
Optimaler Entwurf von Tragwerken mit Hilfe der Mehrzielmethode
- Nr. 24 Ekkehard Großmann: Februar 1981
Kovarianzanalyse mechanischer Zufallsschwingungen bei Darstellung der mehrfachkorrelierten Erregungen durch stochastische Differentialgleichungen
- Nr. 25 Dieter Weichert: März 1981
Variational Formulation and Solution of Boundary-Value Problems in the Theory of Plasticity and Application to Plate Problems
- Nr. 26 Wojciech Pietraszkiewicz: Juni 1981
On Consistent Approximations in the Geometrically Non-Linear Theory of Shells
- Nr. 27 Georg Zander: September 1981
Zur Bestimmung von Verzweigungslasten dünnwandiger Kreiszyylinder unter kombinierter Längs- und Torsionslast
- Nr. 28 Pawel Rafalski: September 1981
An Alternative Approach to the Elastic-Viscoplastic Initial-Boundary Value Problem
- Nr. 29 Heinrich Oeynhausen: November 1981
Verzweigungslasten elastoplastisch deformierter, dickwandiger Kreiszyylinder unter Innendruck und Axialkraft
- Nr. 30 F.-J. Biehl: Dezember 1981
Zweiseitige Eingrenzung von Feldgrößen beim einseitigen Kontaktproblem
- Nr. 31 Maria K. Duszek: Juni 1982
Foundations of the Non-Linear Plastic Shell Theory
- Nr. 32 Reinhard Piltner: Juli 1982
Spezielle finite Elemente mit Löchern, Ecken und Rissen unter Verwendung von analytischen Teillösungen
- Nr. 33 Petrisor Mazilu: Dezember 1982
Variationsprinzip der Thermoplastizität I. Wärmeausbreitung und Plastizität
- Nr. 34 Helmut Stumpf: Dezember 1982
Unified Operator Description, Nonlinear Buckling and Post-Buckling Analysis of Thin Elastic Shells
- Nr. 35 Bernd Kaempf: März 1983
Ein Exremal-Variationsprinzip für die instationäre Wärmeleitung mit einer Anwendung auf thermoelastische Probleme unter Verwendung der finiten Elemente

- Nr. 36 Alfred Kraft: Juli 1983
Zum methodischen Entwurf mechanischer Systeme im Hinblick auf optimales Schwingungsverhalten
- Nr. 37 Petrisor Mazilu: August 1983
Variationsprinzip der Thermoplastizität II. Gekoppelte thermomechanische Prozesse
- Nr. 38 Klaus-Detlef Mickley: November 1983
Punktweise Eingrenzung von Feldgrößen in der Elastomechanik und ihre numerische Realisierung mit Fundamental-Splinefunktionen
- Nr. 39 Lutz-Peter Nolte: Dezember 1983
Beitrag zur Herleitung und vergleichende Untersuchung geometrisch nichtlinearer Schalentheorien unter Berücksichtigung großer Rotationen
- Nr. 40 Ulrich Blix: Dezember 1983
Zur Berechnung der Einschnürung von Zugstäben unter Berücksichtigung thermischer Einflüsse mit Hilfe der Finite-Element-Methode
- Nr. 41 Peter Becker: Februar 1984
Zur Berechnung von Schallfeldern mit Elementmethoden
- Nr. 42 Diemar Bouchard: Februar 1984
Entwicklung und Anwendung eines an die Diskrete-Fourier-Transformation angepaßten direkten Algorithmus zur Bestimmung der modalen Parameter linearer Schwingungssysteme
- Nr. 43 Uwe Zdebel: Dezember 1984
Theoretische und experimentelle Untersuchungen zu einem thermo-plastischen Stoffgesetz
- Nr. 44 Jan Kubik: April 1985
Thermosdiffusion Flows in a Solid with a Dominant Constituent
- Nr. 45 Horst J. Klepp: Juni 1985
