

**Fourierreihen zur Berechnung
repräsentativer Volumenelemente
mit Mikrostruktur**

Von der Fakultät V der TU Berlin
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
genehmigte Dissertation

von Sven Kaßbohm

Promotionsausschuss

Vorsitzender:	Prof. Dr.-Ing. G. Brunk
1. Gutachter:	Prof. Dr. rer. nat. W.H. Müller
2. Gutachter:	Prof. Dr.-Ing. G. Silber
Wissenschaftliche Aussprache am:	17.5.2006

Berlin 2006
D 83

Symbolverzeichnis

Symbole ohne Index (kleine griechische Buchstaben)

α_i	Abminderungsfaktoren, Seite 36
α_T	Wärmeausdehnungskoeffizient, Seite 42
$\lambda, \mu, \kappa, \alpha, \beta, \gamma$	Materialkonstanten für isotrope Cosserat-Kontinua, Seite 19
ϱ	Materialkonstante $\varrho = \lambda + (2\mu + \kappa)$, Seite 19
E	Elastizitätsmodul, Seite 42
G	Schubmodul, Seite 42
ι	komplexe Einheit, $\iota^2 = -1$, Seite 28
ν	Querkontraktionszahl, Seite 42

Symbole mit einem Index (lateinische Buchstaben)

a	dimensionsloser diskreter Ortsvektor $a_i = n_i/N_{(i)}$, Seite 30
a	dimensionsloser kontinuierlicher Ortsvektor $0 \leq a_i < 1$, Seite 31
f^\wedge	Funktion aller Fourierkoeffizienten von f , Seite 28
f_k^\wedge	(komplexe) Zahlenfolge, Seite 30
f_n	(reelle) Zahlenfolge, Seite 30
k, l	(Kreis-)Wellenvektoren, z.B. $k = (-2/L_1, 4/L_2, 1/L_3)$, Seite 28
k^L	dimensionsloser Wellenvektor, z.B. $k^L = (-2, 4, 1)$, Seite 30
L	L_i sind Längen, Seite 28
m	Diskretisierung $m_i \in \mathbb{N}$, Seite 30
N	Diskretisierung $N_i = 2m_i + 1$, Seite 30

\tilde{p}	Störung, Seite 52
$p(\tilde{X})$	Lage eines Teilchens, Element von \mathbf{E} , Seite 10
\tilde{q}	Störung, Seite 52
$q(\tilde{X})$	Ordnung eines Teilchens, Element von \mathbf{M} , Seite 10
\tilde{r}	verallgemeinerte Störung $\tilde{r} = (\tilde{p}, \tilde{q})$, Seite 24
u	Verschiebung, Seite 19
v	axialer Vektor der Drehung, Seite 19
w	verallgemeinerte Verschiebung $w = (u, v)$ (geordnetes Paar), Seite 17
\tilde{X}	Teilchen, Element von \mathbf{B} , Seite 10
x	Ortsvektor, Seite 24
$x(\tilde{X})$	Ortvektor eines Teilchens, Seite 10
Symbole mit zwei Indizes (kleine griechische fette Buchstaben)	
δ	Kroneckersymbol, Seite 20
ε	Verzerrung, Seite 20
ε_0	(örtlicher) Mittelwerte von ε_0 , Seite 21
γ	verallgemeinerte Verzerrung $\gamma = (\varepsilon, \kappa)$ (geordnetes Paar), Seite 19
κ	Verkrümmung, Seite 20
μ	Momentenspannung, Paarspannung, Seite 19
τ	Kraftspannung, Seite 19
τ^*	Eigen(Kraft-)spannungen, Seite 19
ω	Koeffizientenmatrix, Seite 52

Symbole mit drei Indizes (kleine lateinische fette Buchstaben)

e Permutationssymbol, Seite 19

$\bar{g}^\epsilon, \bar{g}^\kappa$ Lösungsoperatoren im Fourierraum, Seite 54

Operatoren

$(\boxtimes, \boxtimes, \boxtimes)$ geordnetes Tripel, Seite 2

(\boxtimes, \boxtimes) geordnetes Paar, Seite 17

$\langle \boxtimes, \boxtimes \rangle$ inneres Produkt, Seite 29

$\langle \boxtimes \rangle$ (örtlicher) Mittelwert von \boxtimes , Seite 21

\boxtimes^T Transponierter Tensor zu \boxtimes , Seite 60

$\text{sp } \boxtimes$ Spur des Tensors \boxtimes , Seite 60

$\mathcal{F}^{d-1}\{\boxtimes\}$ Operator, der die Folge oder die Interpolationsfunktion aus Fourierkoeffizienten ermittelt, Seite 31

$\mathcal{F}^d\{\boxtimes\}$ Operator, der Fourierkoeffizienten des trigonometrischen Interpolationspolynoms der niedrigsten Ordnung durch die Punkte der Zahlenfolge \boxtimes ermittelt, Seite 30

$\mathcal{F}^{-1}\{\boxtimes\}$ Operator, der die Funktion selbst aus ihren Fourierkoeffizienten ermittelt, Seite 28

$\mathcal{F}\{\boxtimes\}$ Operator, der Fourierkoeffizienten der Funktion \boxtimes ermittelt, Seite 25

\bar{G} Lösungsoperator aus dem Problem mit konstanten Koeffizienten, Seite 24

I Operator der identischen Abbildung, Seite 51

$L^\gamma, \tilde{L}^\gamma, \bar{L}^\gamma$ Differentialoperatoren für Problem formuliert in γ , Seite 24

$L^w, \tilde{L}^w, \bar{L}^w$ Differentialoperatoren für Problem formuliert in w , Seite 24

$L^{\gamma w}$ Differentialoperator für Verschiebungs-Verzerrungs-Gln., Seite 20

$Q^N\{\boxtimes\}$ Näherung der Funktion \boxtimes basierend auf einer N -Diskretisierung, Seite 49

Sonstige Symbole

B Körper, Seite 10

E dreidimensionaler Euklidischer Punktraum, Seite 10

$GL(3)$ allgemeine lineare Gruppe: Liegruppe der linearen Abbildungen mit positiver Determinante, Seite 12

M Mannigfaltigkeit, Seite 10

$SO(3)$ spezielle orthogonale Gruppe: Liegruppe der Drehmatrizen mit Determinante 1, Seite 12

S^1 1-Sphäre, Oberfläche einer zweidimensionalen Kugel mit Radius 1, Seite 12

S^2 2-Sphäre, Oberfläche einer dreidimensionalen Kugel mit Radius 1, Seite 12

\boxtimes Platzhalter, beliebiger Operand, Seite 55

Hinweis zur Rechnung mit Indizes

Lateinische Indizes durchlaufen die Werte 1, 2, 3. Griechische Indizes durchlaufen die Werte 1, 2, 3, 4, 5, 6. Über doppelt auftauchende Indizes wird summiert.

INHALTSVERZEICHNIS

Inhaltsverzeichnis

Inhaltsverzeichnis	6
Abbildungsverzeichnis	7
Tabellenverzeichnis	8
1 Einleitung	9
1.1 Grundlagen, Spezialisierung und Ziel der Arbeit	9
1.1.1 Grundlagen	9
1.1.2 Spezialisierung und Ziel	13
1.2 Gliederung der Arbeit	15
2 Problemstellung und Lösungsweg	17
2.1 Problemstellung	17
2.1.1 Differentialgleichungen	17
2.1.2 Randbedingungen	20
2.1.3 Lasten	21
2.2 Lösungsweg	22
2.2.1 Implizite Lösung im Realraum	24
2.2.2 Explizite Lösung im Fourierraum	25
3 Numerische Behandlung mit Fourierreihen	27
3.1 Fourierkoeffizienten und Fourierreihe	27
3.1.1 Orthogonalitätsrelationen	28
3.2 Diskrete Fourier Transformation	30
3.3 Faltung	31
3.3.1 Periodische Faltung	31
3.3.2 Diskrete periodische Faltung	32
3.4 Zusammenfassung	32
4 Beispiele	35
4.1 Zwei akademische Beispiele	35
4.1.1 Beschreibung der akademischen Beispiele	35
4.1.2 Diskretisierung	35
4.1.3 Ergebnisse für die akademischen Beispiele	39
4.2 Ein praktisches Beispiel	39

ABBILDUNGSVERZEICHNIS

4.2.1	Beschreibung des praktischen Beispiels	39
4.2.2	Diskretisierung	44
4.2.3	Ergebnisse für das praktische Beispiel	44
4.3	Grenzen und Größenordnung der Materialwerte	46
5	Zusammenfassung und Ausblick	48
5.1	Zusammenfassung	48
5.2	Ausblick	48
A	Anhang	50
A.1	Abminderungsfaktoren	50
A.2	Neumann Reihe	52
A.3	Berechnung von $\bar{g} = \mathcal{F}\{\bar{G}\}$	53
A.3.1	Berechnung von ω	53
A.3.2	Berechnung von \bar{g}^ϵ und \bar{g}^κ	54
A.4	Berechnung von $\mathcal{F}\{L^{\gamma m-1}\gamma\}$	56
A.5	Rekursionsformel mit $\mathcal{F}\{\bar{G}\}\mathcal{F}\{L^{\gamma m-1}\gamma\}$	59
A.6	Fehler im Bildbereich	59
B	Fortran Programme	61
B.1	Struktogramm	61
B.2	Hauptprogramm prog	63
B.3	Weitere Programmteile	71
B.4	Unterprogramm bel	74
B.5	Unterprogramm verz	77
B.6	Unterprogramm falt	79
B.7	Unterprogramm fehl	80
B.8	Unterprogramm verb	83
B.9	Unterprogramme fk_s und fk	84
B.10	Unterprogramm fr	87
	Literaturverzeichnis	90
	Index	94
	Abbildungsverzeichnis	
1	Konfigurationsraum des Kontinuums m. Mikrostruktur .	10

TABELLENVERZEICHNIS

2	Erweiterung zum Kontinuum mit Mikrostruktur	11
3	Mathematisches Problem	18
4	Repräsentatives Volumenelement	21
5	Schematische Darstellung des Lösungswegs	23
6	Stellung der numerischen Behandlung im Lösungsweg	34
7	„Lokal isotropes“ RVE: Weiche Faser in steifer Matrix	37
8	Diskretisierung für $m_1 \times m_2 = 2 \times 2$	38
9	λ entlang eines Schnitts für akademische Beispiele	39
10	λ ohne Abminderungsfaktoren	40
11	λ mit Abminderungsfaktoren	40
12	Akademisches Beispiel 1: Spannungen	41
13	Akademisches Beispiel 2: Spannungen	42
14	Praktisches Beispiel: Steifigkeitsverteilung	44
15	λ entlang eines Schnitts für praktisches Beispiel	45
16	Praktisches Beispiel: Spannungen	47
17	Übersicht Programmablauf	62

Tabellenverzeichnis

1	Beispiele für Kontinua mit Mikrostruktur	12
2	Beschreibung der akademischen Beispiele	36
3	klassische Materialwerte für Zinn (Sn) und Blei (Pb)	43
4	„Cosserat“-Materialwerte für ausgewählte Stoffe	46

1 Einleitung

1.1 Grundlagen, Spezialisierung und Ziel der Arbeit

1.1.1 Grundlagen

Das Kontinuum mit Mikrostruktur ist eine Verallgemeinerung des herkömmlichen Kontinuums, das in den verschiedensten Bereichen der Physik - insbesondere der rationalen Mechanik - zur Beschreibung von Prozessen verwendet wurde und wird. Im Gegensatz zum herkömmlichen Kontinuum wird beim Kontinuum mit Mikrostruktur zusätzlich eine „Ordnung“ auf der atomaren Ebene berücksichtigt, die für das Verhalten eines Körpers von entscheidender Bedeutung ist. Dies ist nun bei einer vielfältigen Reihe von Materialien der Fall, aus denen Körper bestehen können. Beispiele hierfür sind Stoffe mit näherungsweise starren und stäbchenartigen Strukturen im Inneren, Blutplasma, granulare Stoffe, Festkörper mit kontinuierlich verteilten Versetzungen, Flüssigkeiten mit Bläschen, Flüssigkristalle usw. .

Das Materialverhalten hängt beim Kontinuum mit Mikrostruktur u.U. also auch von anderen als den observablen Größen wie der Bewegung oder Verschiebung ab. Unter einer einzigen bestimmten Bewegung kann das Material ganz verschiedene Zustände annehmen, weil die atomare Ordnung auch ganz verschieden sein kann. Aus diesem Grund wird der Konfigurationsraum des herkömmlichen Kontinuums passend zu den Eigenschaften des Materials erweitert. Aus Gründen der praktischen Umsetzung wird der spezielle atomare Zustand durch ein Element einer Mannigfaltigkeit¹ dargestellt (siehe Abbildung 1).

Welche „Art“ von Mannigfaltigkeit für den Stoff bzw. Körper „passend“ ist, hängt natürlich von der Art der Eigenschaften ab, die das spezielle Stoffverhalten ausmachen. Beispiele sind in Tabelle 1 dargestellt. Der Vorteil der moderneren Beschreibung eines Körpers als Kontinuum mit Mikrostruktur liegt darin, dass man sich die Vorteile der Kontinuumstheorien bewahrt (Anwendung der Differential- und Integralrechnung etc.), gleichzeitig aber wesentlich mehr Möglichkeiten zur Beschreibung des Materialverhaltens eröffnet. Insbesondere bietet sich die Möglichkeit, das Materialverhalten völlig verschiedenartiger Stoffe verschiedener Aggregatzustände in einem einheitlichen Rahmen (nämlich dem des Kontinuums mit Mikrostruktur) zu untersuchen.

¹Näheres zur Differentialgeometrie findet man z.B. in [Spivak 70].

1 EINLEITUNG

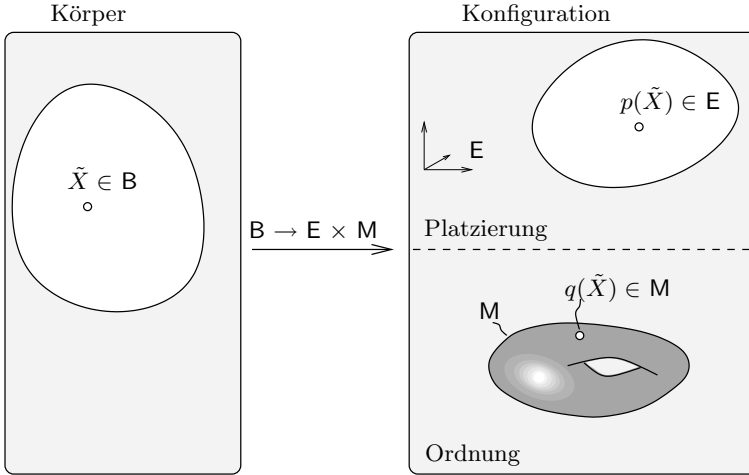


Abbildung 1: Konfigurationsraum des Kontinuums mit Mikrostruktur, der „Zustand“ eines Teilchens \tilde{X} ist festgelegt durch p und q , also durch die Lage und die Ordnung von \tilde{X} . Die Menge aller Zustände, die alle Teilchen zu jeder Zeit annehmen können, bilden den Konfigurationsraum.

Das Forschungsgebiet Kontinua mit Mikrostruktur beinhaltet Aspekte einer makroskopischen Theorie, aber auch solche aus mikroskopischen Theorien. Damit ist es möglich, Eigenschaften verschiedener Größenskalen abzubilden. Die Konsequenzen, die eine Erweiterung der Theorien für gewöhnliche Kontinua auf Theorien für Kontinua mit Mikrostruktur mit sich bringt, sind in Abbildung 2 dargestellt. Der Versuch, das Verhalten eines Stoffes unter Berücksichtigung irgendeiner atomaren Ordnung (Ordnungsparameter) zu beschreiben hat eine lange Tradition. Für Festkörper und Fluide sind in [Cosserat 09, Eringen 68, Toupin 64, Mindlin 62, Mindlin 64, Mindlin 65, Green 64] verschiedene Ansätze zur Berücksichtigung der atomaren Ordnung für Festkörper und Fluide gegeben. Neben der schon aufgezeigten Möglichkeit der Erweiterung des Konfigurationsraums wurden auch sogenannte Gradiententheorien entwickelt. Hierbei wird das „Empfinden“ eines Stoffes da-

1.1 GRUNDLAGEN, SPEZIALISIERUNG UND ZIEL DER ARBEIT

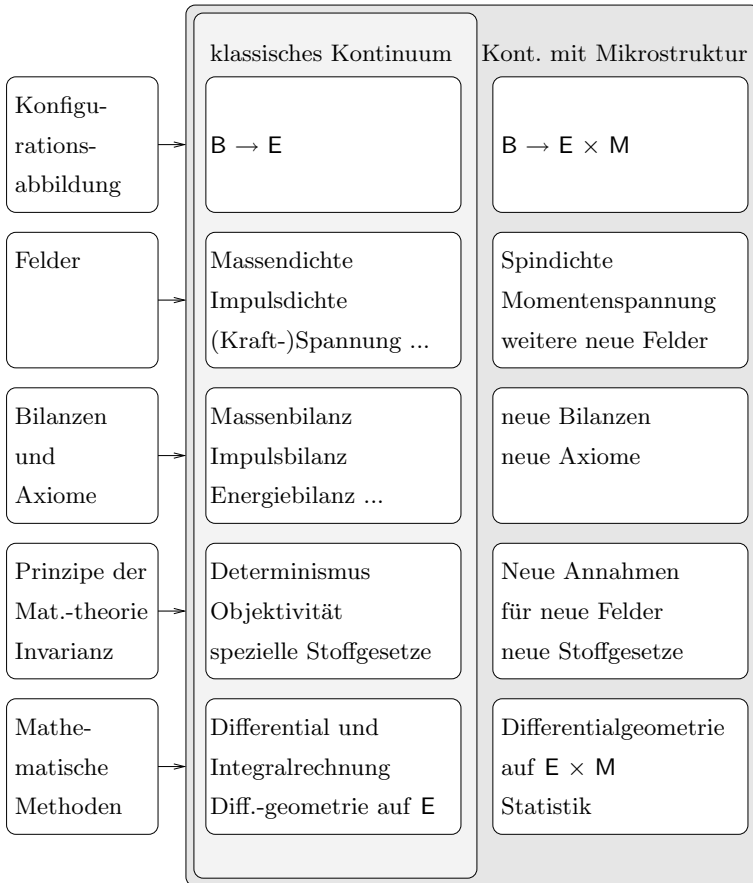


Abbildung 2: Erweiterung des herkömmlichen klassischen Kontinuums auf das Kontinuum mit Mikrostruktur

1 EINLEITUNG

„Charakter“ des Kontinuums	Bedeutung von $q \in M$	Mannigfaltigkeit M
porös, mit kugelförmigen Löchern	Volumenanteil der Löcher	$[0, \frac{\sqrt{3}\pi}{8})$
flüssig mit Bläschen	Volumenanteil der Bläschen	$[0, 1]$
mit ebenem Spin	Lage/Orientierung eines Einheitsvektors in einer Ebene	$[0, 2\pi)$ oder S^1
mit Spin	Lage/Orientierung eines Einheitsvektors	S^2
„Cosserat-artig“	Lage eines Dreibeins aus orthogonalen Einheitsvektoren	$SO(3)$
mikromorph	Lage eines deformierbaren Ellipsoids	$GL(3)^+$

Tabelle 1: Beispiele für Kontinua mit Mikrostruktur, siehe auch [Capriz 89]

durch erweitert, dass man die Einflussumgebung durch Berücksichtigung höherer Gradienten vergrößert. Der Konfigurationsraum kann dabei der des herkömmlichen Kontinuums bleiben. Gemischte Theorien (Erweiterung des Konfigurationsraums und Erweiterung der Einflussumgebung) sind auch möglich. Eine systematische Einordnung verschiedener (auch gemischter) Theorien ist in [Trostel 80] erläutert. Darin wird unterschieden zwischen einerseits der Erweiterung der Einflussumgebung in dem Sinne, dass höhere Gradienten der Bewegung berücksichtigt werden und andererseits der Erweiterung der Kinematik in dem Sinne, dass der Konfigurationsraum vergrößert wird.

Die Theorien - insbesondere die, bei denen der Konfigurationsraum erweitert wird - kommen seit einigen Jahren zu einer neuen Blüte. Inzwischen werden auch Methoden der statistischen Physik mit den Theorien für Kontinua mit Mikrostruktur gekoppelt, was einer weiteren Ver-

1.1 GRUNDLAGEN, SPEZIALISIERUNG UND ZIEL DER ARBEIT

allgemeinerung entspricht. Hierbei wird jedem Kontinuumselement eine bestimmte statistische Verteilung des atomaren Zustands zugeordnet (siehe hierzu [Svendsen 99, Svendsen 01]).

Beim der Untersuchen von Kontinua mit Mikrostruktur gelangt man zu für die Kontinuumsmechanik neuen Fragestellungen (siehe auch Abbildung 2): Welche zusätzlichen Axiome analog zu den Newtonschen Axiomen der klassischen Mechanik und deren Erweiterungen und Ergänzungen für die Kontinuumsmechanik sollten gefordert werden? Wie lassen sich spezielle Stoffgesetze finden? Welche Größen sind invariant gegen welche speziellen Transformationen? Hierzu gibt es verschiedene Ansätze [Capriz 89, Svendsen 99, Svendsen 01, Muschik 01, Muschik 04], die hier nicht im Einzelnen diskutiert werden sollen. Stattdessen wird aus der großen Menge aller möglichen Themen in diesem Problemkreis eine Spezialisierung hin zu einer praxisorientierten Problemstellung gegeben.

1.1.2 Spezialisierung und Ziel, Einbettung in aktuelle Forschung

In dieser Arbeit wird es speziell um Kontinua mit Mikrostruktur aus der Klasse der Festkörper gehen und hierbei weiter um die Verzerrungs- und Spannungsberechnung in repräsentativen Volumenelementen. Alle Berechnungen werden im Rahmen der linearen Elastizitätstheorie durchgeführt, so dass die untersuchten Differentialgleichungssysteme linear sind. Es wird gezeigt werden, dass alle Kontinua mit Mikrostruktur dieser Klasse (lineare Elastizitätstheorie und periodische Randbedingungen) mit derselben Prozedur behandelt werden können. Beispielfhaft werden drei verschiedene Kontinua mit Mikrostruktur untersucht: Das Cosserat-Kontinuum, das Kontinuum mit Momentenspannungen (couple-stress-theory) und das klassische Kontinuum, bei dem zwar keine Mikrostruktur vorhanden ist, das aber trotzdem im selben Rahmen untersucht werden kann. Sowohl das Couple-Stress-Kontinuum, als auch das klassische Kontinuum lassen sich als Spezialfälle des Cosserat-Kontinuums ableiten.

Trotz dieses engen Korsetts an Voraussetzungen bzw. trotz des hohen Grades an Spezialisierung, können die Formeln und die Rechenergebnisse für eine Vielzahl von Anwendungen genutzt werden, da die Spannungsberechnung in repräsentativen Volumenelementen eine vielfach benutzte Methode darstellt. Solche Anwendungen sind z.B. Homo-

1 EINLEITUNG

genisierung, Modellierung der verzerrungsabhängigen Vergrößerung in repräsentativen Volumenelementen und vieles mehr.

Die vorgestellte Methode zur Lösung der Randwertprobleme mit Fourierreihen (siehe Abschnitt 3.4) bietet gegenüber anderen Methoden besondere Vorteile. Zum einen ist die Methode besonders übersichtlich, so dass nur kurze Zeit für das Verständnis und für die Einarbeitung in die Programme nötig ist (siehe Abschnitt B). Dies ist für die weitere Benutzung der Programme sehr hilfreich. Zum anderen lässt sich der Fehler (siehe Abschnitt A.6) exakt ermitteln, was bei anderen Verfahren (z.B. bei der Methode der Finiten Elemente) unmöglich ist (siehe Abschnitt A.6).

Das klassische Kontinuum wurde bereits in [Kaßbohm 05a] untersucht. Eine ähnliche Behandlungsmethode für das klassische Kontinuum findet man in [Müller 96, Moulinec 98, Moulinec 94, Eyre 99]. Dieselbe Problemstellung (teils auch für Cosserat-Kontinua), mit der Methode der Finiten Elemente behandelt, liest man z.B. in [Kanit 03, Forest 99, Forest 00, Cailletaud 03] nach. Ähnliche Problemstellungen werden von [Chang 05, Oden 70, Nakamura 84, Padovan 78] mit der Methode der Finiten Elemente untersucht. Asymptotische Verfahren für Cosserat-Kontinua werden z.B. in [Forest 01] behandelt.

Obwohl die Berechnung repräsentativer Volumenelemente mit Fourierreihen äußerst naheliegend ist, wurde die Erweiterung auf Kontinua mit Mikrostruktur bisher nicht durchgeführt. Das wesentliche Ziel der Arbeit ist es aufzuzeigen, dass die vorgestellte Methode unter den gegebenen Voraussetzungen zur Berechnung repräsentativer Volumenelemente ein universelles Werkzeug ist. Alle Kontinua mit Mikrostruktur in der linearen Elastizitätstheorie können bei periodischen Randbedingungen mit der vorgestellten Methode untersucht werden. Ein weiterer wesentlicher Bestandteil der Arbeit ist die Verbesserung und Erweiterung der Verfahren von [Müller 96, Moulinec 98, Moulinec 94, Eyre 99]. Es werden folgende Verbesserungen beschrieben:

- Zwei Fehler, die bei einer numerischen Berechnung eines vorgelegten Problems auftauchen, werden unterschieden. Es werden Wege aufgezeigt, wie sich einer der Fehler durch Abminderungsfaktoren reduzieren lässt.
- Durch die Ausnutzung der periodischen Faltung kann der Rechenaufwand gegenüber den bisher benutzten Verfahren deutlich

verringert werden. Das Verfahren wird dadurch auch übersichtlicher.

- Die bekannten Verfahren sind vom zweidimensionalen Fall (bei dem ungeklärt ist, ob es sich um einen ebenen Spannungs- oder einen ebenen Verzerrungszustand handeln sollte) verallgemeinert worden auf den Fall eines dreidimensionalen repräsentativen Volumenelements.
- Durch die Behandlung mit Fourierreihen können die Funktionen an jeder Stelle im repräsentativen Volumenelement ausgewertet werden, und nicht nur an den bei der diskreten Fourier Transformation benutzten Stellen. Alle Ableitungen können exakt ermittelt werden.

Ein weiteres Ziel der Arbeit ist, ein dokumentiertes Programmpaket bereitzustellen, mit dem praktische Problemstellungen bearbeitet werden können, die am Institut für Mechanik der TU Berlin (Fachgebiet Kontinuumsmechanik und Materialtheorie) auftauchen. Um die Anwendbarkeit und Erweiterbarkeit der Programme zu gewährleisten, ist die Programmierung mit der sehr einfachen und für die Anforderungen ausreichenden Programmiersprache Fortran77 umgesetzt.

1.2 Gliederung der Arbeit

Da insbesondere das Cosserat-Kontinuum inzwischen vielfache Anwendungen gefunden hat (siehe hierzu [Lakes 81, Lakes 82, Nakamura 95, Lakes 95]), wird in dieser Arbeit das Cosserat Kontinuum zuerst untersucht. Zwei Spezialfälle lassen sich daraus unmittelbar ableiten: Das klassische Kontinuum und das Couple-Stress Kontinuum. Die entscheidenden Gleichungen werden daher im folgenden Abschnitt 2 immer blockweise für alle drei Typen angegeben. Am Ende des Abschnitts 2 liegen alle Gleichungen vor, die man zur analytischen Lösung des Problems benötigt.

Die benötigten Lösungsoperatoren können nur im Fourierraum berechnet werden, und daher wird das Problem in den Fourierraum übertragen mit der Folge, dass von bestimmten Eingangswerten die Fourierkoeffizienten ermittelt werden müssen. Diese Berechnung erfolgt numerisch und basiert auf einer Diskretisierung. Grundlage bietet die Theo-

1 EINLEITUNG

rie der Fourierreihen und die diskrete Fourier Transformation. Auf diese Themen wird in Abschnitt 3.1 eingegangen.

In Abschnitt 4 wird die Methode anhand von zwei akademischen Beispielen illustriert. Dabei wird auch die Diskretisierung genau beschrieben. Nachfolgend wird als ein praktisches Problem ein Beispiel aus [Müller 04] untersucht.

Schließlich werden in Abschnitt 5 die Ergebnisse diskutiert und zusammengefasst. Einige mögliche Wege zur Erweiterung oder Verbesserung des Verfahrens werden aufgezeigt. Zusätzlich werden Vorschläge gegeben, in welche Richtung eine weiterführende Forschung laufen könnte.

Im Anhang A werden einige Formeln gezeigt, die bei der Problem- und Lösungsbeschreibung (Abschnitt 2) aus Gründen der Übersichtlichkeit übergangen werden. Insbesondere wird die Berechnung der Lösungsoperatoren dort beschrieben. kann. Im Anhang B werden alle benutzten Fortran-Programme jeweils dokumentiert aufgelistet.

Die Arbeit schließt mit einem Literaturverzeichnis und einem Index.

2 Problemstellung und Lösungsweg

2.1 Formulierung des abstrakten, verallgemeinerten Problems und Spezialisierung für die drei Kontinua mit Mikrostruktur

Das mathematische Problem, um dessen Lösung es geht, ist ein Randwertproblem. Es umfasst die drei Bereiche Differentialgleichungen, Randbedingungen und Lasten (siehe Abbildung 3).

Es wird gezeigt werden, dass das beschriebene Verfahren für alle Kontinua mit Mikrostruktur in der linearen Elastizitätstheorie geeignet ist. Daher werden Operatoren definiert, die eine verallgemeinerte Darstellung des Problems möglich machen. Für jedes spezielle Kontinuum sind diese Operatoren dann aufzuschlüsseln. Sie haben damit auch für die drei untersuchten Kontinua verschiedene Bedeutungen.

Das Cosserat-, das klassische und das Couple-Stress-Kontinuum fallen alle in die Rubrik „Cosserat-artig“ (siehe Tabelle 1). Das Cosserat-Kontinuum enthält das klassische Kontinuum und das Couple-Stress-Kontinuum als Spezialfälle.

Das verallgemeinerte Problem wird beherrscht von linearen partiellen Differentialgleichungen für die verallgemeinerten Verschiebungen und periodischen Randbedingungen an die verallgemeinerten Verzerren. Außerdem sind als „Lasten“ Mittelwerte der verallgemeinerten Verzerrungen vorgegeben (siehe Abbildung 3).

2.1.1 Differentialgleichungen

Fasst man den Verschiebungsvektor u und den axialen Vektor der Drehung v zu einem geordneten Paar $w \stackrel{\text{def}}{=} (u, v)$ zusammen, so lassen sich die Verschiebungs-Differentialgleichungen des Cosserat-Kontinuums, des klassischen Kontinuums und die des Couple-Stress-Kontinuums mit Hilfe eines linearen Differentialoperators L^w angeben als²

$$L^w w = 0 \quad . \quad (2.1)$$

²Gleichungen gelten für homogene Körper bei Abwesenheit von Volumenkräften und -momenten und Abwesenheit von Beschleunigungskräften (Statik)

2 PROBLEMSTELLUNG UND LÖSUNGSWEG

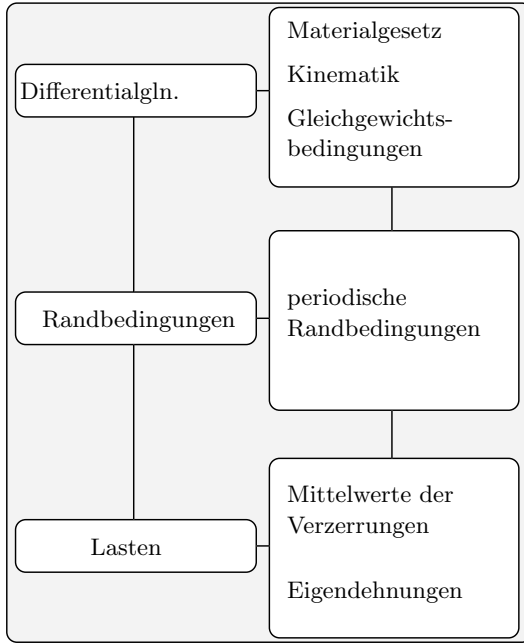


Abbildung 3: Mathematisches Problem

Für die drei verschiedenen Kontinua nimmt diese Gleichung folgende Gestalten an³(vgl. [Eringen 68]):

$$\begin{array}{ll}
 \text{Cosserat} & \left[\begin{array}{l} (\lambda + \mu)u_{l,lk} + (\mu + \kappa)u_{k,ll} + \kappa e_{klm}v_{m,l} = 0 \\ (\alpha + \beta)v_{l,lk} + \gamma v_{k,ll} + \kappa e_{klm}u_{m,l} - 2\kappa v_k = 0 \end{array} \right. \\
 \text{klassisch} & \left[(\lambda + \mu)u_{l,lk} + \mu u_{k,ll} = 0 \right. \\
 \text{Couple-Str.} & \left[\begin{array}{l} (\lambda + \mu + 1/2 \kappa)u_{k,lk} + 1/4 \gamma u_{k,lkmm} + \\ (\mu + 1/2 \kappa)u_{l,kk} - 1/4 \gamma u_{l,kkmm} = 0 \end{array} \right.
 \end{array} \quad (2.2)$$

³Kommas im Index bedeuten partielle Ableitungen nach den kartesischen Koordinaten.

Das Cosserat-Kontinuum umfasst, wie gesagt, das klassische Kontinuum und das Couple-Stress-Kontinuum als Spezialfälle. Für das klassische Kontinuum setzt man $\alpha = \beta = \gamma = \kappa = 0$. Und für das Couple-Stress-Kontinuum setzt man $v_m = 1/2 e_{mno} u_{o,n}$.

Diese Gleichungen erhält man aus der Gleichgewichtsbedingung nach Elimination der Spannungen (Einsetzen des Zusammenhangs zwischen den Spannungen und den Verzerrungen, also des Materialgesetzes) und nach Elimination der Verzerrungen (Einsetzen des Zusammenhangs zwischen den Verschiebungen und den Verzerrungen). Als Annahmen gehen dabei verschiedene Dinge ein: Neben der Isotropieannahme geht man davon aus, dass das Materialgesetz, das für homogene Körper hergeleitet wird, auch für inhomogene Körper gilt. Weiterhin nimmt man an, dass die Beschleunigungskräfte und die Volumenkräfte vernachlässigt werden können.

Setzt man andererseits in die (identischen) Gleichgewichtsbedingungen der drei untersuchten Kontinua mit Mikrostruktur, nämlich

$$\begin{array}{ll} \text{Cosserat} & \left[\begin{array}{l} \tau_{lk,l} = 0 \\ \mu_{lk,l} + e_{kmn} \tau_{mn} = 0 \end{array} \right. \\ \text{klassisch} & \left[\tau_{lk,l} = 0 \right. \\ \text{Couple-Str.} & \left[\begin{array}{l} \tau_{lk,l} = 0 \\ \mu_{lk,l} + e_{kmn} \tau_{mn} = 0 \end{array} \right. \end{array} \quad (2.3)$$

die jeweiligen Materialgesetze ein, nämlich

$$\begin{array}{ll} \text{Cosserat} & \left[\begin{array}{l} \tau_{lk} = \lambda \varepsilon_{rr} \delta_{lk} + (\mu + \kappa) \varepsilon_{lk} + \mu \varepsilon_{kl} - \varrho \epsilon^* \delta_{lk} \\ \mu_{lk} = \alpha \kappa_{rr} \delta_{lk} + \beta \kappa_{lk} + \gamma \kappa_{kl} \end{array} \right. \\ \text{klassisch} & \left[\tau_{lk} = \lambda \varepsilon_{rr} \delta_{lk} + 2\mu \varepsilon_{lk} - \varrho \epsilon^* \delta_{lk} \right. \\ \text{Couple-Str.} & \left[\begin{array}{l} \tau_{(lk)} = \lambda \varepsilon_{rr} \delta_{lk} + (2\mu + \kappa) \varepsilon_{lk} - \varrho \epsilon^* \delta_{lk} \\ \tau_{[lk]} = -1/2 e_{rlk} \mu_{nr,n} \\ \mu_{lk} = \beta \kappa_{lk} + \gamma \kappa_{kl} \end{array} \right. \end{array} \quad (2.4)$$

und ersetzt aber noch nicht die Verzerrungen durch die Verschiebungen, so erhält man mit der verallgemeinerten Verzerrung $\gamma \stackrel{\text{def}}{=} (\varepsilon, \kappa)$,

2 PROBLEMSTELLUNG UND LÖSUNGSWEG

äquivalent zu Gleichung (2.1)

$$\begin{aligned} \mathbf{L}^\gamma \gamma &= 0 \\ \gamma &= \mathbf{L}^{\gamma w} w \end{aligned} \quad (2.5)$$

mit $\mathbf{L}^\gamma \mathbf{L}^{\gamma w} = \mathbf{L}^w$.

Für die drei Kontinua ist das:

$$\begin{aligned} \text{Cosserat} & \quad \begin{cases} (\lambda \varepsilon_{rr} \delta_{lk} + (\mu + \kappa) \varepsilon_{lk} + \mu \varepsilon_{kl} - \varrho \epsilon^* \delta_{lk}),_{,l} = 0 \\ (\alpha \kappa_{rr} \delta_{lk} + \beta \kappa_{lk} + \gamma \kappa_{kl}),_{,l} + \\ e_{kmn} (\lambda \varepsilon_{rr} \delta_{mn} + (\mu + \kappa) \varepsilon_{mn} + \mu \varepsilon_{nm}) = 0 \end{cases} \\ \text{klassisch} & \quad [(\lambda \varepsilon_{rr} \delta_{lk} + 2\mu \varepsilon_{lk} - \varrho \epsilon^* \delta_{lk}),_{,l} = 0 \\ \text{Couple-Str.} & \quad \begin{cases} \{ \lambda \varepsilon_{rr} \delta_{lk} + (2\mu + \kappa) \varepsilon_{lk} - \varrho \epsilon^* \delta_{lk} - \\ 1/2 e_{rlk} (\beta \kappa_{nr} + \gamma \kappa_{rn}),_{,n} \},_{,l} = 0 \\ (\beta \kappa_{lk} + \gamma \kappa_{kl}),_{,l} + e_{kmn} \{ \lambda \varepsilon_{rr} \delta_{mn} + (2\mu + \kappa) \varepsilon_{mn} - \\ 1/2 e_{rmn} (\beta \kappa_{nr} + \gamma \kappa_{rn}),_{,n} \} = 0 \end{cases} \end{aligned} \quad (2.6)$$

Und die Verzerrungs-Verschiebungs-Zusammenhänge lauten:

$$\begin{aligned} \text{Cosserat} & \quad \begin{cases} \varepsilon_{kl} = u_{l,k} + e_{lkm} v_m \\ \kappa_{kl} = v_{k,l} \end{cases} \\ \text{klassisch} & \quad [\varepsilon_{kl} = 1/2 (u_{l,k} + u_{k,l}) \\ \text{Couple-Str.} & \quad \begin{cases} \varepsilon_{kl} = 1/2 (u_{l,k} + u_{k,l}) \\ \kappa_{kl} = 1/2 e_{krs} u_{s,r,l} \end{cases} \end{aligned} \quad (2.7)$$

2.1.2 Randbedingungen

Als nächstes ist das Gebiet festzulegen, auf dem die Differentialgleichungen gelten und die Randbedingungen, die an den Grenzen des Gebiets erfüllt werden sollen.

In dieser Arbeit geht es aus ausschließlich um periodische Randbedingungen. Solche findet man „in der Natur“ dort, wo eine Struktur sich periodisch ausgebildet hat und die Belastung auf der Struktur homogen ist, so dass alle Abschnitte denselben Belastungen ausgesetzt sind und

zusätzlich dieselbe Struktur aufweisen. Das betrachtete Gebiet (welches sich in alle Raumrichtungen periodisch fortsetzt) sei dreidimensional und quaderförmig, und es wird als repräsentatives Volumenelement bezeichnet (siehe Abbildung 4).

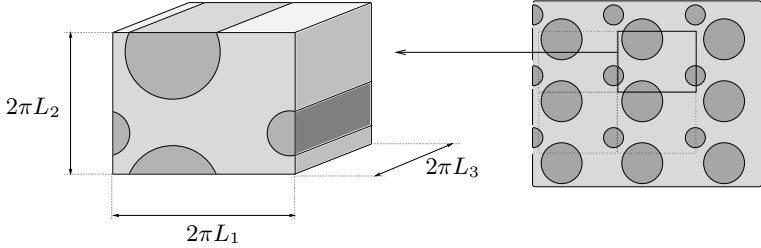


Abbildung 4: Zweidimensionale unendlich ausgedehnte Faser-Matrix Struktur bestehend aus zwei Komponenten (rechts) und dreidimensionales repräsentatives Volumenelement mit periodischen Randbedingungen (links)

2.1.3 Lasten

Als Lasten seien Mittelwerte der Verzerrungen vorgegeben. Mit verallgemeinerten Verzerrungen ist gemeint:

$$\langle \gamma \rangle = \gamma_0 \quad (2.8)$$

Für die drei Kontinua ist dies gleichbedeutend mit:

$$\begin{array}{ll} \text{Cosserat} & \left[\begin{array}{l} \langle \varepsilon_{kl} \rangle = \varepsilon_{0_{kl}} \\ \langle \kappa_{kl} \rangle = \kappa_{0_{kl}} \end{array} \right. \\ \text{klassisch} & \left[\langle \varepsilon_{kl} \rangle = \varepsilon_{0_{kl}} \right. \\ \text{Couple-Str.} & \left[\langle \varepsilon_{kl} \rangle = \varepsilon_{0_{kl}} \right. \end{array} \quad (2.9)$$

Unabhängig von den Mittelwerten kann den drei untersuchten Kontinua mit Mikrostruktur (zusätzlich) eine dilatorische Eigendehnung ϵ^*

2 PROBLEMSTELLUNG UND LÖSUNGSWEG

aufgeprägt werden derart, dass bei ausbleibenden Gesamtdehnungen Eigenspannungen

$$\tau_{lk}^* = - \underbrace{(3\lambda + (2\mu + \kappa))}_{\stackrel{\text{def}}{=} \varrho} \epsilon^* \delta_{lk} \quad (2.10)$$

$$= -\varrho \epsilon^* \delta_{lk} \quad (2.11)$$

resultieren.

2.2 Lösungsweg

Das Problem besteht aus den Differentialgleichungen, den Randbedingungen und den Lasten (siehe Abbildung 3). Die periodischen Randbedingungen werden bereits dadurch befriedigt, dass für die Lösung ein Ansatz in Form einer Fourierreihe gemacht wird. Es bleiben also die Differentialgleichungen und die Lasten. Diese lauten in Operator-schreibweise für alle drei Kontinua:

$$\gamma = \mathbf{L}^{\gamma w} w, \quad (2.12a)$$

$$\mathbf{L}^{\gamma} \gamma = 0, \quad (2.12b)$$

$$\langle \gamma \rangle = \gamma_0. \quad (2.12c)$$

Außerdem ist für jedes konkrete Problem die örtliche Verteilung aller Materialwerte vorgegeben, ebenso wie die örtliche Verteilung der Eigendehnungen. Auch die Abmessungen des repräsentativen Volumenelements liegen bei jeder Problemstellung fest.

Die Lösung des gestellten Problems geschieht in mehreren Schritten: An einem vereinfachten Problem definiert man einen Lösungsoperator, der dann auch am Originalproblem Anwendung findet. Mit ihm kann die Lösung des Originalproblems in impliziter Form abstrakt hingeschrieben werden. Mit Hilfe einer Neumann-Reihe gelangt man weiter zu einer abstrakten Formulierung der Lösung des Originalproblems in expliziter Form. Der Lösungsoperator kann jedoch nur im Fourierraum berechnet werden. Eine Übersicht zur Lösungsstrategie bietet Abbildung 5. Details zur Lösung findet man in den folgenden Abschnitten und im Anhang.

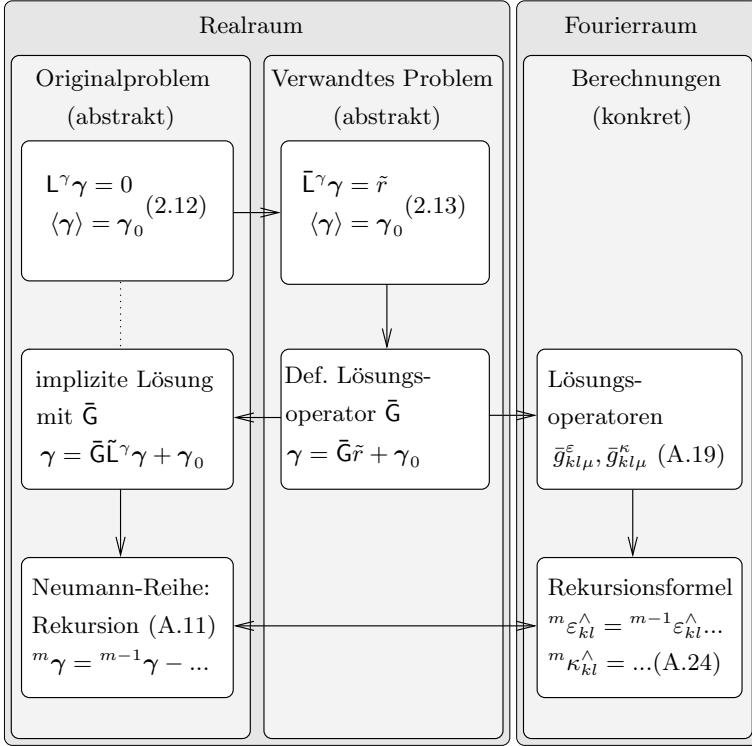


Abbildung 5: Schematische Darstellung des Lösungswegs: Das gegebene Problem wird vereinfacht, und am vereinfachten Problem wird ein Lösungsoperator definiert, mit dessen Hilfe sich eine implizite Lösung des gegebenen Problems aufschreiben lässt. Durch eine Neumann-Reihe erhält man aus der impliziten eine explizite iterative Lösungsformel. Der Lösungsoperator selbst kann nur im Fourierraum berechnet werden. Er wird in der Rekursion im Fourierraum benutzt.

2.2.1 Implizite Lösung im Realraum

Aufgrund der Ortsabhängigkeit der Materialwerte α, μ usw. sind die (Koeffizienten der) Differentialoperatoren \mathbb{L}^w und \mathbb{L}^γ ebenfalls ortsabhängig, was die Bestimmung der gesuchten Lösung γ erschwert.

Die formale Ähnlichkeit mit dem verwandten Problem mit konstanten Koeffizienten und vorgegebener rechter Seite \tilde{r} , nämlich

$$\gamma = \mathbb{L}^{\gamma w} w, \quad (2.13a)$$

$$\bar{\mathbb{L}}^\gamma \gamma = \tilde{r}, \quad (2.13b)$$

$$\langle \gamma \rangle = \gamma_0, \quad (2.13c)$$

gestattet es dennoch, eine Lösung anzugeben: Man definiert dazu

$$\begin{aligned} \lambda(x) &= \bar{\lambda} - \tilde{\lambda}(x), & \mu(x) &= \bar{\mu} - \tilde{\mu}(x), & \kappa(x) &= \bar{\kappa} - \tilde{\kappa}(x), \\ \gamma(x) &= \bar{\gamma} - \tilde{\gamma}(x), & \alpha(x) &= \bar{\alpha} - \tilde{\alpha}(x), & \beta(x) &= \bar{\beta} - \tilde{\beta}(x), \\ \varrho(x) &= \bar{\varrho} - \tilde{\varrho}(x) \end{aligned}$$

und dazu passend

$$\mathbb{L}^\gamma = \bar{\mathbb{L}}^\gamma - \tilde{\mathbb{L}}^\gamma, \quad (2.14a)$$

$$\mathbb{L}^w = \bar{\mathbb{L}}^w - \tilde{\mathbb{L}}^w. \quad (2.14b)$$

Hierbei bezeichnet die Tilde \sim Ortsabhängigkeit. Und der Überstrich weist auf einen Zusammenhang zu nicht ortsabhängigen Größen hin. Beispielsweise wirkt \mathbb{L}^γ auf die Verzerrungen gekoppelt mit den Materialwerten λ, μ usw. (siehe Gleichungen (2.6)), während $\tilde{\mathbb{L}}^\gamma$ im Unterschied zu \mathbb{L}^γ auf $\tilde{\lambda}$ und $\tilde{\mu}$ wirkt, jedoch dieselbe Form wie dieser besitzt.