Über die Gleichgewichtslagen und Gleichgewichtsbereiche nichtlinearer autonomer Systeme
- Nr. 46 J. Makowski/L.-P. Nolte/H. Stumpf: Juli 1985
Finite In-Plane Deformations of Flexible Rods - Insight into Nonlinear Shell Problems
- Nr. 47 Franz Karl Labisch: August 1985
Grundlagen einer Analyse mehrdeutiger Lösungen nichtlinearer Randwertprobleme der Elastostatik mit Hilfe von Variationsverfahren
- Nr. 48 J. Chroscielewski/L.-P. Nolte: Oktober 1985
Strategien zur Lösung nichtlinearer Probleme der Strukturmechanik und ihre modulare Aufbereitung im Konzept MESY
- Nr. 49 Karl-Heinz Bürger: Dezember 1985
Gewichtsoptimierung rotationssymmetrischer Platten unter instationärer Erregung
- Nr. 50 Ulrich Schmid: Februar 1987
Zur Berechnung des plastischen Setzens von Schraubenfedern
- Nr. 51 Jörg Frischbier: März 1987
Theorie der Stoßbelastung ortotroper Platten und ihr experimentelle Überprüfung am Beispiel einer unidirektional verstärkten CFK-Verbundplatte
- Nr. 52 W. Tampczynski: Juli 1987
Strain history effect in cyclic plasticity
- Nr. 53 Dieter Weichert: Dezember 1987
Zum Problem geometrischer Nichtlinearitäten in der Plastizitätstheorie

- Nr. 54 Heinz Antes/Thomas Meise/Thomas Wiebe: Januar 1988
Wellenausbreitung in akustischen Medien Randelement-Prozeduren im 2-D Frequenzraum und im 3-D Zeitbereich
- Nr. 55 Wojciech Pietraszkiewicz: März 1988
Geometrically non-linear theories of thin elastic shells
- Nr. 56 Jerzy Makowski/Helmut Stumpf: April 1988
Finite strain theory of rods
- Nr. 57 Andreas Pape: Mai 1988
Zur Beschreibung des transienten und stationären Verfestigungsverhaltens von Stahl mit Hilfe eines nichtlinearen Grenzflächenmodells
- Nr. 58 Johannes Groß-Weege: Juni 1988
Zum Einspielverhalten von Flächentragwerken
- Nr. 59 Peihua LIU: Juli 1988
Optimierung von Kreisplatten unter dynamischer nicht rotationssymmetrischer Last
- Nr. 60 Reinhard Schmidt: August 1988
Die Anwendung von Zustandsbeobachtern zur Schwingungsüberwachung und Schadensfrüherkennung auf mechanische Konstruktionen
- Nr. 61 Martin Pitzer: Juli 1988
Vergleich einiger FE-Formulierungen auf der Basis eines inelastischen Stoffgesetzes
- Nr. 62 Jerzy Makowski/Helmut Stumpf: Dezember 1988
Geometric structure of fully nonlinear and linearized Cosserat type shell theory
- Nr. 63 O. T. Bruhns: Januar 1989
Große plastische Formänderungen - Bad Honnef 1988
- Nr. 64 Khanh Chau Le/Helmut Stumpf/Dieter Weichert: Juli 1989
Variational principles of fracture mechanics
- Nr. 65 Guido Obermüller: Juni 1989
Ein Beitrag zur Strukturoptimierung unter stochastischen Lasten
- Nr. 66 Herbert Diehl: Juni 1989
Ein Materialmodell zur Berechnung von Hochgeschwindigkeitsdeformationen metallischer Werkstoffe unter besonderer Berücksichtigung der Schädigung durch Scherbänder
- Nr. 67 Michael Geis: November 1989
Zur Berechnung ebener, elastodynamischer Rißprobleme mit der Randelementmethode
- Nr. 68 Günter Renker: November 1989
Zur Identifikation nichtlinearer strukturmechanischer Systeme
- Nr. 69 Berthold Schieck: November 1989