Weiterhin sei der Lösungsoperator $\tilde{\mathbb{G}}$ definiert über

$$\gamma = \tilde{\mathbb{G}} \tilde{r} + \gamma_0 \quad (2.15a)$$

$$\Leftrightarrow \gamma \text{ löst (2.13) mit periodischen Randbedingungen} \quad (2.15b)$$

Dann ist die gesuchte Lösung von (2.12) in impliziter Form bekannt als

$$\gamma = \tilde{\mathbb{G}} \tilde{\mathbb{L}}^\gamma \gamma + \gamma_0. \quad (2.16)$$

Der Lösungsoperator $\tilde{\mathbb{G}}$ kann ohne Weiteres im Fourierraum berechnet werden (Herleitung des Lösungsoperators im Fourierraum $\mathcal{F}\{\tilde{\mathbb{G}}\}$ in Abschnitt A.3, Definition der Lösungsoperatoren im Fourierraum $\tilde{g}^\varepsilon, \tilde{g}^\kappa$ siehe Definitionen (A.19)).

Außerdem kann statt der impliziten Lösungsformel eine (iterative) explizite Formel zur näherungsweisen Berechnung von γ angegeben werden, was im nächsten Abschnitt geschieht.

2.2.2 Explizite, näherungsweise Lösung im Fourierraum

Im Folgenden werden Begriffe wie Fourierreihe, Fourierkoeffizienten und Fouriertransformierte vorausgesetzt. Die entsprechenden Definitionen sind in den Gleichungen (3.1) und (3.2) gegeben.

Es wird vereinfachend angenommen, dass sich alle Funktionen als Fourierreihe darstellen lassen und dass die Fourierreihe einer Funktion identisch ist mit der Funktion selbst, so dass zwischen der Funktion und ihrer Fourierreihe nicht unterschieden werden muss. Ausgangspunkt ist die Fouriertransformierte von (2.13):

$$\bar{\mathcal{L}}^\gamma \gamma = \tilde{r} \quad \forall x \quad \Leftrightarrow \quad \mathcal{F}\{\bar{\mathcal{L}}^\gamma \gamma\} = \mathcal{F}\{\tilde{r}\} \quad \forall k \neq 0 \quad , \quad (2.17a)$$

$$\langle \gamma \rangle = \gamma_0 \quad \Leftrightarrow \quad \mathcal{F}\{\gamma\} = \gamma_0 \quad \text{falls} \quad k = 0 \quad , \quad (2.17b)$$

wobei die Fouriertransformierte nichts weiter ist als die (diskrete Funktion der) Fourierkoeffizienten.

Man findet folgende Darstellung:

$$\mathcal{F}\{\gamma\} = \begin{cases} \mathcal{F}\{\bar{\mathcal{G}}\} \mathcal{F}\{\tilde{r}\} & \forall k \neq 0 \\ \gamma_0 & k = 0 \end{cases} . \quad (2.18)$$

Die ausgeschriebene Form dieser Gleichung ist (A.18).

Die Lösung von (2.16) ist dann:

$$\mathcal{F}\{\gamma\} = \begin{cases} \mathcal{F}\{\bar{\mathcal{G}}\} \mathcal{F}\{\bar{\mathcal{L}}^\gamma \gamma\} & \forall k \neq 0 \\ \gamma_0 & k = 0 \end{cases} . \quad (2.19)$$

Diese implizite Formel kann mithilfe einer Neumann-Reihe in eine explizite Form gebracht werden - allerdings nur als rekursive Formel. Damit gelingt es (falls die Reihe konvergiert), eine Näherung für die gesuchte Größe anzugeben. Man erhält nach kurzer Rechnung (Berechnung von $\mathcal{F}\{\mathcal{L}^{\gamma \ m-1} \gamma\}$ in Abschnitt A.4) für die m -te Näherung ($m = 2, 3, \dots$)

$$\mathcal{F}\{\gamma^m\} = \begin{cases} \mathcal{F}\{\gamma^{m-1}\} - \mathcal{F}\{\bar{\mathcal{G}}\} \mathcal{F}\{\mathcal{L}^{\gamma \ m-1} \gamma\} & \forall k \neq 0 \\ \gamma_0 & k = 0 \end{cases} . \quad (2.20)$$

2 PROBLEMSTELLUNG UND LÖSUNGSWEG

Hierbei bezeichnet ${}^{m-1}\gamma$ die $(m-1)$ -te Näherung, und ${}^m\gamma$ bezeichnet die m -te Näherung. Als Startwert der Iteration ($m=1$) wird gesetzt $\mathcal{F}\{{}^1\gamma\} = \gamma_0$ - also eine konstante Verzerrung mit den vorgegebenen Mittelwerten.

Damit liegen alle Formeln zur Lösung des Randwertproblems fest.

3 Numerische Behandlung mit Fourierreihen

Sowohl bei der analytischen als auch bei der numerischen Lösung des Problems wird angenommen, dass sich alle betrachteten Felder als Fourierreihen darstellen lassen. Man könnte auch sagen, dass für die gesuchten Felder „Ansätze“ vom Typ einer Fourierreihe benutzt werden. Diese Methode hat den Nachteil, dass sie einer Festlegung auf periodische Randbedingungen entspricht. Das heißt, dass mit dieser Methode nur ganz spezielle Randwertprobleme gelöst werden können - nämlich nur solche mit periodischen Randbedingungen. Die Methode zeigt jedoch auch wichtige Vorteile. Diese sind im einzelnen:

- Der Fehler des gelösten Problems ist Null (abgesehen von Rundungsfehlern, die die numerische Behandlung mit sich bringt), wenn die Iteration konvergiert.
- Es können mit geringem Rechenaufwand Gebiete von „unendlicher“ Größe untersucht werden, denn die periodischen Randbedingungen setzen sich bis in die „Unendlichkeit“ fort.
- Die Methode ist sehr einfach und sehr übersichtlich.
- Die Fourieranalyse ist als grundlegendes Werkzeug der Naturwissenschaften seit langem etabliert.⁴

3.1 Fourierkoeffizienten und Fourierreihe

Grundlegendes zur Fourieranalysis liest man z.B. in [Courant 68] oder in [Brigola 97] nach. Hier seien nur die wichtigsten Definitionen gegeben.

Eine Funktion $f(x)$ sei periodisch bezüglich der Intervalle $x_1 \in [0, 2\pi L_1)$, $x_2 \in [0, 2\pi L_2)$, $x_3 \in [0, 2\pi L_3)$, wobei $2\pi L_i$ die Seitenlängen des dreidimensionalen repräsentativen Volumelements sind (siehe Abbildung 4).

⁴Historische Anmerkung: Jean Baptiste Joseph Fourier (1768 - 1830) benutzte trigonometrische Reihen zur Darstellung von Lösungen der (Fourierschen) Wärmeleitungsgleichung.

3 NUMERISCHE BEHANDLUNG MIT FOURIERREIHEN

Die Definition der Fourierkoeffizienten lautet:

$$\mathcal{F}\{f\}(k) = f^\wedge(k) \quad (3.1a)$$

$$\stackrel{\text{def}}{=} \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 f(x) e^{-\iota k \cdot x} \quad (3.1b)$$

$$= \frac{1}{8\pi^3} \int_{x_1=0}^{2\pi} dx_1 \int_{x_2=0}^{2\pi} dx_2 \int_{x_3=0}^{2\pi} dx_3 f(x) e^{-\iota k^L \cdot x} \quad (3.1c)$$

Hierbei sind zwei Wellenvektoren definiert worden: Der Wellenvektor k ist so definiert, dass $k_i \in \{\dots, -\frac{2}{L_1}, -\frac{1}{L_1}, 0, \frac{1}{L_1}, \frac{2}{L_1}, \dots\}$. Und der dimensionslose Wellenvektor k^L ist hierbei so definiert, dass $k_i^L \in \{\dots, -2, -1, 0, 1, 2, \dots\}$.

Die inverse Fourier Transformation (hier: die Fourierreihe) ist definiert als:

$$f(x) = \mathcal{F}^{-1}\{f^\wedge\}(x) \quad (3.2a)$$

$$= \mathcal{F}^{-1}\{\mathcal{F}\{f\}\}(x) \quad (3.2b)$$

$$\stackrel{\text{def}}{=} \sum_{k_1^L=-\infty}^{\infty} \sum_{k_2^L=-\infty}^{\infty} \sum_{k_3^L=-\infty}^{\infty} \mathcal{F}\{f\} e^{\iota(k_1^L/L_1, k_2^L/L_2, k_3^L/L_3) \cdot (x_1, x_2, x_3)} \quad (3.2c)$$

$$= \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} \sum_{k_3=-\infty}^{\infty} f^\wedge e^{\iota k \cdot x} \quad (3.2d)$$

3.1.1 Orthogonalitätsrelationen

Lineare partielle Differentialgleichungen können mit Ansätzen in Form von Fourierreihen gelöst werden. Im vorliegenden Problem werden Differentialgleichungen mit nicht konstanten Koeffizienten ebenfalls mit Fourierreihen gelöst. Ihre Lösung gelingt jedoch nicht immer, sondern nur, falls die Neumann-Reihe konvergiert.

Bei der Lösung der Gleichungen mit Fourierreihen gewinnt man aus den Differentialgleichungen algebraische Gleichungen für die Fourierkoeffizienten der gesuchten Lösung. Wesentlich hierbei ist die Ausnutzung der Orthogonalitätsrelationen (Orthonormalitätsrelationen).

3.1 FOURIERKOEFFIZIENTEN UND FOURIERREIHE

Im Raum der periodischen Funktionen mit den Perioden $2\pi L_1$, $2\pi L_2$ und $2\pi L_3$ lässt sich ein Skalarprodukt (inneres Produkt im Funktionenraum) definieren. Üblich ist folgende Definition für das innere Produkt aus zwei (komplexen) Funktionen f und g (siehe hierzu auch [Brigola 97]):

$$\langle f, g \rangle \stackrel{\text{def}}{=} \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 f \bar{g} \quad . \quad (3.3)$$

Für diese Abbildung (inneres Produkt) gelten folgende Beziehungen $\forall \alpha, \beta \in \mathbb{C}$ und $\forall f, g, h$ aus dem Funktionenraum der $2\pi L_i$ -periodischen Funktionen ⁵:

$$\begin{aligned} \langle f + g, h \rangle &= \langle f, h \rangle + \langle g, h \rangle \\ \langle f, g + h \rangle &= \langle f, g \rangle + \langle f, h \rangle \\ \langle \alpha f, g \rangle &= \alpha \langle f, g \rangle \\ \langle f, \beta g \rangle &= \bar{\beta} \langle f, g \rangle \\ \langle f, g \rangle &= \overline{\langle g, f \rangle} \\ \langle f, f \rangle &> 0 \\ \langle f, f \rangle &= 0 \Leftrightarrow f(x) = 0 \quad \forall x \quad . \end{aligned} \quad (3.4)$$

Ebenfalls leicht zeigen lässt sich, dass folgende Orthogonalitätsrelatio-

⁵Dies sind die Voraussetzungen dafür, dass es sich hier um ein inneres Produkt handelt. Dass alle Bedingungen erfüllt sind, lässt sich mit Hilfe der Definition leicht zeigen.

3 NUMERISCHE BEHANDLUNG MIT FOURIERREIHEN

nen gültig sind:

$$\begin{aligned}
 \langle e^{\iota k \cdot x}, e^{\iota l \cdot x} \rangle &= \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 e^{\iota k \cdot x} \overline{e^{\iota l \cdot x}} \\
 &= \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 e^{\iota k \cdot x} e^{-\iota l \cdot x} \\
 &= \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 e^{\iota(k-l) \cdot x} \\
 &= \int_{\alpha_1=0}^1 d\alpha_1 \int_{\alpha_2=0}^1 d\alpha_2 \int_{\alpha_3=0}^1 d\alpha_3 e^{2\pi \iota(k-l) \cdot \alpha} \\
 &= \begin{cases} 1 & \text{falls } k = l \\ 0 & \text{sonst} \end{cases} .
 \end{aligned} \tag{3.5}$$

Diese Orthogonalitätsrelationen machen eine anschauliche (geometrische) Interpretation der Fourierkoeffizienten möglich:

$$f^\wedge(k) = \langle f, e^{\iota k \cdot x} \rangle \tag{3.6}$$

Der Fourierkoeffizient zum Wellenvektor k ist die „Projektion“ der Funktion auf die k –(Ansatz-)Funktion, d.h. der Anteil der Funktion in die „Richtung“ der Ansatzfunktion.

3.2 Diskrete Fourier Transformation

Die numerische Berechnung der Fourierkoeffizienten ist im Allgemeinen sehr aufwändig, und zudem können in der Praxis nur endlich viele Koeffizienten berücksichtigt werden. Es werden daher bei der numerischen Umsetzung Näherungen für endlich viele Koeffizienten berechnet. In dieser Arbeit wird dazu die diskrete Fourier Transformation benutzt.

Die diskrete Fourier Transformation einer Zahlenfolge f_n ist definiert als:

$$\mathcal{F}^d\{f_n\} = f_k^\wedge \tag{3.7a}$$

$$\stackrel{\text{def}}{=} \frac{1}{N_1 N_2 N_3} \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \sum_{n_3=0}^{N_3-1} f_n e^{-\iota 2\pi k^L \cdot a_n} . \tag{3.7b}$$

Die Folge f_n ist hier die Funktion f ausgewertet an diskreten Stellen $a_n = (n_1/N_1, n_2/N_2, n_3/N_3)$ mit $f_n = f(a_n)$. Die Diskretisierung in jede der drei Raumrichtungen ist dabei definiert durch die ungeraden Zahlen $N_i = 2m_i + 1$. Die diskrete Fourier Transformation erzeugt Näherungswerte für die Fourierkoeffizienten von f . Diese Näherungen sind die Fourierkoeffizienten eines trigonometrischen Interpolationspolynoms durch die Funktionswerte f_n .

Die Näherung ist nur sinnvoll für $|k_i^L| \leq \frac{N_i-1}{2}$ (Nyquist Frequenz), siehe hierzu [Brigola 97]. Die inverse diskrete Fourier Transformation ist definiert als:

$$f(a_n) = f_n \quad (3.8a)$$

$$= \mathcal{F}^{\text{d}^{-1}} \{ \hat{f}_{k^L} \} \quad (3.8b)$$

$$= \mathcal{F}^{\text{d}^{-1}} \{ \mathcal{F}^{\text{d}} \{ f \} \} \quad (3.8c)$$

$$\stackrel{\text{def}}{=} \sum_{k_1^L=-m_1}^{m_1} \sum_{k_2^L=-m_2}^{m_2} \sum_{k_3^L=-m_3}^{m_3} \hat{f}_{k^L} e^{+i2\pi k^L \cdot a_n} \quad (3.8d)$$

Ersetzt man in der letzten Formel die diskreten Werte $0 \leq a_n < 1$ durch kontinuierliche Werte $0 \leq a_i < 1$, so erhält man ein trigonometrisches Interpolationspolynom durch f_n .

Die näherungsweise Bestimmung der Fourierkoeffizienten kann verbessert werden, indem man sogenannte Abminderungsfaktoren benutzt (siehe hierzu [Kaßbohm 05b]). Außerdem können die Interpolationspolynome z.B. durch Berechnung der Fejérschen Mittel geglättet werden (siehe hierzu [Brigola 97]).

3.3 Faltung

3.3.1 Periodische Faltung

In (A.22) erkennt man, dass vielfach die Fouriertransformierte von Produkten von Funktionen bestimmt werden müssen. Zur Berechnung ist es hierbei günstig, Gebrauch von der periodischen Faltung zu machen. Sie die Fourierkoeffizienten zweier Funktionen g und h bekannt, so lassen sich daraus die Fourierkoeffizienten des Produkts $f = g \cdot h$ bestimmen.

3 NUMERISCHE BEHANDLUNG MIT FOURIERREIHEN

Einen Beweis hierzu findet man z.B. in [Brigola 97].

$$\mathcal{F}\{f\}(k) = \mathcal{F}\{g \cdot h\}(k) \quad (3.9a)$$

$$= \mathcal{F}\{g\} * \mathcal{F}\{h\} \quad (3.9b)$$

$$= \{g\}^\wedge * \{h\}^\wedge \quad (3.9c)$$

$$(g \cdot h)^\wedge(k) \stackrel{\text{def}}{=} \sum_{o_1=-\infty}^{\infty} \sum_{o_2=-\infty}^{\infty} \sum_{o_3=-\infty}^{\infty} g_o^\wedge \cdot h_{k-o}^\wedge \quad (3.9d)$$

3.3.2 Diskrete periodische Faltung

Seien g_n und h_n Zahlenfolgen wie oben, und seien g_{kL}^\wedge sowie h_{kL}^\wedge deren diskrete Fouriertransformierte (bzw. Fourierkoeffizienten eines entsprechenden trigonometrischen Interpolationspolynoms). Und sei f_n die „Produktfolge“ mit näherungsweisen Fourierkoeffizienten f_{kL}^\wedge . Dann ist wegen (3.9) auch:

$$\mathcal{F}^d\{f_n\} = \mathcal{F}^d\{(g \cdot h)_n\} \quad (3.10a)$$

$$f_{kL}^\wedge = \mathcal{F}^d\{g_n\} *^d \mathcal{F}^d\{h_n\} \quad (3.10b)$$

$$((g \cdot h)_n)_{kL}^\wedge = \sum_{o_1^L=-m_1}^{m_1} \sum_{o_2^L=-m_2}^{m_2} \sum_{o_3^L=-m_3}^{m_3} g_{oL}^\wedge \cdot h_{kL-oL}^\wedge \quad (3.10c)$$

3.4 Zusammenfassung

In diesem Abschnitt wurden die Hilfsmittel bereitgestellt, die zur numerischen Lösung eines vorgelegten Problems nötig sind. Die Theorie der Fourierreihen wurde kurz dargestellt, und die wichtigsten Begriffe wurden geklärt. In der folgenden Abbildung wird die Stellung der numerischen Lösung im Lösungsweg dargestellt.

Die mathematische Einleitung zeigt, wie man zu einer Rekursionsformel für die Lösung gelangt. Hierbei ist die Besonderheit, dass der Lösungsoperator nur im Fourierraum berechnet werden kann. Mit Hilfe der Fourierreihen wird das Problem in den Fourierraum überführt. Dabei kommt der (näherungsweisen) Berechnung der Fourierkoeffizienten eine besondere Bedeutung zu. Die Lösung wird im Fourierraum iterativ verbessert, bis der Fehler (die Abweichung vom Gleichgewicht, siehe hierzu ??) akzeptabel ist. Schließlich wird das Problem in den

3.4 ZUSAMMENFASSUNG

Realraum zurückgeführt, indem man die Fourierreihen der Felder berechnet, deren Fourierkoeffizienten nun bekannt sind.

3 NUMERISCHE BEHANDLUNG MIT FOURIERREIHEN

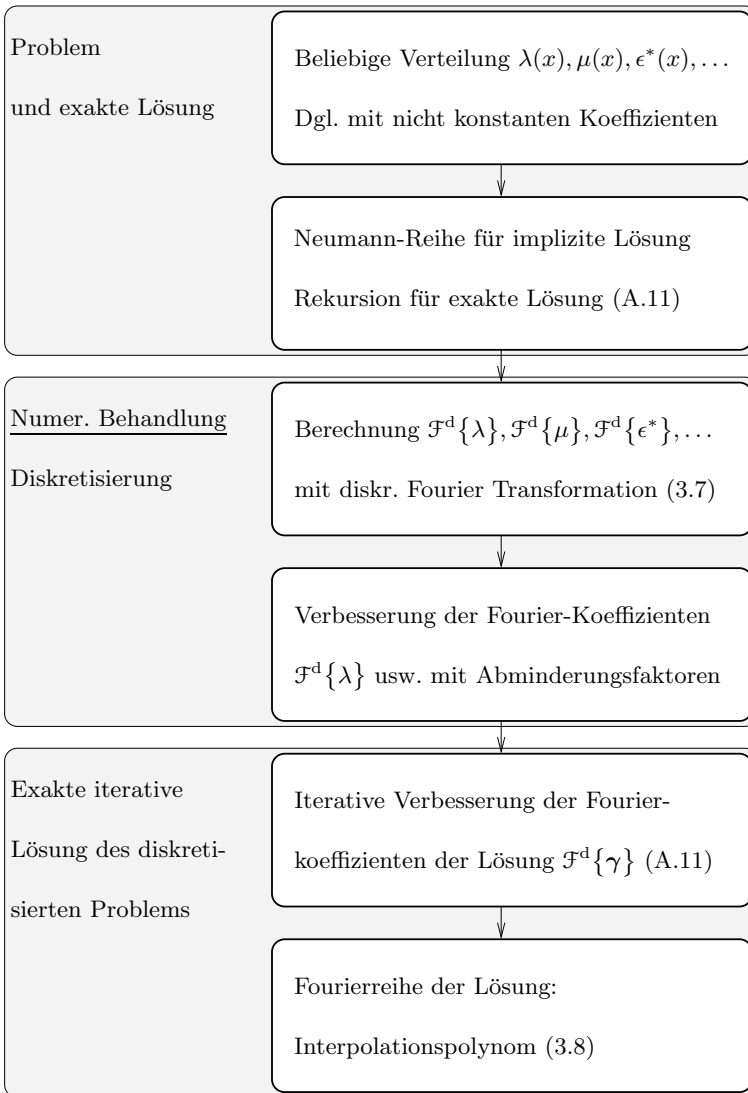


Abbildung 6: Stellung der numerischen Behandlung im Lösungsweg

4 Beispiele

4.1 Zwei akademische Beispiele

4.1.1 Beschreibung der akademischen Beispiele

Als Veranschaulichung werden zwei akademische Beispiele untersucht. Grundlage ist eine zweidimensionale, aus zwei isotropen Komponenten bestehende, Struktur. In die dritte Raumrichtung werden alle Größen konstant gesetzt. Die Abmessungen des repräsentativen Volumenelements sind mit der Wahl $L_1 = L_2 = 1$. festgelegt. Und es werden periodische Randbedingungen gefordert (siehe Abbildung 7).

Das RVE besteht aus zwei Stoffen mit verschiedenen Materialwerten. Es gilt für die Bezugswerte $\bar{\lambda} = \bar{\mu} = \bar{\kappa} = \bar{\alpha} = \bar{\beta} = \bar{\gamma} = 10$ für das Cosserat- und das Couple-Stress-Kontinuum und $\bar{\lambda} = \bar{\mu} = 10, \bar{\kappa} = \bar{\alpha} = \bar{\beta} = \bar{\gamma} = 0$ für das klassische Kontinuum. Diese Werte sind grundlegend für die Berechnung der Lösungsoperatoren. Die Wahl dieser Werte beeinflusst die Konvergenz des Verfahrens. Um Konvergenz zu erzielen, sollten die Bezugswerte etwa den örtlichen Mittelwerten der Steifigkeiten entsprechen.

Für die Wahl der Materialwerte der Faser und der Matrix gilt folgendes: In der Faser sind alle Materialwerte halb so groß wie die Bezugswerte. In der Matrix sind alle Materialwerte doppelt so groß wie die Bezugswerte (siehe Abbildung 7).

Alle eben aufgezählten Parameter gelten für beide akademischen Beispiele. Den Unterschied zwischen den Experimenten bilden die Lasten. Da es zwei Typen von Lasten gibt, werden auch zwei Experimente durchgeführt.

Im ersten Beispiel wird der Mittelwert der 11-Komponente des Verzerrungstensors vorgegeben als 1.. Alle anderen Mittelwerte sind Null. Eigendehnungen gibt es in diesem Fall nicht. Im zweiten Beispiel gibt es Eigendehnungen in der Faser (aber nicht in der Matrix) und alle mittleren Verzerrungen sind Null.

4.1.2 Diskretisierung

Statt das Problem für die vorgegebene (im Beispiel unstetige) Verteilung der Materialwerte und für die vorgegebene Verteilung der Eigendehnungen zu lösen, wird es für genäherte Verteilungen dieser Funktio-

4 BEISPIELE

Beispiel	$\langle \gamma \rangle \neq 0.$	ϵ^* in Faser
1	$\langle \varepsilon_{11} \rangle = 1.$	$\epsilon^* = 0.$
2	-	$\epsilon^* = 1.$

Tabelle 2: Zwei akademische Beispiele: Im Beispiel 1 wird eine von Null verschiedene mittlere Dehnung vorgegeben. Im Beispiel 2 sind alle mittleren Dehnungen Null, und es wird eine dilatorische Eigen-
dehnung vorgegeben.

nen gelöst. Dazu wird das Problem gitterartig diskretisiert.

Die tatsächliche Problemstellung umfasst die Information über die Steifigkeits- und Eigendehnungsverteilung an allen Orten im RVE. Zur Berechnung der Fourierkoeffizienten werden aber nicht alle Funktionswerte benutzt, sondern es werden nur die Funktionswerte an den Kreuzungspunkten des Diskretisierungsgitters verwendet. Daher können die berechneten Fourierkoeffizienten nur Näherungen der exakten Fourierkoeffizienten der betreffenden Funktion sein.

Die Funktionswerte an den Kreuzungspunkten des Diskretisierungsgitters sind die Eingabe für die diskrete Fourier Transformation (3.7). Mit ihr berechnet man die (exakten) Fourierkoeffizienten eines trigonometrischen Interpolationspolynoms durch die Funktionswerte an den Kreuzungspunkten. Und diese Fourierkoeffizienten sind Näherungen für die Fourierkoeffizienten der gegebenen Funktionen (siehe hierzu Abbildung 8). Die Güte der Näherung hängt von der Feinheit der vorgegebenen Diskretisierung ab. Abbildung 10 zeigt die trigonometrischen Interpolationspolynome für verschiedene Diskretisierungen, auf die jeweils die diskrete Fourier Transformation angewendet wurde. Das Interpolationspolynom benutzt die Kreuzungspunkte des Diskretisierungsgitters als Stützstellen. Zwischen den Kreuzungspunkten wird die gegebene Verteilung $\lambda(x)$ jedoch nur näherungsweise nachgebildet. Bei unstetigen Funktionen treten die in der Fourieranalysis üblichen Oszillationen (Gibbs-Phänomen) auf (siehe Abbildung 10).

Die Fourierkoeffizienten können verbessert werden, indem z.B. Abminderungsfaktoren für lineare Splines benutzt werden. Den Abminderungsfaktoren liegt die Idee zugrunde, dass man auch die Funktionswerte zwischen den Kreuzungspunkten der Gitter berücksichtigen kann. Und zwar, indem man einen Spline (beliebiger Ordnung) durch die Git-

4.1 ZWEI AKADEMISCHE BEISPIELE

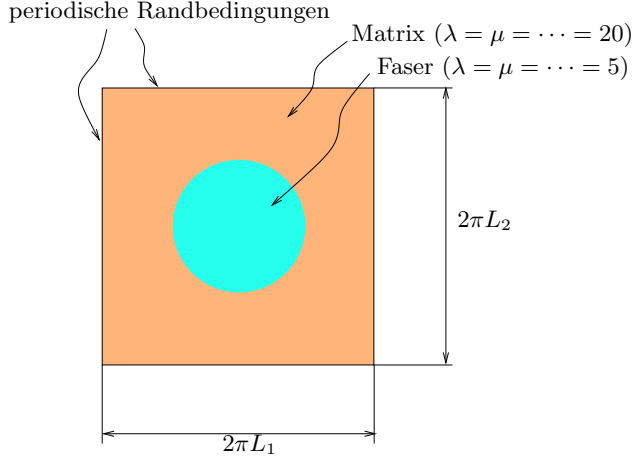


Abbildung 7: „Lokal isotropes“ RVE: Weiche Faser in steifer Matrix

terpunkte legt. Die mit Abminderungsfaktoren berechneten Näherungen für die gegebenen Funktionen $\lambda(x)$ usw. sind im Sinne der L_2 -Norm gute Näherungen - in den meisten Fällen sind sie jedenfalls bessere Näherungen als die Koeffizienten, die nicht “abgemindert” sind. Die Fourierkoeffizienten des Splines lassen sich jedoch exakt ermitteln. Man stellt bei ihrer Berechnung fest, dass zwischen den exakten Koeffizienten des Splines und den Koeffizienten des trigonometrischen Interpolationspolynoms⁶ ein Zusammenhang besteht. Dieser lautet mit den Abminderungsfaktoren α_i :

$$\hat{\boxtimes}_{\text{lin. Spline}} = \alpha_1 \alpha_2 \alpha_3 \hat{\boxtimes}_{\text{trig. Int.-Pol.}} \quad (4.1)$$

Für lineare Splines ergibt sich (hier keine Summation über i):

$$\alpha_i = \begin{cases} \left\{ \frac{N_i}{k_i^L \pi} \sin \frac{k_i^L \pi}{N_i} \right\}^2 & \text{falls } k_i^L \neq 0 \\ 1 & \text{falls } k_i^L = 0 \end{cases} \quad (4.2)$$

Die genäherte Steifigkeitsverteilung für die akademischen Beispiele bei einer Diskretisierung von $m_1 \times m_2 = 2 \times 2$ ist in den Abbildungen 11

⁶also denen, die man mit der diskreten Fourier Transformation ermittelt

4 BEISPIELE

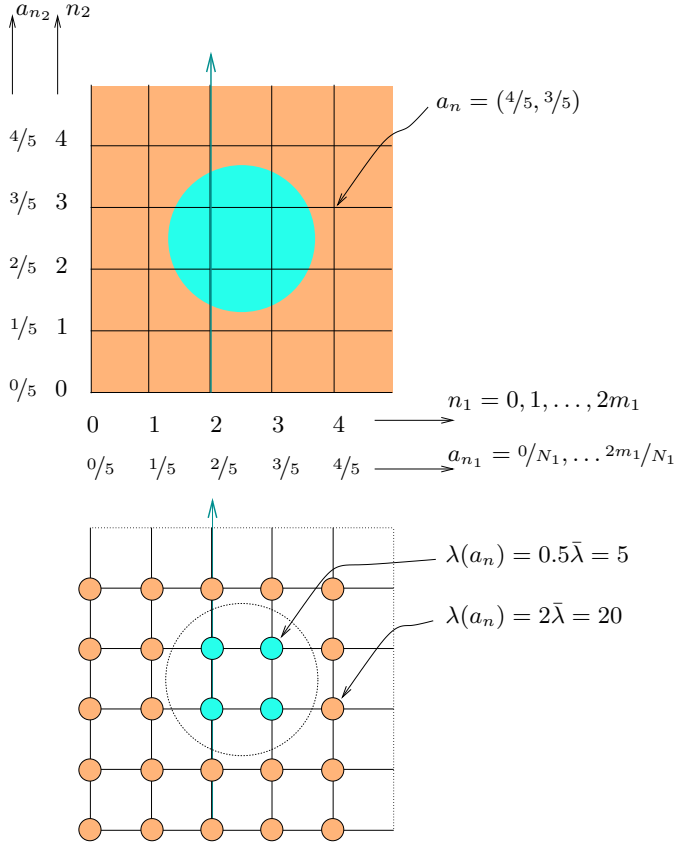


Abbildung 8: Diskretisierung für $m_1 \times m_2 = 2 \times 2$, $N_1 \times N_2 = 5 \times 5$, über dem Schnitt bei $a_{n_1} = m/N = 2/5$ wird später die Steifigkeitsverteilung dargestellt

4.2 EIN PRAKTISCHES BEISPIEL

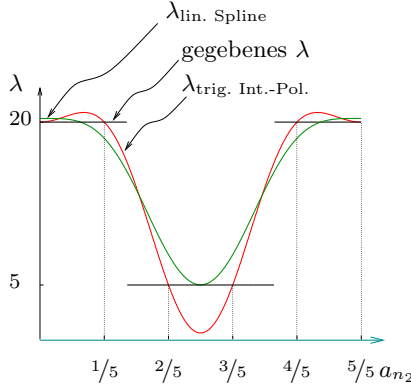


Abbildung 9: λ für akademische Beispiele dargestellt entlang eines Schnitts bei $a_{n_1} = m_1/N_1 = 2/5 = \text{const.}$, zur Lage des Schnitts siehe Abbildung 8

und 10 (jeweils links) dargestellt. Zusätzlich ist der Einfluss der Abminderungsfaktoren in Abbildung 9 dargestellt. Die Lage der Schnittlinie $x = m/N = 2/5 = \text{const.}$ ist in Abbildung 8 dargestellt. Näheres hierzu findet man in [Kaßbohm 05b]. Die Berechnung der Abminderungsfaktoren ist in Abschnitt A.1 beschrieben.

4.1.3 Ergebnisse für die akademischen Beispiele

Dargestellt werden verschiedene Komponenten des Kraftspannungstensors für das erste und für das zweite Beispiel (siehe Tabelle 2). Hierbei wurde die Diskretisierung $m_1 \times m_2 = 2 \times 2$ gewählt. Und Abminderungsfaktoren wurden nicht verwendet.

4.2 Ein praktisches Beispiel

4.2.1 Beschreibung des praktischen Beispiels

Als praktisches Beispiel wird ein repräsentatives Volumenelement untersucht, das aus Zinn (Sn) und Blei (Pb) besteht. Die Steifigkeitsverteilung wird aus [Müller 04] entnommen, wo eine Vergröberung unter Temperatureinfluss untersucht wird. Während der Vergröberung

4 BEISPIELE

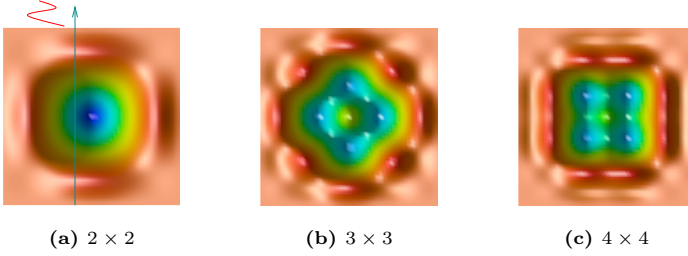


Abbildung 10: Steifigkeitsverteilung $-5.1 < \lambda < 21.7$ für verschiedene Diskretisierungen $m_1 \times m_2$ ohne Abminderungsfaktoren, ganz links die in den akademischen Beispielen verwendete Diskretisierung, Schnitt entlang des Pfeils siehe Abbildung 9

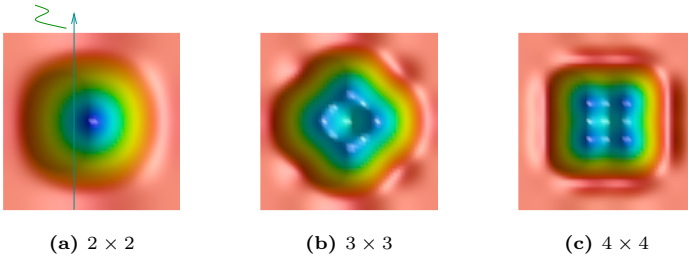
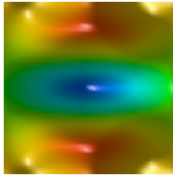
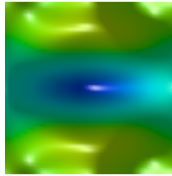


Abbildung 11: $1.6 < \lambda < 20.5$ für verschiedene Diskretisierungen $m_1 \times m_2$ mit Abminderungsfaktoren für lineare Splines, Schnitt entlang des Pfeils siehe Abbildung 9

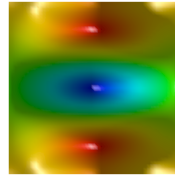
4.2 EIN PRAKTISCHES BEISPIEL



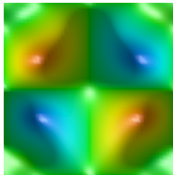
(a) Cosserat



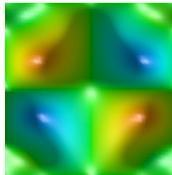
(b) klassisch



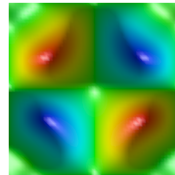
(c) Couple-Stress



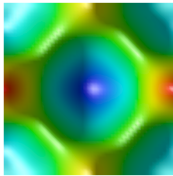
(d) Cosserat



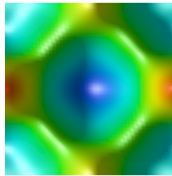
(e) klassisch



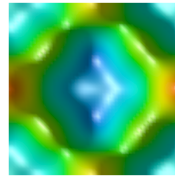
(f) Couple-Stress



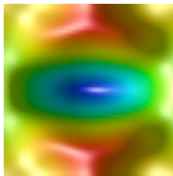
(g) Cosserat



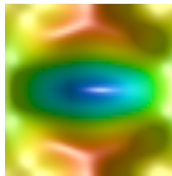
(h) klassisch



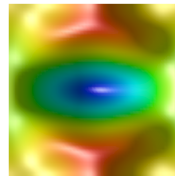
(i) Couple-Stress



(j) Cosserat



(k) klassisch

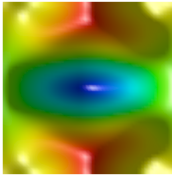


(l) Couple-Stress

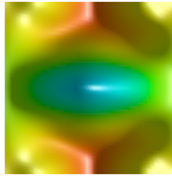
Abbildung 12: Akademisches Beispiel 1:

(a)(b)(c): $11.7 < \tau_{11} < 90.7$; (d)(e)(f): $-9.7 < \tau_{12} < 9.7$
 (g)(h)(i): $7.2 < \tau_{22} < 25.6$; (j)(k)(l): $4.3 < \tau_{33} < 21.5$

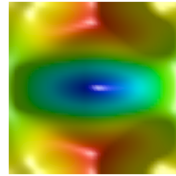
4 BEISPIELE



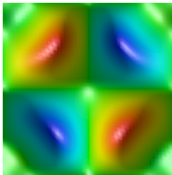
(a) Cosserat



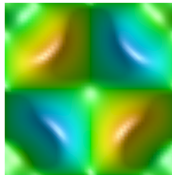
(b) klassisch



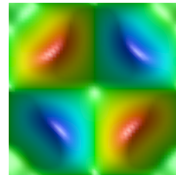
(c) Couple-Stress



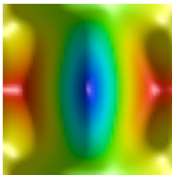
(d) Cosserat



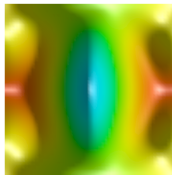
(e) klassisch



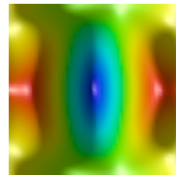
(f) Couple-Stress



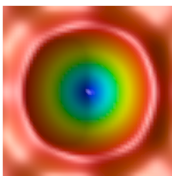
(g) Cosserat



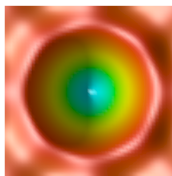
(h) klassisch



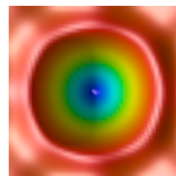
(i) Couple-Stress



(j) Cosserat



(k) klassisch



(l) Couple-Stress

Abbildung 13: Akademisches Beispiel 2:

(a)(b)(c): $-31.3 < \tau_{11} < 4.3$; (d)(e)(f): $-7.6 < \tau_{12} < 7.6$
 (g)(h)(i): $-31.3 < \tau_{22} < 4.3$; (j)(k)(l): $-42.7 < \tau_{33} < -0.7$

4.2 EIN PRAKTISCHES BEISPIEL

bilden sich lamellenartige Strukturen im repräsentativen Volumenelement. Den Bereichen, in denen die Zinn-reiche Phase vorliegt werden die Reinstoff-Materialwerte für Zinn zugeordnet. Den Bereichen, in denen die Blei-reiche Phase vorliegt werden die Reinstoff-Materialwerte für Blei zugeordnet. Der Zinn-Blei-Verbund wird als klassisches Kontinuum untersucht. Um dasselbe Beispiel auch als Cosserat- oder Couple-Stress-Kontinuum zu untersuchen, braucht man die Materialwerte α, β, γ und κ für die Stoffe Blei und Zinn. Diese sind jedoch nicht bekannt. Nur für sehr wenige Stoffe liegen Materialwerte aus Experimenten vor. Einige Beispiele dafür findet man in [Lakes 95].

Material	$E / \frac{10^9 \text{N}}{\text{m}^2}$	$\nu / 1$	$\alpha_T / \frac{10^{-6}}{\text{K}}$	$\lambda / \frac{10^9 \text{N}}{\text{m}^2}$	$\mu / \frac{10^9 \text{N}}{\text{m}^2}$
Sn	50.0	0.36	22.0	47.27	18.38
Pb	16.0	0.44	28.9	40.74	5.56

Tabelle 3: Materialwerte für Zinn (Sn) und Blei (Pb), E , ν und α_T übernommen aus <http://www.webelements.com/>, und es gilt $\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}$ sowie $\mu = \frac{E}{2(1+\nu)}$

Zinn und Blei dehnen sich unter Temperatureinfluss verschieden stark aus. Für beide Stoffe wird angenommen, dass eine Temperaturerhöhung ΔT thermische Dehnungen $\epsilon_{11}^* = \epsilon_{22}^* = \epsilon_{33}^* = \alpha_T \Delta T$ hervorrufen könnten.

Ausgehend von einer Temperaturdifferenz von 300 K ergibt sich als „Belastung“, die dem repräsentativen Volumenelement auferlegt wird:

$$\begin{aligned}
 \epsilon_{11\text{Sn}}^* &= \epsilon_{22\text{Sn}}^* = \epsilon_{33\text{Sn}}^* = \alpha_{T\text{Sn}} \Delta T \\
 &= 22.0 \frac{10^{-6}}{\text{K}} 300 \text{ K} \\
 \epsilon_{11\text{Sn}}^* &= \epsilon_{22\text{Sn}}^* = \epsilon_{33\text{Sn}}^* = 6.60 \cdot 10^{-3} \\
 \epsilon_{11\text{Pb}}^* &= \epsilon_{22\text{Pb}}^* = \epsilon_{33\text{Pb}}^* = 8.67 \cdot 10^{-3} \quad .
 \end{aligned}$$

Alle mittleren Verzerrungen sollen Null sein, d.h.

$$\langle \varepsilon_{ij} \rangle = 0 \quad \forall (i, j) \quad .$$

4 BEISPIELE

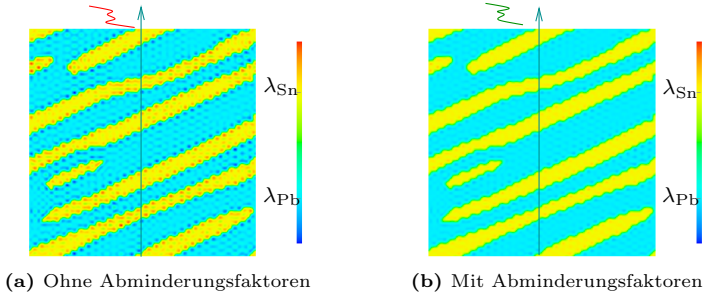


Abbildung 14: Praktisches Beispiel; λ für Diskretisierung $m_1 \times m_2 = 31 \times 31$
 (a) (b): $37.98 \cdot 10^9 \frac{\text{N}}{\text{m}^2} < \lambda < 50.44 \cdot 10^9 \frac{\text{N}}{\text{m}^2}$,
 Schnitt entlang des Pfeils siehe Abbildung 15

Damit liegen die Lasten fest. Als äußere Abmessungen des repräsentativen Volumelements werden gewählt:

$$\begin{aligned} L_1 &= 1 \text{ m} \quad , \\ L_2 &= 1 \text{ m} \quad . \end{aligned}$$

In die dritte Raumrichtung seien alle Größen konstant.

4.2.2 Diskretisierung

Die unstetige Steifigkeitsverteilung wird wieder durch ein trigonometrisches Polynom approximiert. Dies geschieht wieder mit Hilfe der diskreten Fourier Transformation einmal ohne und einmal mit Abminderungsfaktoren. Die Ergebnisse sind in Abbildung 14 dargestellt.

4.2.3 Ergebnisse für das praktische Beispiel

In der Simulation werden Verzerrungen und Spannungen berechnet. Als Ergebnisse werden die Spannungen über dem Ort im repräsentativen Volumelement dargestellt, die verschieden von Null sind (siehe Abbildung 16). Man erkennt, dass sich die Druckspannungen $\tau_{11}, \tau_{22}, \tau_{33}$ für die Simulation ohne und mit Abminderungsfaktoren nur unwesentlich

4.2 EIN PRAKTISCHES BEISPIEL

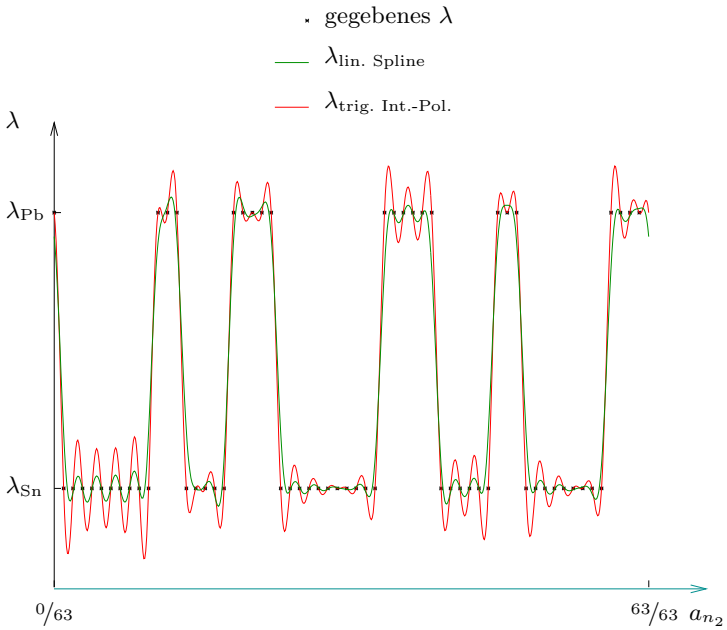


Abbildung 15: λ für praktisches Beispiel dargestellt entlang eines Schnitts bei $a_{n_1} = m_1/N_1 = 31/63 = \text{const.}$, das trigonometrische Interpolationspolynom interpoliert zwischen den gegebenen Werten, das trigonometrische Polynom aus den Koeffizienten des Splines ist eine bessere Approximation, die die gegebenen Werte nicht exakt trifft

4 BEISPIELE

unterscheiden, jedoch nehmen sie - verglichen mit der Bruchspannung - sehr hohe Werte an.

4.3 Grenzen und Größenordnung der Materialwerte

Um numerische Experimente für Cosserat-Kontinua durchzuführen, müssen die Materialkonstanten bekannt sein. Diese werden teilweise aus praktischen Versuche bestimmt, andernteils können sie auch aus Theorien bestimmt werden, in denen die Wechselwirkungen zwischen Atomen modelliert werden. Grenzen für die Materialwerte können mitunter durch theoretische Überlegungen ermittelt oder postuliert werden. So muss, damit die innere Energie für ein Cosserat-Kontinuum nicht negativ werden kann, nach [Eringen 68] gelten:

$$0 \leq 3\lambda + 2\mu + \kappa, \quad 0 \leq 2\mu + \kappa, \quad 0 \leq \gamma, \quad (4.3)$$

$$0 \leq 3\alpha + \beta + \gamma, \quad -\gamma \leq \beta \leq \gamma, \quad 0 \leq \gamma. \quad (4.4)$$

Die Materialwerte müssen durch geeignete Versuche bestimmt werden. In [Lakes 95] wurden zur Ermittlung von Materialkonstanten verschiedene Versuche durchgeführt. Daraus wurden Zahlenwerte für eine Reihe von definierten Größen ermittelt, aus denen die 6 Materialkonstanten $\lambda, \mu, \kappa, \alpha, \beta$ und γ bestimmt werden können. Die resultierenden Materialwerte sind in Tabelle 4 dargestellt.⁷

Material	$\lambda / \frac{10^6 \text{N}}{\text{m}^2}$	$\mu / \frac{10^6 \text{N}}{\text{m}^2}$	$\kappa / \frac{10^6 \text{N}}{\text{m}^2}$	$\alpha / 10^3 \text{N}$	$\beta / 10^3 \text{N}$	$\gamma / 10^3 \text{N}$
Knochen ¹	4000.0	1333.3	5333.3	-0.13	-2.85	3.24
Schaum A ²	797.3	99.7	8.67	-0.026	0.035	0.045
Schaum B ³	2186.8	914.8	228.7	-0.003	0.004	0.004

Tabelle 4: Materialwerte für ausgewählte Stoffe, berechnet mit Daten aus [Lakes 95].

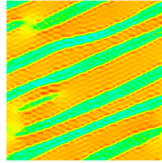
⁷Die meisten der durch Messungen ermittelten Werte genügen der Anforderung, dass die innere Energie nicht negativ werden darf.