Große elastische Dehnungen in Schalen aus hyperelastischen inkompressiblen Materialien
- Nr. 70 Frank Szepan: Dezember 1989
Ein elastisch-viskoplastisches Stoffgesetz zur Beschreibung großer Formänderungen unter Berücksichtigung der thermomechanischen Kopplung
- Nr. 71 Christian Scholz: Dezember 1989
Ein Beitrag zur Gestaltoptimierung druckbelasteter Rotationsschalen
- Nr. 72 J. Badur/H. Stumpf: Dezember 1989
On the influence of E. and F. Cosserat on modern continuum mechanics and field theory

- Nr. 73 Werner Fornefeld: Januar 1990
Zur Parameteridentifikation und Berechnung von Hochgeschwindigkeitsdeformationen metallischer Werkstoffe anhand eines Kontinuums-Damage-Modells
- Nr. 74 J. Sączuk/H. Stumpf: April 1990
On statical shakedown theorems for non-linear problems
- Nr. 75 Andreas Feldmüller: April 1991
Ein thermoplastisches Stoffgesetz isotrop geschädigter Kontinua
- Nr. 76 Ulfert Rott: April 1991
Ein neues Konzept zur Berechnung viskoplastischer Strukturen
- Nr. 77 Thomas Heinrich Pingel: Juli 1991
Beitrag zur Herleitung und numerischen Realisierung eines mathematischen Modells der menschlichen Wirbelsäule
- Nr. 78 O. T. Bruhns: Dezember 1991
Große plastische Formänderungen - Bad Honnef 1991
- Nr. 79 J. Makowski/J. Chrosielewski/H. Stumpf:
Computational Analysis of Shells Undergoing Large Elastic Deformation Part I:Theoretical Foundations
- Nr. 80 J. Chrosielewski/J. Makowski/H. Stumpf:
Computational Analysis of Shells Undergoing Large Elastic Deformation Part II: Finite Element Implementation
- Nr. 81 R. H. Frania/H. Waller: Mai 1992
Entwicklung und Anwendung spezieller finiter Elemente für Kerbspannungsprobleme im Maschinenebau
- Nr. 82 B. Bischoff-Beiermann: Juli 1992
Zur selbstkonsistenten Berechnung von Eigenspannungen in polykristallinem Eis unter Berücksichtigung der Monokristallanisotropie
- Nr. 83 J. Pohé: Februar 1993
Ein Beitrag zur Stoffgesetzentwicklung für polykristallines Eis
- Nr. 84 U. Kikillus: Mai 1993
Ein Beitrag zum zyklischen Kriechverhalten von Ck 15
- Nr. 85 T. Guo: Juni 1993
Untersuchung des singulären Rißspitzenfeldes bei stationärem Rißwachstum in verfestigendem Material
- Nr. 86 Achim Menne: Januar 1994
Identifikation der dynamischen Eigenschaften von hydrodynamischen Wandlern
- Nr. 87 Uwe Folchert: Januar 1994
Identifikation der dynamischen Eigenschaften Hydrodynamischer Kopplungen
- Nr. 88 Jörg Körber: April 1994
Ein verallgemeinertes Finite-Element-Verfahren mit asymptotischer Stabilisierung angewendet auf viskoplastische Materialmodelle
- Nr. 89 Peer Schieße: April 1994
Ein Beitrag zur Berechnung des Deformationsverhaltens anisotrop geschädigter Kontinua unter Berücksichtigung der thermoplastischen Kopplung
- Nr. 90 Egbert Schopphoff: Juli 1994
Dreidimensionale mechanische Analyse der menschlichen Wirbelsäule

- Nr. 91 Christoph Beerens: Juli 1994
Zur Modellierung nichtlinearer Dämpfungsphänomene in der Strukturmechanik
- Nr. 92 K. C. Le/H. Stumpf: November 1994
Finte elastoplasticity with microstructure
- Nr. 93 O. T. Bruhns: Dezember 1994