¹„Human bone“ in [Lakes 95]

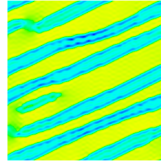
²„Foam, dense polyurethane“ in [Lakes 95]

³„Foam, syntactic“ in [Lakes 95]

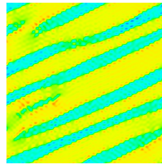
4.3 GRENZEN UND GRÖSSENORDNUNG DER MATERIALWERTE



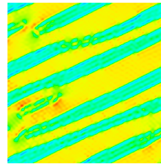
(a) Ohne Abmind.



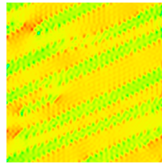
(b) Mit Abmind.



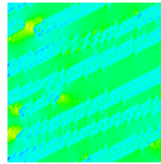
(c) Ohne Abmind.



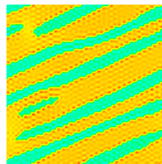
(d) Mit Abmind.



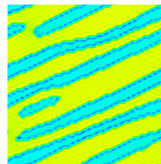
(e) Ohne Abmind.



(f) Mit Abmind.



(g) Ohne Abmind.



(h) Mit Abmind.

Abbildung 16: Praktisches Beispiel: Spannungen ($m_1 \times m_2 = 31 \times 31$)

$$\begin{aligned}
 & \text{(a) (b): } -1.174 \cdot 10^9 \frac{\text{N}}{\text{m}^2} < \tau_{11} < -1.159 \cdot 10^9 \frac{\text{N}}{\text{m}^2} \\
 & \text{(c) (d): } -4.965 \cdot 10^6 \frac{\text{N}}{\text{m}^2} < \tau_{12} < 2.581 \cdot 10^6 \frac{\text{N}}{\text{m}^2} \\
 & \text{(e) (f): } -1.170 \cdot 10^9 \frac{\text{N}}{\text{m}^2} < \tau_{22} < -1.160 \cdot 10^9 \frac{\text{N}}{\text{m}^2} \\
 & \text{(g) (h): } -1.176 \cdot 10^9 \frac{\text{N}}{\text{m}^2} < \tau_{33} < -1.158 \cdot 10^9 \frac{\text{N}}{\text{m}^2}
 \end{aligned}$$

5 Zusammenfassung und Ausblick

5.1 Zusammenfassung

Im Mittelpunkt der Arbeit steht die Behandlung von repräsentativen Volumenelementen mit Fourierreihen. Es werden ausschließlich Randwertprobleme mit periodischen Randbedingungen untersucht. Den repräsentativen Volumenelementen wird eine Materialverteilung zugeordnet, wobei keine Diskontinuitätsflächen berücksichtigt werden. Alle Berechnungen finden im Rahmen der linearen Elastizitätstheorie statt, und es werden drei Typen von Kontinua mit Mikrostruktur untersucht.

Bei der Lösung der Randwertprobleme für die drei Typen von Kontinua stellt sich heraus, dass derselbe Lösungsweg für alle Typen beschritten werden kann und dass sich die Lösungsoperatoren - abhängig von den beherrschenden Differentialgleichungen - auf unterschiedliche Weise im Fourierraum berechnen lassen. Eine Verallgemeinerung des Verfahrens auf beliebige Kontinua mit Mikrostruktur ist also möglich.

Es zeigt sich weiterhin, dass der Fehler im vorgestellten Verfahren einzig auf der Näherung der gegebenen Materialwerte(funktionen) beruht und sich auch zahlenmäßig berechnen lässt.

Mit dem vorgestellten Verfahren ist es möglich, Verzerrungen und Spannungen in repräsentativen Volumenelementen mit beliebiger Steifigkeitsverteilung zu berechnen. Und damit bietet es die Grundlage für viele Problemstellungen auf diesem Gebiet: Beispielsweise können gemittelte Steifigkeiten berechnet werden (Homogenisierung), es können Größeneffekte (size-effects) untersucht werden. Oder die berechneten Spannungen oder Verzerrungen können in der Berechnung einer spannungsabhängigen Diffusion (Vergrößerung) verwendet werden.

5.2 Ausblick

Es gibt ausgehend von dem vorgestellten Verfahren viele verschiedene Richtungen, in die man für die weitere Forschung blicken kann. Zum einen kann das bestehende Verfahren weiter ausgebaut werden, oder es kann für die Lösung praktischer Probleme verwendet werden. Zum anderen kann das vorgestellte Verfahren als ein Weg angesehen werden, den man bei der Berechnung von Kontinua mit Mikrostruktur gehen kann. Und von diesem Weg abweichend können viele andere Zugänge in dieselbe Problematik gefunden werden.

Als weitere Forschungsschwerpunkte, die im Zusammenhang zu dem vorgestellten Thema stehen, bieten sich an:

- Anwendung des vorgelegten Programms zur Berechnung der Spannungen und Verzerrungen in repräsentativen Volumenelementen abhängig von einer gegebenen Steifigkeitsverteilung: Die berechneten Verzerrungen und/oder Spannungen könnten Eingabewerte z.B. für eine Vergrößerungssimulation sein.
- Erweiterung auf inelastische Stoffe: Das bereitgestellte Programm lässt sich erweitern, so dass auch nichtlineares Materialverhalten nachgebildet werden kann. Dies allerdings nur im Rahmen der inkrementellen Plastizität.
- Experimentelle Bestimmung der Materialwerte für Kontinua mit Mikrostruktur: In der aktuellen Forschung im Bereich der Materialtheorie nehmen die Kontinua mit Mikrostruktur eine immer wichtiger Rolle ein. Trotzdem sind konkrete Zahlen für die jeweiligen Materialwerte bisher kaum zu finden.
- Theoretische Überlegungen zum Thema Objektivität bei Kontinua mit Mikrostruktur: Für die Kontinua mit Mikrostruktur müssen ausgehend von mehr oder weniger begründeten Invarianz- und/oder Objektivitätsforderungen neue Materialgesetze „erfunden“ werden. Hierzu findet man in der Literatur bisher nur wenige begründete Ideen.
- Finite Elemente mit Mikrostruktur: Mit der vorgestellten Methode gelingt es nicht, beliebige Randwertproblem zu lösen, sondern man ist auf periodische Randbedingungen beschränkt. Zusätzlich ist man auf lineare Elastizitätstheorie beschränkt. Es wäre sicher sehr interessant, sich von diesen Einschränkungen zu verabschieden und in die nichtlineare Welt zu blicken. Besonders reizvoll scheint mir die Programmierung von Finiten Elementen mit Mikrostruktur.

A Anhang

A.1 Abminderungsfaktoren

Nach Gleichung (3.1c) sind die Fourierkoeffizienten einer Funktion von einer Veränderlichen definiert als:

$$f^\wedge = \frac{1}{2\pi} \int_{x=0}^{2\pi} dx f e^{-\iota k^L x} \quad . \quad (\text{A.1})$$

Und nach Gleichung (3.7b) gilt für die mit der diskreten Fourier Transformation ermittelten Koeffizienten:

$$f_{k^L}^\wedge = \frac{1}{N} \sum_{n=0}^{N-1} f_n e^{-\iota 2\pi k^L a_n} \quad (\text{A.2a})$$

$$= \frac{1}{N} \sum_{n=0}^{N-1} f_n e^{-\iota k^L h n} \quad (\text{A.2b})$$

$$\text{mit } a_n \stackrel{\text{def}}{=} \frac{n}{N} \quad , \quad (\text{A.2c})$$

$$f_n \stackrel{\text{def}}{=} f(x = 2\pi a_n) \quad (\text{A.2d})$$

$$\text{und } h \stackrel{\text{def}}{=} \frac{2\pi}{N} \quad . \quad (\text{A.2e})$$

Die exakten Fourierkoeffizienten nach (A.1) klingen (zumindest für hinreichend glatte Funktionen f) mit wachsender Wellenzahl k^L sehr schnell ab, während die Koeffizienten nach (A.2) N -periodisch sind (also überhaupt nicht abklingen). Im Sinne einer L_2 -Norm sind die Fourierreihen mit den Koeffizienten aus (A.2) meist schlechte Näherungen für f .

Diese Problem lässt sich beheben, indem man f zuerst durch eine Näherung $\mathbf{Q}^N\{f\}$ ersetzt, die nur von den Daten f_n abhängt und dann die Fourierkoeffizienten von $\mathbf{Q}^N\{f\}$ durch die Koeffizienten aus der diskreten Fourier Transformation $f_{k^L}^\wedge$ ausdrückt. Dazu konstruiert man z.B. eine lineare 2π -periodische Hütchenfunktion p gemäß

$$p(x) = \begin{cases} 1 - \frac{x}{h} & \text{falls } 0 < x < h \\ 1 + \frac{x}{h} & \text{falls } 2\pi - h < x < 2\pi \end{cases} \quad (\text{A.3})$$

Und dann berechnet man die Fourierkoeffizienten von $\mathbf{Q}^N\{f\}$:

$$\begin{aligned}
 \{\mathbf{Q}^N\{f\}\}^\wedge &= \frac{1}{2\pi} \int_{x=0}^{2\pi} \mathbf{Q}^N\{f\} e^{-\iota k^L x} dx \\
 &= \frac{1}{2\pi} \sum_{n=0}^{N-1} f_n \int_{x=0}^{2\pi} p(x - nh) e^{-\iota k^L x} dx \\
 &= \frac{1}{2\pi} \sum_{n=0}^{N-1} f_n e^{-\iota k^L nh} \int_{z=0-nh}^{2\pi-nh} p(z) e^{-\iota k^L z} dz \quad (\text{A.4}) \\
 &= \underbrace{\sum_{n=0}^{N-1} f_n e^{-\iota k^L nh}}_{N f_k^\wedge} \underbrace{\frac{1}{2\pi} \int_{x=0}^{2\pi} p(x) e^{-\iota k^L x} dx}_{p^\wedge} \\
 &\stackrel{\text{def}}{=} \alpha f_k^\wedge \quad .
 \end{aligned}$$

Die wellenzahlabhängigen Zahlen α heißen Abminderungsfaktoren. Sie sind Vielfache der Fourierkoeffizienten der Hütchenfunktion. Die Näherung, die die Hütchenfunktion in Kombination mit den diskreten Werten von f hervorbringt, ist ein linearer Spline. Ebenso ließen sich Splines höherer Ordnung konstruieren, die entsprechend andere Abminderungsfaktoren zur Folge hätten. Für die linearen Splines berechnet man weiter:

$$\begin{aligned}
 p^\wedge &= \frac{1}{2\pi} \int_{x=0}^{2\pi} p(x) e^{-\iota k^L x} dx \\
 &= \frac{1}{2\pi} 2 \int_{x=0}^h p(x) \cos(k^L x) dx \\
 &= \frac{1}{\pi} \int_{x=0}^h \left(1 - \frac{x}{h}\right) \cos(k^L x) dx \\
 &= \frac{1}{\pi} \left\{ \left[\left(1 - \frac{x}{h}\right) \frac{1}{k^L} \sin(k^L x) \right]_0^h + \frac{1}{h} \frac{1}{k^L} \int_{x=0}^h \sin(k^L x) dx \right\} \quad (\text{A.5}) \\
 &= \frac{1}{\pi} \left\{ 0 + \frac{1}{h} \frac{1}{k^L} \left[-\cos(k^L x) \right]_0^h \right\} \\
 &= \frac{N}{2\pi^2 k^L} (1 - \cos(k^L \frac{2\pi}{N})) \\
 &= \frac{N}{\pi^2 k^L} \sin^2(k^L \frac{\pi}{N}) \quad .
 \end{aligned}$$

A ANHANG

Und damit ergibt sich für die Abminderungsfaktoren:

$$\alpha = Np^\wedge \quad (\text{A.6a})$$

$$= \left\{ \frac{N}{\pi k^L} \sin(k^L \frac{\pi}{N}) \right\}^2 \quad . \quad (\text{A.6b})$$

Die Erweiterung auf Funktionen von drei Veränderlichen gelingt unmittelbar.

A.2 Neumann Reihe

Die implizite Lösung des Originalproblems war bekannt als (2.16). Dies lässt sich auch schreiben als

$$(\mathbf{I} - \tilde{\mathbf{G}}\tilde{\mathbf{L}}^\gamma)\gamma = \gamma_0 \quad , \quad (\text{A.7})$$

$$\gamma = (\mathbf{I} - \tilde{\mathbf{G}}\tilde{\mathbf{L}}^\gamma)^{-1}\gamma_0 \quad . \quad (\text{A.8})$$

Die Inverse des Operators lässt sich nun als Neumann Reihe darstellen. Es ist (falls die Reihe konvergiert, was hier angenommen wird):

$$(\mathbf{I} - \tilde{\mathbf{G}}\tilde{\mathbf{L}}^\gamma)^{-1} \approx \mathbf{I} + \tilde{\mathbf{G}}\tilde{\mathbf{L}}^\gamma + (\tilde{\mathbf{G}}\tilde{\mathbf{L}}^\gamma)^2 + \dots \quad . \quad (\text{A.9})$$

Und damit erhält man folgende Rekursionsformel für die m -te Näherung bei schon berechneter $(m-1)$ -ter Näherung

$${}^m\gamma = \gamma_0 + \tilde{\mathbf{G}}\tilde{\mathbf{L}}^{\gamma \ m-1}\gamma \quad (\text{A.10})$$

Und wegen (2.14) und (2.15a) ist das

$$\begin{aligned} {}^m\gamma &= \gamma_0 + \tilde{\mathbf{G}}\tilde{\mathbf{L}}^{\gamma \ m-1}\gamma \\ &= \gamma_0 + \tilde{\mathbf{G}}(\tilde{\mathbf{L}}^\gamma - \mathbf{L}^\gamma)^{m-1}\gamma \\ &= \gamma_0 + \tilde{\mathbf{G}}(\tilde{\mathbf{L}}^{\gamma \ m-1}\gamma - \mathbf{L}^{\gamma \ m-1}\gamma) \\ &= \gamma_0 + \tilde{\mathbf{G}}\tilde{\mathbf{L}}^{\gamma \ m-1}\gamma - \tilde{\mathbf{G}}\mathbf{L}^{\gamma \ m-1}\gamma \\ {}^m\gamma &= {}^{m-1}\gamma - \tilde{\mathbf{G}}\mathbf{L}^{\gamma \ m-1}\gamma \quad . \end{aligned} \quad (\text{A.11})$$

A.3 Berechnung von $\bar{g} = \mathcal{F}\{\bar{G}\}$

A.3.1 Berechnung von ω

Ausgangspunkt für die Berechnung des Lösungsoperators im Fourierraum ist das vereinfachte Problem mit konstanten Koeffizienten und gegebener rechter Seite (2.13) formuliert in $w = (u, v)$:

$$\bar{\mathbf{L}}^w w = \tilde{r} \quad (\text{A.12a})$$

$$\gamma = \mathbf{L}^{\gamma w} w \quad (\text{A.12b})$$

$$\langle \gamma \rangle = \gamma_0 \quad (\text{A.12c})$$

Gleichung (A.12a) ausgeschrieben liefert für die drei untersuchten Kontinua:

$$\begin{array}{ll} \text{Cosserat} & \left[\begin{array}{l} (\bar{\lambda} + \bar{\mu})u_{m,km} + (\bar{\mu} + \bar{\kappa})u_{m,ll}\delta_{km} + \bar{\kappa}e_{klm}v_{m,l} = \tilde{p}_k \\ (\bar{\alpha} + \bar{\beta})v_{m,km} + \bar{\gamma}v_{m,ll}\delta_{km} + \bar{\kappa}e_{klm}u_{m,l} - 2\bar{\kappa}v_{m,l}\delta_{km} = \tilde{q}_k \end{array} \right. \\ \text{klassisch} & [(\bar{\lambda} + \bar{\mu})u_{m,km} + \bar{\mu}u_{m,ll}\delta_{km} = \tilde{p}_k \\ \text{Couple-Str.} & \left[\begin{array}{l} (\bar{\lambda} + \bar{\mu} + 1/2\bar{\kappa})u_{m,km} + 1/4\bar{\gamma}u_{m,kmpp} \\ + (\bar{\mu} + 1/2\bar{\kappa})u_{m,pp}\delta_{km} - 1/4\bar{\gamma}u_{m,ppqq}\delta_{km} = \tilde{p}_k \end{array} \right. \end{array} \quad (\text{A.13})$$

Hierbei ist die rechte Seite die verallgemeinerte Störung $\tilde{r} = (\tilde{p}, \tilde{q})$. Diese Gleichungen werden in den Fourierraum übersetzt und liefern Bestimmungsgleichungen für die Fourierkoeffizienten von w bzw. u :

$$\begin{array}{ll} \text{Cosserat} & [\omega_{\kappa\mu}\mathcal{F}\{w_\mu\} = \mathcal{F}\{\tilde{r}_\kappa\} \quad \text{mit } \kappa, \mu \in \{1, 2, \dots, 6\} \\ \text{klassisch} & [\omega_{km}\mathcal{F}\{u_m\} = \mathcal{F}\{\tilde{p}_k\} \quad \text{mit } k, m \in \{1, 2, 3\} \\ \text{Couple-Str.} & [\omega_{km}\mathcal{F}\{u_m\} = \mathcal{F}\{\tilde{p}_k\} \quad \text{mit } k, m \in \{1, 2, 3\} \end{array} \quad (\text{A.14})$$

A ANHANG

Die Werte $\omega_{\kappa\mu}$ können für alle Wellenvektoren (k_1, k_2, k_3) berechnet werden. Es ergibt sich⁸:

$$\begin{aligned}
 \text{Cosserat} \quad & \left[\omega_{\kappa\mu} = \begin{cases} -(\bar{\lambda} + \bar{\mu})k_\kappa k_\mu - (\bar{\mu} + \bar{\kappa})k_l^2 \delta_{\kappa\mu} & \kappa \leq 3, \mu \leq 3 \\ \bar{\kappa} \iota k_l e_{\kappa l \underline{\mu}} & \kappa \leq 3, \mu > 3 \\ \bar{\kappa} \iota k_l e_{\underline{\kappa} l \mu} & \kappa > 3, \mu \leq 3 \\ -(\bar{\alpha} + \bar{\beta})k_{\underline{\kappa}} k_{\underline{\mu}} - \bar{\gamma} k_l^2 \delta_{\underline{\kappa}\underline{\mu}} - 2\bar{\kappa} \delta_{\underline{\kappa}\underline{\mu}} & \kappa > 3, \mu > 3 \end{cases} \right. \\
 \text{klassisch} \quad & \left[\omega_{km} = -(\bar{\lambda} + \bar{\mu})k_k k_m - \bar{\mu} k_l^2 \delta_{km} \right. \\
 \text{Couple-Str.} \quad & \left[\begin{aligned} \omega_{km} = & -(\bar{\lambda} + \bar{\mu} + 1/2\bar{\kappa})k_k k_m + 1/4\bar{\gamma} k_k k_m k_l^2 \\ & -(\bar{\mu} + 1/2\bar{\kappa})k_l^2 \delta_{km} - 1/4\bar{\gamma} k_l^2 k_p^2 \delta_{km} \end{aligned} \right. \end{aligned} \quad (\text{A.15})$$

A.3.2 Berechnung von \bar{g}^ϵ und \bar{g}^κ

Nun lassen sich also die (Fourierkoeffizienten der) Verschiebungen und die Verdrehungen angeben gemäß⁹:

$$\begin{aligned}
 \text{Cosserat} \quad & [\mathcal{F}\{w_\kappa\} = \omega_{\kappa\mu}^{-1} \mathcal{F}\{\tilde{r}_\mu\}] \\
 \text{klassisch} \quad & [\mathcal{F}\{u_\kappa\} = \omega_{km}^{-1} \mathcal{F}\{\tilde{p}_m\}] \\
 \text{Couple-Str.} \quad & [\mathcal{F}\{u_\kappa\} = \omega_{km}^{-1} \mathcal{F}\{\tilde{p}_m\}] \end{aligned} \quad (\text{A.16})$$

Um den Lösungsoperator definiert in (2.15a) im Fourierraum anzugeben, müssen und noch die Verzerrungen abhängig von \tilde{r} berechnet werden. Dazu benutzt man Gleichung (A.12b). Diese lautet ausgeschrieben

⁸Der Unterstrich bedeutet hier, dass der Index um 3 reduziert wird, also z.B.

$\underline{\mu} = \mu - 3$.

⁹Invertierung klappt nur, falls $k \neq 0$.

A.3 BERECHNUNG VON $\bar{g} = \mathcal{F}\{\bar{G}\}$

wie (2.7). Und dies in den Fourierraum übertragen liefert¹⁰:

$$\begin{array}{ll}
 \text{Cosserat} & \left[\begin{array}{l} \mathcal{F}\{\varepsilon_{kl}\} = \iota k_k \mathcal{F}\{u_l\} + e_{lkm} \mathcal{F}\{v_m\} \\ \quad = \iota k_k \mathcal{F}\{w_l\} + e_{lkm} \mathcal{F}\{w_{\bar{m}}\} \\ \quad \mathcal{F}\{\kappa_{kl}\} = \iota k_l \mathcal{F}\{v_k\} \\ \quad \quad = \iota k_l \mathcal{F}\{w_{\bar{k}}\} \end{array} \right. \quad (\text{A.17}) \\
 \text{klassisch} & \left[\mathcal{F}\{\varepsilon_{kl}\} = 1/2(\iota k_k \mathcal{F}\{u_l\} + \iota k_l \mathcal{F}\{u_k\}) \right. \\
 \text{Couple-Str.} & \left[\begin{array}{l} \mathcal{F}\{\varepsilon_{kl}\} = 1/2(\iota k_k \mathcal{F}\{u_l\} + \iota k_l \mathcal{F}\{u_k\}) \\ \mathcal{F}\{\kappa_{kl}\} = -1/2 e_{krs} k_r k_l \mathcal{F}\{u_s\} \end{array} \right.
 \end{array}$$

Setzt man nun (A.16) in (A.17) ein, so folgt¹¹:

$$\begin{array}{ll}
 \text{Cosserat} & \left[\begin{array}{l} \mathcal{F}\{\varepsilon_{kl}\} = \bar{g}_{kl\mu}^{\varepsilon} \mathcal{F}\{\tilde{r}_{\mu}\} \\ \mathcal{F}\{\kappa_{kl}\} = \bar{g}_{kl\mu}^{\kappa} \mathcal{F}\{\tilde{r}_{\mu}\} \end{array} \right. \\
 \text{klassisch} & \left[\mathcal{F}\{\varepsilon_{kl}\} = \bar{g}_{klm}^{\varepsilon} \mathcal{F}\{\tilde{p}_m\} \right. \quad (\text{A.18}) \\
 \text{Couple-Str.} & \left[\begin{array}{l} \mathcal{F}\{\varepsilon_{kl}\} = \bar{g}_{klm}^{\varepsilon} \mathcal{F}\{\tilde{p}_m\} \\ \mathcal{F}\{\kappa_{kl}\} = \bar{g}_{klm}^{\kappa} \mathcal{F}\{\tilde{p}_m\} \end{array} \right.
 \end{array}$$

Die hierzu definierten Größen

$$\begin{array}{ll}
 \text{Cosserat} & \left[\begin{array}{l} \bar{g}_{kl\mu}^{\varepsilon} \stackrel{\text{def}}{=} \iota k_k \omega_{l\mu}^{-1} + e_{lkm} \omega_{\bar{m}\mu}^{-1} \\ \bar{g}_{kl\mu}^{\kappa} \stackrel{\text{def}}{=} \iota k_l \omega_{\bar{k}\mu}^{-1} \end{array} \right. \\
 \text{klassisch} & \left[\bar{g}_{klm}^{\varepsilon} \stackrel{\text{def}}{=} 1/2(k_k \omega_{lm}^{-1} + k_l \omega_{km}^{-1}) \right. \quad (\text{A.19}) \\
 \text{Couple-Str.} & \left[\begin{array}{l} \bar{g}_{klm}^{\varepsilon} \stackrel{\text{def}}{=} 1/2(k_k \omega_{lm}^{-1} + k_l \omega_{km}^{-1}) \\ \bar{g}_{klm}^{\kappa} \stackrel{\text{def}}{=} -1/2 e_{krs} k_r k_l \omega_{sm}^{-1} \end{array} \right.
 \end{array}$$

legen den Lösungsoperator $\mathcal{F}\{\bar{G}\}$ gemäß (2.15a) bzw. (2.20) fest. Die Berechnung dieser „Operatoren“ gelingt nicht für $k = 0$. Hier setzt man

¹⁰Der Überstrich bedeutet hier, dass der Index um 3 erhöht wird, also z.B. $\bar{m} = m + 3$.

¹¹An dieser Stelle sieht man, dass das Problem auch ohne Weiteres in den Spannungen formuliert werden könnte: Die mittleren Spannungen könnte man vorgeben, und statt der Operatoren $\bar{g}_{kl\mu}^{\varepsilon}$ und $\bar{g}_{kl\mu}^{\kappa}$ könnte man hier $g_{kl\mu}^{\tau}$ und $g_{kl\mu}^{\mu}$ berechnen.

die Mittelwerte

$$\begin{array}{ll}
 \text{Cosserat} & \begin{cases} \mathcal{F}\{\varepsilon_{kl}\} = \varepsilon_{0_{kl}} \\ \mathcal{F}\{\kappa_{kl}\} = \kappa_{0_{kl}} \end{cases} \\
 \text{klassisch} & \mathcal{F}\{\varepsilon_{kl}\} = \varepsilon_{0_{kl}} \\
 \text{Couple-Str.} & \mathcal{F}\{\varepsilon_{kl}\} = \varepsilon_{0_{kl}}
 \end{array} \tag{A.20}$$

A.4 Berechnung von $\mathcal{F}\{\mathbb{L}^{\gamma m-1}\gamma\}$

Um die Lösung des gegebenen Problems zu erhalten, muss nun noch $\mathcal{F}\{\mathbb{L}^{\gamma m-1}\gamma\}$ aus Gleichung (2.20) konkret berechnet werden. Der Lösungsoperator ist im letzten Abschnitt berechnet worden. Jetzt geht es um seinen Operanden, also das Objekt, auf das der Operator angewendet wird. Nach Definition in Gleichung (2.6) entspricht das der Berechnung von

$$(\mathcal{F}\{^{m-1}p_k\}, \mathcal{F}\{^{m-1}q_k\}) = \mathcal{F}\{\mathbb{L}^{\gamma m-1}\gamma\} \tag{A.21}$$

wobei für die verschiedenen Kontinua folgende Größen definiert wurden¹²:

$$\begin{array}{ll}
 \text{Cosserat} & \left[\begin{aligned} \mathcal{F}\{p_k^-\} &\stackrel{\text{def}}{=} \mathcal{F}\{(\lambda\varepsilon_{rr}^-\delta_{lk} + (\mu + \kappa)\varepsilon_{lk}^- + \mu\varepsilon_{kl}^- - \varrho\epsilon^*\delta_{lk})_{,l}\} \\ \mathcal{F}\{q_k^-\} &\stackrel{\text{def}}{=} \mathcal{F}\{(\alpha\kappa_{rr}^-\delta_{lk} + \beta\kappa_{lk}^- + \gamma\kappa_{kl}^-)_{,l} \\ &\quad + e_{kmn}(\lambda\varepsilon_{rr}^-\delta_{mn} + (\mu + \kappa)\varepsilon_{mn}^- + \mu\varepsilon_{nm}^-)\} \end{aligned} \right] \\
 \text{klassisch} & \left[\mathcal{F}\{p_k^-\} \stackrel{\text{def}}{=} \mathcal{F}\{(\lambda\varepsilon_{rr}^-\delta_{lk} + 2\mu\varepsilon_{lk}^- - \varrho\epsilon^*\delta_{lk})_{,l}\} \right] \\
 \text{Couple-Str.} & \left[\begin{aligned} \mathcal{F}\{p_k^-\} &\stackrel{\text{def}}{=} \mathcal{F}\{(\lambda\varepsilon_{rr}^-\delta_{lk} + (2\mu + \kappa)\varepsilon_{lk}^- \\ &\quad - \varrho\epsilon^*\delta_{lk} - 1/2e_{rlk}(\beta\kappa_{nr}^- + \gamma\kappa_{rn}^-)_{,n})_{,l}\} \\ \mathcal{F}\{q_k^-\} &\stackrel{\text{def}}{=} \mathcal{F}\{(\beta\kappa_{lk}^- + \gamma\kappa_{kl}^-)_{,l} \\ &\quad + e_{kmn}(\lambda\varepsilon_{rr}^-\delta_{mn} + (2\mu + \kappa)\varepsilon_{mn}^- \\ &\quad - 1/2e_{rmn}(\beta\kappa_{sr}^- + \gamma\kappa_{rs}^-)_{,s})\} \end{aligned} \right]
 \end{array} \tag{A.22}$$

¹²Zur kürzeren Schreibung ist hier $^{m-1}\boxtimes = \boxtimes^-$ benutzt.

A.4 BERECHNUNG VON $\mathcal{F}\{\mathbf{L}^{\gamma m-1}\gamma\}$

Nun schreibt man statt $\mathcal{F}\{\boxtimes\}$ kürzer \boxtimes^\wedge und berechnet die Terme auf der rechten Seite dieser Gleichungen. Dies geschieht mithilfe der

58

A.5 Rekursionsformel mit $\mathcal{F}\{\bar{\mathbf{G}}\}\mathcal{F}\{\mathbf{L}^\gamma{}^{m-1}\gamma\}$

Setzt man nun (A.19) und (A.21) in (2.20) ein, so liefert das:

$$\begin{array}{ll}
 \text{Cosserat} & \begin{cases} {}^m\varepsilon_{kl}^\wedge = {}^{m-1}\varepsilon_{kl}^\wedge - [\bar{g}_{klp}^\varepsilon {}^{m-1}p_p^\wedge + \bar{g}_{kl\bar{p}}^\varepsilon {}^{m-1}q_p^\wedge] \\ {}^m\kappa_{kl}^\wedge = {}^{m-1}\kappa_{kl}^\wedge - [\bar{g}_{klp}^\kappa {}^{m-1}p_p^\wedge + \bar{g}_{kl\bar{p}}^\kappa {}^{m-1}q_p^\wedge] \end{cases} \\
 \text{klassisch} & \begin{cases} {}^m\varepsilon_{kl}^\wedge = {}^{m-1}\varepsilon_{kl}^\wedge - \bar{g}_{klp}^\varepsilon {}^{m-1}p_p^\wedge \end{cases} \\
 \text{Couple-Str.} & \begin{cases} {}^m\varepsilon_{kl}^\wedge = {}^{m-1}\varepsilon_{kl}^\wedge - \bar{g}_{klp}^\varepsilon {}^{m-1}p_p^\wedge \\ {}^m\kappa_{kl}^\wedge = {}^{m-1}\kappa_{kl}^\wedge - \bar{g}_{klp}^\kappa {}^{m-1}p_p^\wedge \end{cases}
 \end{array} \tag{A.24}$$

An Gleichung (A.24) sowie an Gleichung (2.12b) und (2.20) erkennt man, dass die Terme ${}^{m-1}p_p^\wedge$ und ${}^{m-1}q_p^\wedge$ zu Null werden, falls $\langle {}^{m-1}\varepsilon, {}^{m-1}\kappa \rangle$, also die $(m-1)$ -te Näherung, bereits die exakte Lösung eines gegebenen Problems darstellt. In diesem Fall kann die Lösung nicht weiter verbessert werden, was sich darin äußert, dass die Terme in den eckigen Klammern verschwinden. Entsprechendes gilt natürlich auch für das klassische Kontinuum. Damit lässt sich ein Fehler definieren, der ein Maß für die Güte der Näherung darstellt, was im nächsten Abschnitt geschieht.

A.6 Fehler im Bildbereich

Die vorgestellte numerische Methode bringt zwei unterschiedliche Fehler hervor. Zum einen wird die tatsächliche, gegebene Steifigkeitsverteilung durch eine auf einer Diskretisierung basierende endliche Fourierreihe genähert. Dieser Fehler ist ausführlich in [Kaßbohm 05b] diskutiert.

Zum anderen muss die Neumann-Iteration aus praktischen Gründen irgendwann abgebrochen werden. Dies resultiert in einem weiteren Fehler, der soweit reduziert werden kann, bis numerische Rundungsfehler eine weitere Verringerung unmöglich machen. Dieser Fehler wird während jeder Iteration ermittelt. Dazu werden globale absolute Fehler e_p^{abs} und e_q^{abs} sowie relative Fehler e_p und e_q im Bildbereich definiert und mittels

der Parseval-Gleichung berechnet gemäß:

$$\begin{array}{l}
 \text{Cosserrat} \\
 \left[\begin{array}{l}
 e_p^{\text{abs}} \stackrel{\text{def}}{=} \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 \|p\|^2 \\
 = \sum_{a_1=-\infty}^{\infty} \sum_{a_2=-\infty}^{\infty} \sum_{a_3=-\infty}^{\infty} \|p^\wedge\|^2 \\
 e_q^{\text{abs}} \stackrel{\text{def}}{=} \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 \|q\|^2 \\
 = \sum_{a_1=-\infty}^{\infty} \sum_{a_2=-\infty}^{\infty} \sum_{a_3=-\infty}^{\infty} \|q^\wedge\|^2 \\
 e_p \stackrel{\text{def}}{=} \frac{\|L\|^2}{\|\sigma^\wedge(0) + \mu^\wedge(0)\|^2} e_p^{\text{abs}} \\
 e_q \stackrel{\text{def}}{=} \frac{\|L\|^2}{\|\sigma^\wedge(0) + \mu^\wedge(0)\|^2} e_q^{\text{abs}}
 \end{array} \right. \\
 \\
 \text{klassisch} \\
 \left[\begin{array}{l}
 e_p^{\text{abs}} \stackrel{\text{def}}{=} \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 \|p\|^2 \\
 = \sum_{a_1=-\infty}^{\infty} \sum_{a_2=-\infty}^{\infty} \sum_{a_3=-\infty}^{\infty} \|p^\wedge\|^2 \\
 e_p \stackrel{\text{def}}{=} \frac{\|L\|^2}{\|\sigma^\wedge(0)\|^2} e_p^{\text{abs}}
 \end{array} \right. \\
 \\
 \text{Couple-Str.} \\
 \left[\begin{array}{l}
 e_p^{\text{abs}} \stackrel{\text{def}}{=} \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 \|p\|^2 \\
 = \sum_{a_1=-\infty}^{\infty} \sum_{a_2=-\infty}^{\infty} \sum_{a_3=-\infty}^{\infty} \|p^\wedge\|^2 \\
 e_q^{\text{abs}} \stackrel{\text{def}}{=} \frac{1}{8\pi^3 L_1 L_2 L_3} \int_{x_1=0}^{2\pi L_1} dx_1 \int_{x_2=0}^{2\pi L_2} dx_2 \int_{x_3=0}^{2\pi L_3} dx_3 \|q\|^2 \\
 = \sum_{a_1=-\infty}^{\infty} \sum_{a_2=-\infty}^{\infty} \sum_{a_3=-\infty}^{\infty} \|q^\wedge\|^2 \\
 e_p \stackrel{\text{def}}{=} \frac{\|L\|^2}{\|\sigma^\wedge(0) + \mu^\wedge(0)\|^2} e_p^{\text{abs}}
 \end{array} \right.
 \end{array}
 \tag{A.25}$$

B Fortran Programme

Es folgt nun eine Beschreibung der Programme, die es ermöglichen soll, sie fehlerfrei zu verwenden, zu erweitern und zu verbessern. Alle verwendeten Programme sowie die zugehörige Dokumentation können aus der Datei `prog.nw` erzeugt werden (siehe noweb-homepage <http://www.eecs.harvard.edu/~nr/noweb/>), was die Erweiterung der Programme und der Dokumentation sehr einfach möglich macht.

B.1 Struktogramm

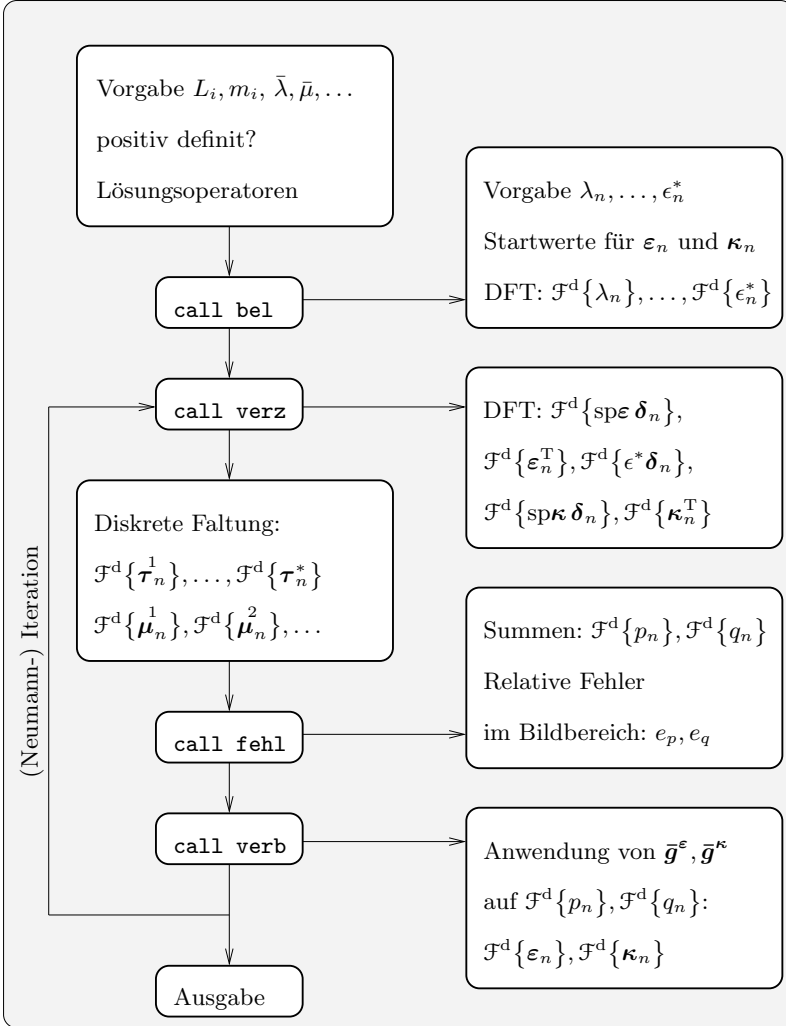


Abbildung 17: Übersicht über die wichtigsten Bestandteile im Programmablauf

B.2 Hauptprogramm prog

Das Hauptprogramm ist aus folgenden Blöcken zusammengesetzt:

```

(prog.f)≡
  program prog
    <Abmessungen und Diskretisierung des Quaders>
    <Variablendeklaration>
    open(18,file='../data/mm_tau')
    open(19,file='../data/er')
    open(20,file='../data/mm_mat')
    <Eröffnung, Materialwerte, Prüfung auf pos. Definitheit>
    <Berechnung der Komponenten des Lösungsoperators>
    <Verteilung der Materialwerte im Quader, Initialisierung>
    <Iteration zur Verbesserung der Verzerrungen, Ausgabe>
  end
  <Einige Funktionen zur Berechnung des Lösungsoperators>

```

Die einzelnen Blöcke werden nun genauer beschrieben, und die wichtigsten Variablen werden in Tabellen erklärt.

Zuerst werden die Abmessungen des repräsentativen Volumenelements definiert, und die Diskretisierung wird festgelegt. Anschließend werden die verwendeten Variablen deklariert.

<Abmessungen und Diskretisierung des Quaders>≡

Variable	Symbol	Kurzbeschreibung	Referenz
m1,m2,m3	m_i	Wellenzahlbereich	(3.8)
N1,N2,N3	N_i	Diskretisierung	(3.7b)
L1,L2,L3	L_i	Längen	(3.1)

<Abmessungen und Diskretisierung des Quaders>+≡

```

  implicit none
  integer m1,m2,m3,N1,N2,N3
  double precision L1,L2,L3
  complex*16 I
  PARAMETER(I=(0.d0,1.d0))
  PARAMETER(L1 = 1.d0, L2 = 1.d0, L3 = 3.d0)
  PARAMETER(
    &      m1 = 2,
    &      m2 = 2,
    &      m3 = 0
C      &      m1 = 31,
C      &      m2 = 31,
C      &      m3 = 0
    &      )
  PARAMETER(N1=2*m1+1, N2=2*m2+1, N3=2*m3+1)

```

<Variablendeklaration>≡

B FORTRAN PROGRAMME

Variable	Symbol	Kurzbeschreibung	Referenz
iter		Iterationszähler	
ikon		Kontinuum-Typ Nr.	
del	δ_{ij}	Funktion	
levi_civita	e_{ijk}	Funktion	
la,mu	$\bar{\lambda}, \bar{\mu}$	mittl. Mat.-werte	
lam,mpk	$\lambda_n, \mu_n + \kappa_n$	Folgen $\lambda_n \in \mathbb{R}$	(2.2)
ep,eq	e_p, e_q	relative Fehler	(A.25)
lam_F,mpk_F	$\mathcal{F}^d\{\lambda_n\}, \dots$	Folgen $\mathcal{F}^d\{\lambda_n\} \in \mathbb{C}$	
ome_1,ome_2,...	ω	Koeffizientenmatrizen	(A.15)
g_e_F,g_k_F	\bar{g}^e, \bar{g}^k	Lösungsoperatoren	(A.19)
e_F,k_F	$\mathcal{F}^d\{\varepsilon_n\}, \dots$	Verzerrungen	
e_del_F,k_del_F	$\mathcal{F}^d\{\text{sp}\varepsilon \delta_n\}, \dots$		(A.23)
e_T_F,k_T_F	$\mathcal{F}^d\{\varepsilon_n^T\}, \dots$		(A.23)
tau_1_F,mut_1_F	$\mathcal{F}^d\{\tau_n\}, \dots$		(A.23)
tau_stern_1_F...	$\mathcal{F}^d\{\tau_n^*\}$		(A.23)
p_F,q_F	$\mathcal{F}^d\{p_n\}$	„Fehler“ $\in \mathbb{C}$	(A.23),(A.25)

```

<Variablendeklaration>+≡
  integer iter,ikon,del,levi_civita,
    &i1,i2,i3,k1,k2,k3,k,l,m,N,o
  double precision
    & la,mu,ka,al,be,ga, KL(3),min,max,
    & mint(3,3),maxt(3,3),minf(3,3),maxf(3,3),
    & ome_NN,ome_L,KKLL,
    & lam(0:N1-1,0:N2-1,0:N3-1),
    & mue(0:N1-1,0:N2-1,0:N3-1),
    & zmu(0:N1-1,0:N2-1,0:N3-1),
    & mpk(0:N1-1,0:N2-1,0:N3-1),
    & zpk(0:N1-1,0:N2-1,0:N3-1),
    & alp(0:N1-1,0:N2-1,0:N3-1),
    & bet(0:N1-1,0:N2-1,0:N3-1),
    & gam(0:N1-1,0:N2-1,0:N3-1),
    & ep,eq
  complex*16
    & lam_F(-m1:m1,-m2:m2,-m3:m3),
    & mue_F(-m1:m1,-m2:m2,-m3:m3),
    & zmu_F(-m1:m1,-m2:m2,-m3:m3),
    & mpk_F(-m1:m1,-m2:m2,-m3:m3),
    & zpk_F(-m1:m1,-m2:m2,-m3:m3),
    & alp_F(-m1:m1,-m2:m2,-m3:m3),
    & bet_F(-m1:m1,-m2:m2,-m3:m3),

```


B.2 HAUPTPROGRAMM prog

```

&      gam_F(-m1:m1,-m2:m2,-m3:m3),
C
&      ome_1(6,6),ome_2(3,3),ome_3(3,3), ome_NK,
&      b_1(6,1), b_2(3,1), b_3(3,1),addi,
C
&      g_e_F      (-m1:m1,-m2:m2,-m3:m3,3,3,6),
&      g_k_F      (-m1:m1,-m2:m2,-m3:m3,3,3,6),
&      e_stern_F(-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_F        (-m1:m1,-m2:m2,-m3:m3,3,3),
&      k_F        (-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_del_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_stern_del_F (-m1:m1,-m2:m2,-m3:m3,3,3),
&      k_del_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_T_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      k_T_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      tau_1_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      tau_2_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      tau_3_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      tau_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      tau_stern_1_F(-m1:m1,-m2:m2,-m3:m3,3,3),
&      tau_stern_2_F(-m1:m1,-m2:m2,-m3:m3,3,3),
&      mat_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      mut_1_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      mut_2_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      mut_3_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      mut_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      p_F        (-m1:m1,-m2:m2,-m3:m3,3),
&      q_F        (-m1:m1,-m2:m2,-m3:m3,3)

```

Jede Berechnung wird immer für alle untersuchten Kontinuumstypen durchgeführt (40). Zur Berechnung der Lösungsoperatoren werden “mittlere” Materialwerte definiert (50), und positive Definitheit wird abgeprüft (60) nach [Eringen 68].

(*Eröffnung, Materialwerte, Prüfung auf pos. Definitheit*) \equiv

```

      do i1=1,3
      do i2=1,3
          mint(i1,i2)= 1.d20
          maxt(i1,i2)=-1.d20
      enddo
    enddo

40  do ikon=1,1
50  if (ikon.eq.1) then
      la=12.5d0
      mu=12.5d0
      ka=12.5d3
      al=12.5d0
      be=12.5d0
      ga=12.5d0
    else if (ikon.eq.2) then
C-----praktisches Beispiel

```

B FORTRAN PROGRAMME

```

C          la=(47.27d9 + 40.74d9)/2.d0
C          mu=(18.38d9 + 5.56d9)/2.d0
C-----akad. Beispiel
          la=12.5d0
          mu=la
          ka=0.d0
          al=0.d0
          be=0.d0
          ga=0.d0
          else if (ikon.eq.3) then
            la=12.5d0
            mu=la
            ka=la
            al=la
            be=la
            ga=la
          endif

60  if ( ( 3.d0*la + 2.d0*mu + ka .lt.0 ) .or.
      &   ( 2.d0*mu + ka .lt.0 ) .or.
      &   (ka.lt.0) .or.
      &   (3.d0*al + be + ga.lt.0) .or.
      &   (be.gt.ga) .or.
      &   (ga.lt.0) ) then
      write(*,*)'Materialkonstanten fehlerhaft.'
      stop
    endif

    Im folgenden Block werden die Einträge von  $\omega$  gemäß (A.15) be-
    rechnet (70). Diese Matrix wird dann invertiert, und damit werden die
    Einträge von  $\bar{g}_{kl\mu}^\varepsilon, \bar{g}_{kl\mu}^\kappa$  gemäß (A.19) usw. berechnet (80).

    <Berechnung der Komponenten des Lösungsoperators>≡
    write(*,*)'Berechnung des Lösungsoperators...'
    do k1 = -m1, m1
      do k2 = -m2, m2
        do k3 = -m3, m3
          if (.not.(K1.eq.0 .and. K2.eq.0 .and.K3.eq.0)) then
            KL(1) = dble(k1)/L1
            KL(2) = dble(k2)/L2
            KL(3) = dble(k3)/L3
70      do k = 1, 3
          do l = 1, 3
            ome_1(k,l) = (la + mu) * ome_NN(k,l,KL)
            &             + (mu + ka) * ome_L(k,l,KL)
            ome_2(k,l) = (la + mu) * ome_NN(k,l,KL)
            &             + mu * ome_L(k,l,KL)
            ome_3(k,l) =
            &             (la + mu + 0.5d0 * ka ) * ome_NN(k,l,KL)
            &             + 0.25d0 * ga * KL(k) * KL(l) * KKLL(KL)
            &             + (mu + 0.5d0 * ka) * ome_L(k,l,KL)
            &             + 0.25d0 * ga * KKLL(KL) * ome_L(k,l,KL)

```

B.2 HAUPTPROGRAMM prog

```

      enddo
      do l = 4, 6
        ome_1(k,l) = ka * ome_NK(k,l-3,KL)
      enddo
    enddo
    do k = 4, 6
      do l = 1, 3
        ome_1(k,l)= ka * ome_NK(k-3,l,KL)
      enddo
      do l = 4, 6
        ome_1(k,l)=
&          (al + be) * ome_NN(k-3,l-3,KL)+
&          ga      * ome_L(k-3,l-3,KL)-
&          2 * ka   * del(k-3,l-3)
      enddo
    enddo
80  if (ikon.eq.1) then
      call gaussj(ome_1,6,6,b_1,1,1)
      do k = 1, 3
        do l = 1, 3
          do N = 1, 6
            addi=(0.d0,0.d0)
            do m = 1, 3
              addi=addi
&              +levi_civita(l,k,m) * ome_1(m+3,N)
            enddo
            g_e_F(k1,k2,k3,k,l,N)=
&              I * KL(k) * ome_1(l,N) + addi
            g_k_F(k1,k2,k3,k,l,N)=
&              I * KL(l) * ome_1(k+3,N)
          enddo
        enddo
      enddo
    else if (ikon.eq.2) then
      call gaussj(ome_2,3,3,b_2,1,1)
      do k = 1, 3
        do l = 1, 3
          do n = 1, 3
            g_e_F(k1,k2,k3,k,l,n)=
&              0.5d0 * I *(
&              KL(k) * ome_2(l,n)+
&              KL(l) * ome_2(k,n) )
            g_k_F(k1,k2,k3,k,l,n)=(0.d0,0.d0)
          enddo
          do n = 4, 6
            g_e_F(k1,k2,k3,k,l,n)=(0.d0,0.d0)
            g_k_F(k1,k2,k3,k,l,n)=(0.d0,0.d0)
          enddo
        enddo
      enddo
    else if (ikon.eq.3) then
      call gaussj(ome_3,3,3,b_3,1,1)

```

B FORTRAN PROGRAMME

```

      do k = 1, 3
        do l = 1, 3
          do n = 1, 3
            g_e_F(k1,k2,k3,k,l,n)=
&              0.5d0 * I *(
&              KL(k) * ome_3(l,n)+
&              KL(l) * ome_3(k,n) )
            addi=(0.d0,0.d0)
            do m = 1, 3
              do o = 1, 3
                addi=addi
&              +levi_civita(k,m,o)*KL(m)
&              * KL(l)*ome_3(o,n)
              enddo
            enddo
            g_k_F(k1,k2,k3,k,l,n)=
&              -0.5d0 * addi
            enddo
            do n = 4, 6
              g_e_F(k1,k2,k3,k,l,n)=(0.d0,0.d0)
              g_k_F(k1,k2,k3,k,l,n)=(0.d0,0.d0)
            enddo
          enddo
        enddo
      endif
    else
      do k = 1, 3
        do l = 1, 3
          do N = 1, 6
            g_e_F(k1,k2,k3,k,l,N)=(0.d0,0.d0)
            g_k_F(k1,k2,k3,k,l,N)=(0.d0,0.d0)
          enddo
        enddo
      enddo
    endif
  enddo
enddo
write(*,*)'fertig.'
do k1 = -m1, m1
  do k2 = -m2, m2
    do k3 = -m3, m3
      do i1 = 1,3
        do i2 = 1,3
          e_F      (k1,k2,k3,i1,i2)=(0.d0,0.d0)
          e_T_F    (k1,k2,k3,i1,i2)=(0.d0,0.d0)
          e_del_F  (k1,k2,k3,i1,i2)=(0.d0,0.d0)
          k_F      (k1,k2,k3,i1,i2)=(0.d0,0.d0)
          k_T_F    (k1,k2,k3,i1,i2)=(0.d0,0.d0)
          k_del_F  (k1,k2,k3,i1,i2)=(0.d0,0.d0)
          tau_1_F  (k1,k2,k3,i1,i2)=(0.d0,0.d0)
          tau_2_F  (k1,k2,k3,i1,i2)=(0.d0,0.d0)

```

B.2 HAUPTPROGRAMM prog

```

tau_3_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
mut_1_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
mut_2_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
mut_3_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
e_stern_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
e_del_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
e_stern_del_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
    enddo
  enddo
enddo
enddo
enddo

```

Im folgenden Block wird die örtliche Verteilung der Materialwerte festgelegt (95, siehe B.4).