Große plastische Formänderungen - Bad Honnef 1994
- Nr. 94 Armin Lenzen: Dezember 1994
Untersuchung von dynamischen Systemen mit der Singulärwertzerlegung - Erfassung von Strukturveränderungen
- Nr. 95 J. Makowski/H. Stumpf: Dezember 1994
Mechanics of Irregular Shell Structures
- Nr. 96 J. Chroscielewski/J. Makowski/H. Stumpf: Dezember 1994
Finte Elements for Irregular Nonlinear Shells
- Nr. 97 W. Krings/A. Lenzen/u. a.: Februar 1995
Festschrift zum 60. Geburtstag von Heinz Waller
- Nr. 98 Ralf Podleschny: April 1995
Untersuchung zum Instabilitätsverhalten scherbeanspruchter Risse
- Nr. 99 Bernd Westerhoff: Juli 1995
Eine Untersuchung zum geschwindigkeitsabhängigen Verhalten von Stahl
- Nr. 100 Marc Mittelbach: Dezember 1995
Simulation des Deformations- und Schädigungsverhaltens beim Stoßversuch mit einem Kontinuums-Damage-Modell
- Nr. 101 Ulrich Hoppe: Mai 1996
Über grundlegende Konzepte der nichtlinearen Kontinuumsmechanik und Schalentheorie
- Nr. 102 Marcus Otto: Juni 1996
Erweiterung des Kaustikenverfahrens zur Analyse räumlicher Spannungskonzentrationen
- Nr. 103 Horst Lanzerath: Juli 1996
Zur Modalanalyse unter Verwendung der Randelementemethode
- Nr. 104 Andreas Wichtmann: August 1996
Entwicklung eines thermodynamisch konsistenten Stoffgesetzes zur Beschreibung der Reckalterung
- Nr. 105 Bjarne Fossa: Oktober 1996
Ein Beitrag zur Fließflächenmessung bei vorgedehnten Stoffen
- Nr. 106 Khanh Chau Le: Dezember 1996
Kontinuumsmechanisches Modellieren von Medien mit veränderlicher Mikrostruktur
- Nr. 107 Holger Behrens: Januar 1997
Nichtlineare Modellierung und Identifikation hydrodynamischer Kupplungen mit allgemeinen diskreten Modellansätzen
- Nr. 108 Johannes Moosheimer: Juli 1997
Gesteuerte Schwingungsdämpfung mit Elektrorheologischen Fluiden
- Nr. 109 Dirk Klaus Anding: Oktober 1997
Zur simultanen Bestimmung materialabhängiger Koeffizienten inelastischer Stoffgesetze
- Nr. 110 Stephan Weng: Dezember 1997
Ein Evolutionsmodell zur mechanischen Analyse biologischer Strukturen

- Nr. 111 Michael Straßberger: Dezember 1997
Aktive Schallreduktion durch digitale Zustandsregelung der Strukturschwingungen mit Hilfe piezokeramischer Aktoren
- Nr. 112 Hans-Jörg Becker: Dezember 1997
Simulation des Deformationsverhaltens polykristallinen Eises auf der Basis eines monokristallinen Stoffgesetzes
- Nr. 113 Thomas Nerzak: Dezember 1997
Modellierung und Simulation der Ausbreitung adiabatischer Scherbänder in metallischen Werkstoffen bei Hochgeschwindigkeitsdeformationen
- Nr. 114 O. T. Bruhns: März 1998
Große plastische Formänderungen
- Nr. 115 Jan Steinhausen: August 1998
Die Beschreibung der Dynamik von Antriebssträngen durch Black-Box-Modelle hydrodynamischer Kupplungen
- Nr. 116 Thomas Pandorf: August 1998
Experimentelle und numerische Untersuchungen zur Kerbspitzenbeanspruchung bei schlagbelasteten Biegeproben
- Nr. 117 Claus Oberste-Brandenburg: Juni 1999
Ein Materialmodell zur Beschreibung der Austenit-Martensit Phasentransformation unter Berücksichtigung der transformationsinduzierten Plastizität