(Verteilung der Materialwerte im Quader, Initialisierung)≡

```

95  call bel(la,mu,ka,al,be,ga,e_F,k_F,e_stern_F,
    &lam_F,mue_F,zmu_F,mpk_F,zpk_F,alp_F,bet_F,gam_F,
    &lam, mue, zmu, mpk, zpk, alp, bet, gam,
    &m1,m2,m3,N1,N2,N3)

```

Im folgenden Block findet die Neumann-Iteration statt. Zuerst werden aus den verallgemeinerten Verzerrungen **e_F** und **k_F** die Größen **e_T_F**, **e_del_F**, **k_T_F**, **k_del_F** berechnet (110, siehe B.5). Schließlich werden diverse diskrete Faltungen berechnet (130), um **tau_1_F** usw. zu bestimmen. Nun sind die Kraft- und die Momentenspannungen bekannt, und es werden die Fehlervektoren **p_F** und **q_F** bestimmt (140). Mit **p_F** und **q_F** und den Lösungsoperatoren werden dann die verbesserten verallgemeinerten Verzerrungen berechnet (150, siehe B.8) und die Iteration beginnt von neuem (160), wenn die Fehler nicht unter einer vorgegebenen Schranke liegen. Andernfalls wird eine entsprechende Ausgabe erzeugt.

(Iteration zur Verbesserung der Verzerrungen, Ausgabe)≡

```

    ep  = 17.d0
    iter = 0
    write(*,*)'Iteration beginnt:'
C      write(*,*)'Iteration beginnt nicht.'
C      goto 170
100  if ((ep.gt.1.d-7).or.(eq.gt.1.d-7)) then
        write(19,*)iter, ep
        iter = iter +1
        write(*,*)'Iteration ', iter
110   call verz(e_F,e_T_F,e_del_F,k_F,k_T_F,k_del_F,
    &      e_stern_F,e_stern_del_F,
    &      N1,N2,N3,m1,m2,m3)

```

B FORTRAN PROGRAMME

```

130      if (ikon.eq.1) then
           call falt(lam_F,e_del_F,tau_1_F,N1,N2,N3,m1,m2,m3)
           call falt(mpk_F,   e_F,tau_2_F,N1,N2,N3,m1,m2,m3)
           call falt(mue_F,  e_T_F,tau_3_F,N1,N2,N3,m1,m2,m3)
           call falt(alp_F,k_del_F,mut_1_F,N1,N2,N3,m1,m2,m3)
           call falt(bet_F,   k_F,mut_2_F,N1,N2,N3,m1,m2,m3)
           call falt(gam_F,  k_T_F,mut_3_F,N1,N2,N3,m1,m2,m3)
           else if (ikon.eq.2) then
               write(*,*)'tau1 tau2 berechnen...'
               call falt(lam_F,e_del_F,tau_1_F,N1,N2,N3,m1,m2,m3)
               call falt(zmu_F,   e_F,tau_2_F,N1,N2,N3,m1,m2,m3)
               write(*,*)'fertig.'
           else if (ikon.eq.3) then
               call falt(lam_F,e_del_F,tau_1_F,N1,N2,N3,m1,m2,m3)
               call falt(zpk_F,   e_F,tau_2_F,N1,N2,N3,m1,m2,m3)
               call falt(bet_F,   k_F,mut_1_F,N1,N2,N3,m1,m2,m3)
               call falt(gam_F,  k_T_F,mut_2_F,N1,N2,N3,m1,m2,m3)
           endif

C      Thermo:
           write(*,*)'tau*1 tau*2 berechnen...'
           call falt(lam_F,e_stern_del_F,tau_stern_1_F,N1,N2,N3,m1,m2,m3)
           call falt(zpk_F,e_stern_F,tau_stern_2_F,N1,N2,N3,m1,m2,m3)
           write(*,*)'fertig.'

C      Ende

           write(*,*)'Fehler berechnen...'
140      call fehl(
           &      ikon,ep,eq,p_F,q_F,
           &      tau_1_F,tau_2_F,tau_3_F,
           &      tau_stern_1_F,tau_stern_2_F,
           &      mut_1_F,mut_2_F,mut_3_F,
           &      tau_F,mut_F,
           &      L1,L2,L3,m1,m2,m3)
C      write(*,*) ep, eq
           write(*,*)'fertig.'

           write(*,*)'Verbesserung berechnen...'
150      call verb(p_F,q_F,e_F,k_F,g_e_F,g_k_F,m1,m2,m3)
           write(*,*)'fertig.'
160      goto 100
           else
               write(*,*)'fertig.'

C      write(*,*)'Ausgabedateien erzeugen...'
C      call fr(ikon,mut_F,1,1,N1,N2,N3,m1,m2,m3,'-2-11',min,max)
C      minf(1,1)=min
C      maxf(1,1)=max
C      write(18,9002)1,1,min,max

```

```

C          write(*,*)'fertig.'

C      Ende der Abfrage nach Fehler:
endif

C      Ende der Schleife über ikont:
enddo

do i1=1,3
  do i2=1,3
    write(*,*) i1,i2
    if (abs(tau_F(0,0,0,i1,i2)).gt.1.d-5) then
      write(*,*)'tau: ', abs(tau_F(0,0,0,i1,i2))
    endif
    if (abs(mut_F(0,0,0,i1,i2)).gt.1.d-5) then
      write(*,*)'mut: ', abs(mut_F(0,0,0,i1,i2))
    endif
    write(*,*)
  enddo
enddo
9000 format(2F25.8)
9002 format(2I5, 2F25.8)

170 continue

```

B.3 Weitere Programmteile zur Berechnung des Lösungsoperators

Es folgen nun einige Funktionen, die sich selbst erklären, sowie ein Unterprogramm zur Berechnung der Inversen nach Numerical Recipes für Fortran ([Press 92]), modifiziert für Matrizen mit komplexen Einträgen (90).

(Einige Funktionen zur Berechnung des Lösungsoperators) ≡

```

double precision function ome_NN(k,l,KL)
  implicit none
  double precision KL(3)
  integer k,l
  ome_NN = - KL(k) * KL(l)
  return
end

double precision function ome_L(k,l,KL)
  implicit none
  integer k,l,del
  double precision KL(3)
  double precision KKLL

```

B FORTRAN PROGRAMME

```
ome_L = - KKLL(KL) * del(k,l)
return
end

double precision function KKLL(KL)
implicit none
double precision KL(3)
KKLL = KL(1) * KL(1) + KL(2) * KL(2) + KL(3) * KL(3)
return
end

complex*16 function ome_NK(k,l,KL)
implicit none
integer levi_civita, m,k,l
double precision KL(3)
complex*16 I
PARAMETER(I=(0.d0,1.d0))
ome_NK=(0.d0,0.d0)
do m=1,3
    ome_NK=ome_NK+ I * levi_civita(k,m,l) * KL(m)
enddo
return
end

integer function del(i,j)
implicit none
integer i,j
if ( (i.eq.1 .and. j .eq. 1)
& .or. (i.eq.2 .and. j .eq. 2)
& .or. (i.eq.3 .and. j .eq. 3)) then
    del=1
else
    del=0
endif
return
end

integer function levi_civita(i,k,n)
implicit none
integer i,k,n
if(
& ( (i.eq.1).and.(k.eq.2).and.(n.eq.3) )
& .or.
& ( (i.eq.3).and.(k.eq.1).and.(n.eq.2) )
& .or.
& ( (i.eq.2).and.(k.eq.3).and.(n.eq.1) ) )
& then
    levi_civita = 1
elseif(
& ( (i.eq.1).and.(k.eq.3).and.(n.eq.2) )
& .or.
& ( (i.eq.2).and.(k.eq.1).and.(n.eq.3) )
```


B.3 WEITERE PROGRAMMTEILE

```

&      .or.
&      ( (i.eq.3).and.(k.eq.2).and.(n.eq.1) ) )
&      then
      levi_civita = - 1
    else
      levi_civita = 0
    endif
    return
  end
90  SUBROUTINE gaussj(a,n,np,b,m,mp)
    INTEGER m,mp,n,np,NMAX
    complex*16 a(np,np),b(np,mp)
    PARAMETER (NMAX=50)
    INTEGER i,icol,irow,j,k,l,ll,indx(NMAX),indxr(NMAX),ipiv(NMAX)
    complex*16 big,dum,pivinv
    do 11 j=1,n
      ipiv(j)=0
11   continue
    do 22 i=1,n
      big=0.
      do 13 j=1,n
        if(ipiv(j).ne.1)then
          do 12 k=1,n
            if (ipiv(k).eq.0) then
              if (abs(a(j,k)).ge.abs(big))then
                big=abs(a(j,k))
                irow=j
                icol=k
              endif
            else if (ipiv(k).gt.1) then
              pause 'singular matrix in gaussj'
            endif
          continue
12        endif
13        continue
      ipiv(icol)=ipiv(icol)+1
      if (irow.ne.icol) then
        do 14 l=1,n
          dum=a(irow,l)
          a(irow,l)=a(icol,l)
          a(icol,l)=dum
14        continue
        do 15 l=1,m
          dum=b(irow,l)
          b(irow,l)=b(icol,l)
          b(icol,l)=dum
15        continue
      endif
      indxr(i)=irow
      indx(i)=icol
      if (a(icol,icol).eq.0.) pause 'singular matrix in gaussj'
      pivinv=1./a(icol,icol)

```

B FORTRAN PROGRAMME

```

      a(icol,icol)=1.
      do 16 l=1,n
        a(icol,l)=a(icol,l)*pivinv
16     continue
      do 17 l=1,m
        b(icol,l)=b(icol,l)*pivinv
17     continue
      do 21 ll=1,n
        if(ll.ne.icol)then
          dum=a(ll,icol)
          a(ll,icol)=0.
          do 18 l=1,n
            a(ll,l)=a(ll,l)-a(icol,l)*dum
18          continue
          do 19 l=1,m
            b(ll,l)=b(ll,l)-b(icol,l)*dum
19          continue
        endif
      21 continue
22    continue
      do 24 l=n,1,-1
        if(indxr(l).ne.indxc(1))then
          do 23 k=1,n
            dum=a(k,indx(1))
            a(k,indx(1))=a(k,indxc(1))
            a(k,indxc(1))=dum
23          continue
        endif
      24 continue
      return
      END
```

B.4 Unterprogramm bel

In diesem Unterprogramm wird die diskrete Verteilung der Materialwerte festgelegt (200), und hierfür werden die Fourierkoeffizienten mit der diskreten Fourier Transformation (siehe Abschnitt 3.2) berechnet (300) und an das Hauptprogramm zurückgegeben.

Ausgehend von den “mittleren” Materialwerten (1a,mu, usw.), mit denen die Lösungsoperatoren berechnet wurden, wird hier eine zweiphasige Zusammensetzung in der repräsentativen Zelle angenommen (siehe 4). Im Innern der zylindrischen Faser sind alle Materialwerte halb so groß wie die “mittleren” Materialwerte, außerhalb davon sind sie doppelt so groß wie sie¹³. Der Mittelpunkt und der relative Radi-

¹³Dies ist natürlich nur ein Beispiel. Hier kann eine beliebige Verteilung der Materialwerte an diskreten Punkten vorgegeben werden. Außerdem ist man natürlich

B.4 UNTERPROGRAMM `bel`

us der Faser wird bei 210 bzw. bei 220 festgelegt. Bei 230 werden die verallgemeinerten Verzerrungen initialisiert, und an jeder Stelle werden die nach (A.20) vorgegebenen Mittelwerte als Vorschätzung angesetzt (240). Die Fourierkoeffizienten der vorgeschätzten Verzerrungen werden ebenfalls berechnet (310) und an das Hauptprogramm zurückgegeben.

```

<bel.f>≡
      subroutine bel(la,mu,ka,al,be,ga,e_F,k_F,e_stern_F,
&lam_F,mue_F,zmu_F,mpk_F,zpk_F,alp_F,bet_F,gam_F,
&lam, mue, zmu, mpk, zpk, alp, bet, gam,
&m1,m2,m3,N1,N2,N3)
      implicit none
      integer al,a2,a3,m1,m2,m3,N1,N2,N3,i1,i2,
& mat(0:62, 0:62, 0:0)
      double precision
& xf,yf,cxf,cyf,raf,axf,ayf,bef,
& lam(0:N1-1,0:N2-1,0:N3-1),
& mue(0:N1-1,0:N2-1,0:N3-1),
& zmu(0:N1-1,0:N2-1,0:N3-1),
& mpk(0:N1-1,0:N2-1,0:N3-1),
& zpk(0:N1-1,0:N2-1,0:N3-1),
& alp(0:N1-1,0:N2-1,0:N3-1),
& bet(0:N1-1,0:N2-1,0:N3-1),
& gam(0:N1-1,0:N2-1,0:N3-1),
& e_stern(0:N1-1, 0:N2-1, 0:N3-1, 1:3, 1:3),
& e (0:N1-1, 0:N2-1, 0:N3-1, 1:3, 1:3),
& k (0:N1-1, 0:N2-1, 0:N3-1, 1:3, 1:3),
& la,mu,ka,al,be,ga,
& la_1,mu_1,ka_1,al_1,be_1,ga_1,
& la_2,mu_2,ka_2,al_2,be_2,ga_2,
& delta_la, delta_mu
      complex*16
& lam_F(-m1:m1,-m2:m2,-m3:m3),
& mue_F(-m1:m1,-m2:m2,-m3:m3),
& zmu_F(-m1:m1,-m2:m2,-m3:m3),
& mpk_F(-m1:m1,-m2:m2,-m3:m3),
& zpk_F(-m1:m1,-m2:m2,-m3:m3),
& alp_F(-m1:m1,-m2:m2,-m3:m3),
& bet_F(-m1:m1,-m2:m2,-m3:m3),
& gam_F(-m1:m1,-m2:m2,-m3:m3),
& e_stern_F(-m1:m1,-m2:m2,-m3:m3, 3,3),
& e_F (-m1:m1,-m2:m2,-m3:m3, 3,3),
& k_F (-m1:m1,-m2:m2,-m3:m3, 3,3)
      open(22,file='../data/mat_schnitt')

C      -----Beginn akademisches Problem:
C C      2-Stoff-System
      la_1 = 0.4d0*la
      la_2 = 1.6d0*la

```

auch nicht gezwungen, sich auf isotropes Material zu beschränken.

B FORTRAN PROGRAMME

```

mu_1 = 0.4d0*mu
mu_2 = 1.6d0*mu

ka_1 = 0.4d0*ka
ka_2 = 1.6d0*ka

al_1 = 0.4d0*al
al_2 = 1.6d0*al

be_1 = 0.4d0*be
be_2 = 1.6d0*be

ga_1 = 0.4d0*ga
ga_2 = 1.6d0*ga

C      Schleife über alle Punkte im RVE
200 do a1 = 0, N1-1
      xf = dble(a1) / N1
      do a2 = 0, N2-1
        yf = dble(a2) / N2
        do a3 = 0, N3-1
          cxf=0.5d0
          cyf=0.5d0
          raf=0.25d0
220      axf=cxf-xf
          ayf=cyf-yf
          bef=dsqrt(axf**2.d0+ayf**2.d0)
          if (bef.le.raf) then
            lam(a1,a2,a3)= la_1
            mue(a1,a2,a3)= mu_1
            zmu(a1,a2,a3)= 2.d0*mu_1
            mpk(a1,a2,a3)= mu_1+ka_1
            zpk(a1,a2,a3)= 2.d0*mu_1+ka_1
            alp(a1,a2,a3)= al_1
            bet(a1,a2,a3)= be_1
            gam(a1,a2,a3)= ga_1
          else
            lam(a1,a2,a3)= la_2
            mue(a1,a2,a3)= mu_2
            zmu(a1,a2,a3)= 2.d0*mu_2
            mpk(a1,a2,a3)= mu_2+ka_2
            zpk(a1,a2,a3)= 2.d0*mu_2+ka_2
            alp(a1,a2,a3)= al_2
            bet(a1,a2,a3)= be_2
            gam(a1,a2,a3)= ga_2
          endif
230      do i1 =1,3
        do i2 = 1,3
          e(a1,a2,a3,i1,i2)      =0.d0
          k(a1,a2,a3,i1,i2)      =0.d0
          e_stern(a1,a2,a3,i1,i2)=0.d0
        
```

B.5 UNTERPROGRAMM verz

```

                                enddo
                                enddo
C -----
C Experiment 2:
C Vorgabe der Thermo-Dehnungen:
C     if (bef.le.raf) then
C         e_stern(a1,a2,a3,1,1)=1.d0
C         e_stern(a1,a2,a3,2,2)=1.d0
C         e_stern(a1,a2,a3,3,3)=1.d0
C     endif
C Thermo Ende

C Vorgabe der konstanten Verzerrungen (und auch Startwert):
240     e(a1,a2,a3,1,1)=1.d0
        k(a1,a2,a3,1,1)=0.d0
        enddo
        enddo
        enddo
C-----Ende akad. Problem
C
C
C
        write(*,*)'Fourierkoeffizienten berechnen...'
300    call fk_s(lam,lam_F,N1,N2,N3,m1,m2,m3)
        call fk_s(mue,mue_F,N1,N2,N3,m1,m2,m3)
        call fk_s(zmu,zmu_F,N1,N2,N3,m1,m2,m3)
        call fk_s(mpk,mpk_F,N1,N2,N3,m1,m2,m3)
        call fk_s(zpk,zpk_F,N1,N2,N3,m1,m2,m3)
        call fk_s(alp,alp_F,N1,N2,N3,m1,m2,m3)
        call fk_s(bet,bet_F,N1,N2,N3,m1,m2,m3)
        call fk_s(gam,gam_F,N1,N2,N3,m1,m2,m3)
310    call fk(e,e_F,N1,N2,N3,m1,m2,m3)
        call fk(k,k_F,N1,N2,N3,m1,m2,m3)
C Thermo
        call fk(e_stern,e_stern_F,N1,N2,N3,m1,m2,m3)
        write(*,*)'fertig.'
        end

```

B.5 Unterprogramm verz

In diesem Unterprogramm werden aus den vorgegebenen Verzerrungstensoren $\mathcal{F}^d\{\epsilon_n\}$ (e_F) und $\mathcal{F}^d\{\kappa_n\}$ (k_F) und aus den vorgegebenen Eigendehnungen $\mathcal{F}^d\{\epsilon_n^*\}$ (e_stern_F) verschiedene Terme berechnet, die zur Berechnung von $\mathcal{F}^d\{p_n\}$ und $\mathcal{F}^d\{q_n\}$ notwendig sind (siehe Gleichungen (A.23)). Dies sind z.B. $\mathcal{F}^d\{sp\epsilon\delta_n\}$ (e_del_F), $\mathcal{F}^d\{\epsilon_n^T\}$ (e_T_F) usw. .

$\langle verz.f \rangle \equiv$
subroutine verz(e_F,e_T_F,e_del_F,k_F,k_T_F,k_del_F,

B FORTRAN PROGRAMME

```

&e_stern_F,e_stern_del_F,
&N1,N2,N3,m1,m2,m3)

implicit none
integer N1,N2,N3,m1,m2,m3
complex*16
&      e_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_T_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_del_F(-m1:m1,-m2:m2,-m3:m3,3,3),
&      k_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      k_T_F    (-m1:m1,-m2:m2,-m3:m3,3,3),
&      k_del_F(-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_stern_F      (-m1:m1,-m2:m2,-m3:m3,3,3),
&      e_stern_del_F(-m1:m1,-m2:m2,-m3:m3,3,3),
&      sp_e_stern_F,
&      sp_e_F, sp_k_F

integer k1,k2,k3,i1,i2

do k1 =-m1, m1
  do k2 =-m2, m2
    do k3 =-m3, m3
      do i1 =1, 3
        do i2 =1, 3
          e_del_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
          k_del_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
          e_stern_del_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
          e_T_F(k1,k2,k3,i2,i1)=e_F(k1,k2,k3,i1,i2)
          k_T_F(k1,k2,k3,i2,i1)=k_F(k1,k2,k3,i1,i2)
        enddo
      enddo
      sp_e_F=
&      e_F(k1,k2,k3,1,1)+
&      e_F(k1,k2,k3,2,2)+
&      e_F(k1,k2,k3,3,3)
      sp_k_F=
&      k_F(k1,k2,k3,1,1)+
&      k_F(k1,k2,k3,2,2)+
&      k_F(k1,k2,k3,3,3)
C      Thermo
      sp_e_stern_F=
&      e_stern_F(k1,k2,k3,1,1)+
&      e_stern_F(k1,k2,k3,2,2)+
&      e_stern_F(k1,k2,k3,3,3)
      e_stern_del_F(k1,k2,k3,1,1)=sp_e_stern_F
      e_stern_del_F(k1,k2,k3,2,2)=sp_e_stern_F
      e_stern_del_F(k1,k2,k3,3,3)=sp_e_stern_F
C      Ende
      e_del_F(k1,k2,k3,1,1)=sp_e_F
      e_del_F(k1,k2,k3,2,2)=sp_e_F
      e_del_F(k1,k2,k3,3,3)=sp_e_F

```

B.6 UNTERPROGRAMM `falt`

```

      k_del_F(k1,k2,k3,1,1)=sp_k_F
      k_del_F(k1,k2,k3,2,2)=sp_k_F
      k_del_F(k1,k2,k3,3,3)=sp_k_F
    enddo
  enddo
enddo
end

```

B.6 Unterprogramm `falt`

In diesem Unterprogramm wird die diskrete Faltung nach Abschnitt 3.3.2 berechnet. Eingang ist jeweils ein die Folge von skalaren Werten (`c_F`) sowie eine Folge von neun skalaren Komponenten eines Tensors (`d_F`). Aus diesen beiden Eingaben kann man die diskrete Faltung gemäß (3.10) berechnen. Ausgabe ist der Tensor `h_F`. 1, 2 ist die Schleife über die Tensorkomponenten. 3,4,5 ist die Schleife über die Wellenzahlen. 6 ist die Initialisierung. 7,8,9 ist die Schleife der Faltung.

```

<falt.f>≡
  subroutine falt(c_F,d_F,h_F,N1,N2,N3,m1,m2,m3)
    implicit none
  C   in:
  C   c_F , d_F    FK der Funktionen
  C
  C   out:
  C   h[k1, k2, k3]    FK von c mal d,  also ( c d )^ = c^ * d^

    integer N1,N2,N3,m1,m2,m3,k1,k2,k3,i1,i2,p1,p2,p3,j1,j2,j3
    complex*16 I,
    &   c_F(-m1:m1,-m2:m2,-m3:m3),
    &   d_F(-m1:m1,-m2:m2,-m3:m3, 3, 3),
    &   h_F(-m1:m1,-m2:m2,-m3:m3, 3, 3)
  1   do i1=1,3
  2     do i2=1,3
  3       do k1=-m1, m1
  4         do k2=-m2, m2
  5           do k3=-m3, m3
  6             h_F(k1,k2,k3,i1,i2)=(0.d0,0.d0)
  7             do p1=-m1, m1
  8               j1=int((- (k1-p1)+N1+(N1-1)/2)/N1)*N1
  9               + (k1-p1)-N1
  10              do p2=-m2, m2
  11                j2=int((- (k2-p2)+N2+(N2-1)/2)/N2)*N2
  12                + (k2-p2)-N2
  13              do p3=-m3, m3
  14                j3=int((- (k3-p3)+N3+(N3-1)/2)/N3)*N3
  15                + (k3-p3)-N3
  16              h_F(k1,k2,k3,i1,i2)=h_F(k1,k2,k3,i1,i2)
  17              +c_F(p1,p2,p3)*d_F(j1,j2,j3,i1,i2)

```

```

enddo
enddo
enddo
enddo
enddo
enddo
enddo
enddo
enddo
end
```

B.7 Unterprogramm fehl

In diesem Unterprogramm wird der Fehler im Bildbereich (siehe Abschnitt A.6) berechnet und ausgegeben. Dies geschieht für jeden Schritt in der (Neumann-) Iteration.