- Nr. 118 Michael Märtens: Dezember 1999
Regelung mechanischer Strukturen mit Hilfe piezokeramischer Stapelaktoren
- Nr. 119 Dirk Kamarys: Dezember 1999
Detektion von Systemveränderungen durch neue Identifikationsverfahren in der experimentellen Modalanalyse
- Nr. 120 Wolfgang Hiese: Januar 2000
Gültigkeitskriterien zur Bestimmung von Scherbruchzähigkeiten
- Nr. 121 Peter Jaschke: Februar 2000
Mathematische Modellierung des Betriebsverhaltens hydrodynamischer Kupplungen mit hybriden Modellansätzen
- Nr. 122 Stefan Müller: Februar 2000
Zum Einsatz von semi-aktiven Aktoren zur optimalen Schwingungsreduktion in Tragwerken
- Nr. 123 Dirk Eichel: Juni 2000
Zur Kondensation strukturdynamischer Aufgaben mit Hilfe von Polynommatrizen
- Nr. 124 Andreas Bürgel: August 2000
Bruchmechanische Kennwerte beim Wechsel im Versagensverhalten dynamisch scherbeanspruchter Risse
- Nr. 125 Daniela Lürding: März 2001
Modellierung großer Deformationen in orthotropen, hyperelastischen Schalenstrukturen
- Nr. 126 Thorsten Quent: Mai 2001
Ein mikromechanisch begründetes Modell zur Beschreibung des duktilen Verhaltens metallischer Werkstoffe bei endlichen Deformationen unter Berücksichtigung von Porenschädigung
- Nr. 127 Ndzi C. Bongmba: Mai 2001
Ein finites anisotropes Materialmodell auf der Basis der Hencky-Dehnung und der logarithmischen Rate zur Beschreibung duktiler Schädigung

- Nr. 128 Henning Schütte: August 2001
Ein finites Modell für spröde Schädigung basierend auf der Ausbreitung von Mikrorissen
- Nr. 129 Henner Vogelsang: Dezember 2001
Parameteridentifikation für ein selbstkonsistentes Stoffmodell unter Berücksichtigung von Phasentransformationen
- Nr. 130 Jörn Mosler: Dezember 2002
Finite Elemente mit sprungstetigen Abbildungen des Verschiebungsfeldes für numerische Analysen lokalisierter Versagenszustände
- Nr. 131 Karin Preusch: Mai 2003
Hierarchische Schalenmodelle für nichtlineare Kontinua mit der p-Version der Finite-Element Methode
- Nr. 132 Christoph Müller: August 2003
Thermodynamic modeling of polycrystalline shape memory alloys at finite strains
- Nr. 133 Martin Heiderich: Juni 2004
Ein Beitrag zur zerstörungsfreien Schädigungsanalyse
- Nr. 134 Raoul Costamagna: Juli 2004
Globale Materialbeziehungen für das geklüftete Gebirge
- Nr. 135 Markus Böl: Januar 2005
Numerische Simulation von Polymernetzwerken mit Hilfe der Finite-Elemente-Methode
- Nr. 136 Gregor Kotucha: August 2005
Regularisierung von Problemen der Topologieoptimierung unter Einbeziehung von Dichtegradienten
- Nr. 137 Michael Steiner: Februar 2006
Deformations- und Versagensverhalten innendruckbeanspruchter Stahlrohre durch Stoßbelastung
- Nr. 138 Dirk Bergmannshoff: Dezember 2006
Das Instabilitätsverhalten zug-/scherbeanspruchter Risse bei Variation des Belastungspfad
- Nr. 139 Olaf Schilling: Januar 2007
Über eine implizite Partikelmethode zur Simulation von Umformprozessen

**Mitteilungen aus dem Institut für Mechanik
RUHR-UNIVERSITÄT BOCHUM
Nr. 139**

978-3-935892-14-8