1, 2, 3 ist die Schleife über die Wellenzahlen. 4, 5 ist die Schleife über die Tensorkomponenten von $\mathcal{F}^d\{\boldsymbol{\tau}_n\}$ bzw. $\mathcal{F}^d\{\boldsymbol{\mu}_n\}$. Sind diese Tensoren (bzw. deren Folgen im Fourierraum) berechnet, so lassen sich die Fehlervektoren $\mathcal{F}^d\{p_n\}$ und $\mathcal{F}^d\{q_n\}$ berechnen. Grundlage bieten die Formeln ab (A.23). Für jedes Kontinuum werden die Fehlervektoren verschieden berechnet (6, 7, 8). Aus den Fehlervektoren lässt sich ein relativer Gesamtfehler berechnen (9). Grundlage bietet Formel (A.25) in Abschnitt A.6.

```

<fehl.f>≡
      subroutine fehl(
         &ikon,ep,eq,p_F,q_F,
         &tau_1_F,tau_2_F,tau_3_F,
         &tau_stern_1_F,tau_stern_2_F,
         &mut_1_F,mu_2_F,mu_3_F,
         &tau_F,mu_F,
         &L1,L2,L3,m1,m2,m3)

      implicit none
      integer ikon,k1,k2,k3,m1,m2,m3,i1,i2,i3,levi_civita,n,
         & p,q,r,s
      double precision L1,L2,L3,KL(3),sum_p,sum_q,n_L_quad,ee(3,3),
         & n_nue_quad,ep,eq,n_p_quad,n_q_quad
      complex*16 I,
         & tau_F (-m1:m1,-m2:m2,-m3:m3,3,3),
         & mu_F (-m1:m1,-m2:m2,-m3:m3,3,3),
         & tau_1_F (-m1:m1,-m2:m2,-m3:m3,3,3),
         & tau_2_F (-m1:m1,-m2:m2,-m3:m3,3,3),
         & tau_3_F (-m1:m1,-m2:m2,-m3:m3,3,3),
         & tau_stern_1_F (-m1:m1,-m2:m2,-m3:m3,3,3),
         & tau_stern_2_F (-m1:m1,-m2:m2,-m3:m3,3,3),
         & mu_1_F (-m1:m1,-m2:m2,-m3:m3,3,3),

```


B.7 UNTERPROGRAMM fehl

```

&      mut_2_F (-m1:m1,-m2:m2,-m3:m3,3,3),
&      mut_3_F (-m1:m1,-m2:m2,-m3:m3,3,3),
&      p_F      (-m1:m1,-m2:m2,-m3:m3,3),
&      p_1_F    (-m1:m1,-m2:m2,-m3:m3,3),
&      p_2_F    (-m1:m1,-m2:m2,-m3:m3,3),
&      q_F      (-m1:m1,-m2:m2,-m3:m3,3),
&      q_1_F    (-m1:m1,-m2:m2,-m3:m3,3),
&      q_2_F    (-m1:m1,-m2:m2,-m3:m3,3),
&      nue_F(3,3),
&      t_quad,m_quad
PARAMETER(I=(0.d0,1.d0))

n_p_quad = 0.d0
n_q_quad = 0.d0
1  do k1 = -m1, m1
2    do k2 = -m2, m2
3      do k3 = -m3, m3
          KL(1) = dble(k1)/L1
          KL(2) = dble(k2)/L2
          KL(3) = dble(k3)/L3
4      do i1 = 1,3
5        do i2 = 1,3
            tau_F(k1,k2,k3,i1,i2)=
&              tau_1_F(k1,k2,k3,i1,i2)+
&              tau_2_F(k1,k2,k3,i1,i2)+
&              tau_3_F(k1,k2,k3,i1,i2)
C      Thermo
&              -tau_stern_1_F(k1,k2,k3,i1,i2)
&              -tau_stern_2_F(k1,k2,k3,i1,i2)
C      Ende
            mut_F(k1,k2,k3,i1,i2)=
&              mut_1_F(k1,k2,k3,i1,i2)+
&              mut_2_F(k1,k2,k3,i1,i2)+
&              mut_3_F(k1,k2,k3,i1,i2)
          enddo
        enddo
      do n = 1,3
          p_F (k1,k2,k3,n) = (0.d0,0.d0)
          p_1_F(k1,k2,k3,n) = (0.d0,0.d0)
          p_2_F(k1,k2,k3,n) = (0.d0,0.d0)
          q_F (k1,k2,k3,n) = (0.d0,0.d0)
          q_1_F(k1,k2,k3,n) = (0.d0,0.d0)
          q_2_F(k1,k2,k3,n) = (0.d0,0.d0)
6        if (ikon.eq.1) then
            do p = 1,3
                p_F(k1,k2,k3,n) = p_F(k1,k2,k3,n)
&                  + I * KL(p) * tau_F(k1,k2,k3,p,n)
                q_1_F(k1,k2,k3,n)= q_1_F(k1,k2,k3,n)
&                  + I * KL(p) * mut_F(k1,k2,k3,p,n)
                do q = 1,3
                    q_2_F(k1,k2,k3,n)=q_2_F(k1,k2,k3,n)
&                  +levi_civita(n,p,q) *

```

B FORTRAN PROGRAMME

```

&          tau_F(k1,k2,k3,p,q)
&          enddo
&          enddo
&          q_F(k1,k2,k3,n) =
7          q_1_F(k1,k2,k3,n)+q_2_F(k1,k2,k3,n)
&          else if (ikon.eq.2) then
&          do p = 1,3
&          p_F(k1,k2,k3,n) = p_F(k1,k2,k3,n)
&          + I * KL(p) * tau_F(k1,k2,k3,p,n)
&          enddo
8          else if (ikon.eq.3) then
&          do r=1,3
&          do s=1,3
&          ee(r,s)=0.d0
&          do p=1,3
&          do q=1,3
&          ee(r,s)=ee(r,s)+
&          levi_civita(r,p,q)*levi_civita(s,p,q)
&          enddo
&          enddo
&          enddo
&          do p = 1,3
&          p_1_F(k1,k2,k3,n) = p_1_F(k1,k2,k3,n)
&          + I * KL(p) * tau_F(k1,k2,k3,p,n)
&          q_1_F(k1,k2,k3,n) = q_1_F(k1,k2,k3,n)
&          + I * KL(p) * mut_F(k1,k2,k3,p,n)
&          do q = 1,3
&          q_2_F(k1,k2,k3,n)=q_2_F(k1,k2,k3,n)
&          +levi_civita(n,p,q) *
&          tau_F(k1,k2,k3,p,q)
&          - I * KL(p)/2.d0 * ee(n,q)*
&          mut_F(k1,k2,k3,p,q)
&          do r = 1,3
&          p_2_F(k1,k2,k3,n) = p_2_F(k1,k2,k3,n)
&          + 0.5d0 * levi_civita(q,p,n)*
&          KL(p)*KL(r)*
&          mut_F(k1,k2,k3,r,q)
&          enddo
&          enddo
&          enddo
&          p_F(k1,k2,k3,n) =
&          p_1_F(k1,k2,k3,n)+p_2_F(k1,k2,k3,n)
&          q_F(k1,k2,k3,n) =
&          q_1_F(k1,k2,k3,n)+q_2_F(k1,k2,k3,n)
&          end if
&          enddo
&          do n=1,3
&          n_p_quad = n_p_quad +
&          abs(p_F(k1,k2,k3,n))*abs(p_F(k1,k2,k3,n))
&          n_q_quad = n_q_quad +
&          abs(q_F(k1,k2,k3,n))*abs(q_F(k1,k2,k3,n))

```

B.8 UNTERPROGRAMM verb

```

                                enddo
                                enddo
                                enddo
                                enddo
                                do i1 = 1,3
                                  do i2 = 1,3
                                    nue_F(i1,i2)=tau_F(0,0,0,i1,i2)+mut_F(0,0,0,i1,i2)
                                  enddo
                                enddo

                                n_nue_quad=0.d0
                                do i1 = 1,3
                                  do i2 = 1,3
                                    n_nue_quad=n_nue_quad +
&                                abs(nue_F(i1,i2))*abs(nue_F(i1,i2))
                                  enddo
                                enddo
                                n_L_quad = L1*L1 + L2*L2 + L3*L3
                                ep = n_p_quad * n_L_quad / n_nue_quad
                                eq = n_q_quad * n_L_quad / n_nue_quad

C      Berechnung des Quadrats der Norm
C      vom Mittelwert der Spannungen:
                                t_quad=0.d0
                                m_quad=0.d0
                                do i1 = 1,3
                                  do i2 = 1,3
                                    t_quad = t_quad +
&                                tau_F(0,0,0,i1,i2) * tau_F(0,0,0,i1,i2)
                                    m_quad = m_quad +
&                                mut_F(0,0,0,i1,i2) * mut_F(0,0,0,i1,i2)
                                  enddo
                                enddo

                                write(*,*)ep,eq,ikon,abs(tau_F(0,0,0,1,1)),abs(mut_F(0,0,0,1,1))
                                write(*,*)'mitt: ',sqrt(abs(t_quad)), sqrt(abs(m_quad))
                                end

```

B.8 Unterprogramm verb

In diesem Unterprogramm wird eine Verbesserung der Verzerrungen e_F und k_F berechnet. Dazu wird der Lösungsoperator (g_{e_F} , g_{k_F}) auf die Fehlervektoren (p_F , q_F) angewendet (8,9). Die verbesserten Verzerrungen werden mittels (A.11) berechnet (10,11). 1,2 ist die Schleife über die Tensorkomponenten. 3,4,5 ist die Schleife über die Wellenzahlen. 6,7 ist die Initialisierung der Korrekturterme.

B FORTRAN PROGRAMME

```

<verb.f>≡
      subroutine verb(p_F,q_F,e_F,k_F,g_e_F,g_k_F,m1,m2,m3)
      implicit none
      integer k1,k2,k3,i1,i2,m1,m2,m3,n
      complex*16
      &      g_e_F(-m1:m1,-m2:m2,-m3:m3,3,3,6),
      &      g_k_F(-m1:m1,-m2:m2,-m3:m3,3,3,6),
      &      e_F  (-m1:m1,-m2:m2,-m3:m3,3,3),
      &      k_F  (-m1:m1,-m2:m2,-m3:m3,3,3),
      &      p_F  (-m1:m1,-m2:m2,-m3:m3,3),
      &      q_F  (-m1:m1,-m2:m2,-m3:m3,3),
      &      korr_e_F, korr_k_F
C      aaa
1      do i1 = 1,3
2          do i2 = 1,3
3              do k1 = -m1,m1
4                  do k2 = -m2,m2
5                      do k3 = -m3,m3
7                          if (.not.
                              &      ((k1.eq.0).and.(k2.eq.0).and.(k3.eq.0))
                              &      ) then
8                              korr_e_F=(0.d0,0.d0)
8                              korr_k_F=(0.d0,0.d0)
8                              do n = 1,3
8                                  korr_e_F = korr_e_F +
8                                      g_e_F(k1,k2,k3,i1,i2,n)
8                                      *p_F(k1,k2,k3,n) +
8                                      g_e_F(k1,k2,k3,i1,i2,n+3)
8                                      *q_F(k1,k2,k3,n)
9                                  korr_k_F = korr_k_F +
8                                      g_k_F(k1,k2,k3,i1,i2,n)
8                                      *p_F(k1,k2,k3,n) +
8                                      g_k_F(k1,k2,k3,i1,i2,n+3)
8                                      *q_F(k1,k2,k3,n)
8                                  enddo
10                                 e_F(k1,k2,k3,i1,i2)=e_F(k1,k2,k3,i1,i2)-korr_e_F
11                                 k_F(k1,k2,k3,i1,i2)=k_F(k1,k2,k3,i1,i2)-korr_k_F
8                                  endif
8                                  enddo
8                              enddo
8                              enddo
8                              enddo
8                              enddo
8                              end

```

B.9 Unterprogramme fk_s und fk

In diesen Unterprogrammen wird die diskrete Fourier Transformation einer skalar- bzw. einer tensorwertigen Funktion berechnet. Eingabe ist

B.9 UNTERPROGRAMME **fk_s** UND **fk**

die (reelle) Zahlenfolge **K**. Ausgabe die (komplexe) Zahlenfolge **K_F**. Die Berechnung geschieht gemäß Gleichung (3.7).

```
(fk_s.f)≡
subroutine fk_s(K,K_F,N1,N2,N3,m1,m2,m3)
implicit none
c   IN:
integer N1,N2,N3,m1,m2,m3
double precision K(0:N1-1,0:N2-1,0:N3-1)
c   local:
integer k1,k2,k3,a1,a2,a3
double precision expo, scale, ab1,ab2,ab3
double precision PIPI, PI
complex*16 I
PARAMETER (PIPI=6.283185307179586d0)
PARAMETER (PI=PIPI/2.d0)
PARAMETER (I=(0.d0, 1.d0))
c   OUT:
complex*16 K_F(-m1:m1,-m2:m2,-m3:m3)
scale=dblE(N1)*N2*N3
do k1= -m1,m1
  if (k1.eq.0) then
    ab1=1.d0
  else
    ab1=((N1/PI/dblE(k1)
&      *dsin(k1*PI/dblE(N1))))**2.d0
  endif
  do k2= -m2,m2
    if (k2.eq.0) then
      ab2=1.d0
    else
      ab2=((N2/PI/dblE(k2)
&      *dsin(k2*PI/dblE(N2))))**2.d0
    endif
    do k3= -m3,m3
      if (k3.eq.0) then
        ab3=1.d0
      else
        ab3=((N3/PI/dblE(k3)
&      *dsin(k3*PI/dblE(N3))))**2.d0
      endif
      K_F(k1,k2,k3) =(0.d0,0.d0)
      do a1=0, N1-1
        do a2=0, N2-1
          do a3=0, N3-1
            expo = dblE(k1)*a1/N1
&      + dblE(k2)*a2/N2
&      + dblE(k3)*a3/N3
            K_F(k1,k2,k3)
&      = K_F(k1,k2,k3)
&      + K(a1,a2,a3)*
&      exp(-I*PIPI*expo)
```

B FORTRAN PROGRAMME

```

                                enddo
                                enddo
                                enddo
C   Ohne Abminderungsfaktoren:
                                K_F(k1,k2,k3)= K_F(k1,k2,k3)/scale
C   Mit Abminderungsfaktoren für lineare Splines:
C   K_F(k1,k2,k3)= K_F(k1,k2,k3)*ab1*ab2*ab3/scale
                                enddo
                                enddo
                                enddo
                                end

```

$\langle f_{k,f} \rangle \equiv$

```

subroutine fk(K,K_F,N1,N2,N3,m1,m2,m3)
implicit none
c   IN:
integer N1,N2,N3,m1,m2,m3
double precision K(0:N1-1,0:N2-1,0:N3-1,3,3)
c   local:
integer k1,k2,k3,a1,a2,a3,i1,i2
double precision expo, scale, ab1,ab2,ab3
double precision PIPI, PI
complex*16 I
PARAMETER (PIPI=6.283185307179586d0)
PARAMETER (PI=PIPI/2.d0)
PARAMETER (I=(0.d0, 1.d0))
c   OUT:
complex*16 K_F(-m1:m1,-m2:m2,-m3:m3,3,3)
scale=dbl(N1)*N2*N3
do i1 = 1,3
  do i2 = 1,3
    do k1= -m1,m1
      if (k1.eq.0) then
        ab1=1.d0
      else
        ab1=((N1/PI/dbl(k1)
&          *dsin(k1*PI/dbl(N1))))**2.d0
      endif
      do k2= -m2,m2
        if (k2.eq.0) then
          ab2=1.d0
        else
          ab2=((N2/PI/dbl(k2)
&          *dsin(k2*PI/dbl(N2))))**2.d0
        endif
        do k3= -m3,m3
          if (k3.eq.0) then
            ab3=1.d0
          else
            ab3=((N3/PI/dbl(k3)
&          *dsin(k3*PI/dbl(N3))))**2.d0
          endif

```

B.10 UNTERPROGRAMM **fr**

```

c          init:
          K_F(k1,k2,k3,i1,i2) =(0.d0,0.d0)
          do a1=0, N1-1
            do a2=0, N2-1
              do a3=0, N3-1
                expo = dble(k1)*a1/N1
                &      + dble(k2)*a2/N2
                &      + dble(k3)*a3/N3
                K_F(k1,k2,k3,i1,i2)
                &      = K_F(k1,k2,k3,i1,i2)
                &      + K(a1,a2,a3,i1,i2)*
                &      exp(-I*PIPI*expo)
              enddo
            enddo
          enddo
C      Ohne Abminderungsfaktoren:
          K_F(k1,k2,k3,i1,i2)
          &      = K_F(k1,k2,k3,i1,i2)/scale
C      Mit Abminderungsfaktoren für lineare Splines:
C      K_F(k1,k2,k3,i1,i2)
C      &      = K_F(k1,k2,k3,i1,i2)*ab1*ab2*ab3/scale
C
C
C
C      if (abs(K_F(k1,k2,k3,1,1)).gt.1.e-5)
C      &      write(*,*)k1,k2,k3,K_F(k1,k2,k3,1,1)
          enddo
        enddo
      enddo
    enddo
  enddo
end

```

B.10 Unterprogramm **fr**

In diesem Unterprogramm wird die Fourierreihe bzw. die inverse diskrete Fourier Transformation nach Gleichung (3.8) berechnet. Auch zwischen den Stützstellen können die Funktionswerte des trigonometrischen Interpolationspolynoms berechnet werden.

Die berechneten Werte werden in Dateien ausgegeben.

$\langle fr.f \rangle \equiv$

```

subroutine fr(ikon,K_F,i1,i2,N1,N2,N3,m1,m2,m3,zk2,min,max)
implicit none

character*5 zk2
integer m1,m2,m3,N1,N2,N3,i1,i2,ikon,xn,yn
complex*16 I

```

B FORTRAN PROGRAMME

```

double precision PIPI,min,max,dbet,dx,dy
PARAMETER (I=(0.d0, 1.d0))
PARAMETER (PIPI=6.283185307179586d0)

character*20 zk
integer k1,k2,k3,p1,p2,p3
double precision xx,yy,out
complex*16 K_F(-m1:m1,-m2:m2,-m3:m3,3,3)

if (ikon.eq.0) then
    zk='../data/0'//zk2
else if (ikon.eq.1) then
    zk='../data/1'//zk2
else if (ikon.eq.2) then
    zk='../data/2'//zk2
else if (ikon.eq.3) then
    zk='../data/3'//zk2
endif
write(*,*)zk, ' wird geschrieben'
open(17,file=zk)
C    open(19,file='../data/schnitt')

min= 1.d20
max= -1.d20

xn=-1
C    do xx=0.d0, PIPI + 0.00001d0, PIPI/630.d0
do xx=0.d0, PIPI + 0.00001d0, PIPI/63.d0
    xn=xn+1
    yn=-1
C    do yy=0.d0, PIPI + 0.00001d0, PIPI/630.d0
do yy=0.d0, PIPI + 0.00001d0, PIPI/63.d0
    yn=yn+1

C    Fejersche Mittel -----
C    do p1= m1,0,-1
C    do p2= m2,0,-1
C    do p3= m3,0,-1
C    do k1= -p1,p1
C    do k2= -p2,p2
C    do k3= -p3,p3
C    Fejersche Mittel Ende-----

C    ohne Mittelung:
    out=0.d0
    do k1= -m1,m1
    do k2= -m2,m2
    do k3= -m3,m3
        out=out+K_F(k1,k2,k3,i1,i2)
&        *exp(I*(k1*xx+k2*yy))

```


B.10 UNTERPROGRAMM **fr**

```

        enddo
        enddo
    enddo
    write(17,6000)xx,yy,out

    if(out.lt.min)then
        min=out
    endif
    if(out.gt.max)then
        max=out
    endif
C      Fejersche Mittel -----
C      enddo
C      enddo
C      enddo
C      out = out / ( (m1+1)*(m2+1)*(m3+1) )
C      Fejersche Mittel Ende-----
        enddo
C      write(17,*)
        enddo

6000 format(3F25.8)
end

```

LITERATURVERZEICHNIS

Literaturverzeichnis

- [Brigola 97] R. Brigola. *Fourieranalysis, Distributionen und Anwendungen*. Vieweg, 1997.
- [Cailletaud 03] G. Cailletaud, S. Forest *et al.* *Some elements of microstructural mechanics*. Computational Materials Science 27, pp. 351-374, 2003.
- [Capriz 89] G. Capriz. *Continua with microstructure*. Springer tracts in natural philosophy, 35, 1989.
- [Chang 05] C. S. Chang & Q. Shi. *Mix-Mode Elastic Finite Element Formulation for Bonded Granular Material Considering Rotation of Particles*. J. of Engng. Mech., Vol. 131, No. 2, pp. 120-130, 2005.
- [Cosserat 09] E. Cosserat & F. Cosserat. *Théorie des Corps Déformables*. Paris: A. Hermann & Fils, 1909.
- [Courant 68] R. Courant & D. Hilbert. *Methoden der Mathematischen Physik I*. Springer, New York, 1968.
- [Eagle 28] A. Eagle. *On the Relations between the Fourier Constants of a Periodic Function and the Coefficients determined by Harmonic Analysis*. Phil. Mag., S. 7., vol. 5, no. 27, pp. 113-132, 1928.
- [Eringen 68] A. C. Eringen. *Fracture*. Vol 2, pp. 621-729, Acad. Press, edited by R. Liebowitz, 1968.
- [Eyre 99] D. J. Eyre & G. W. Milton. *A fast numerical scheme for computing the response of composites using grid refinement*. Eur. Phys. J. AP 6, 41, 1999.
- [Forest 99] S. Forest, R. Dendievel & G.R. Canova. *Estimating the overall properties of heterogeneous Cosserat materials*. Modelling Simul. Mater. Sci. Eng. 7, pp. 829-840, 1999.

LITERATURVERZEICHNIS

- [Forest 00] S. Forest, F. Barbe & G. Cailletaud. *Cosserat modeling of size effects in the mechanical behaviour of polycrystals and multi-phase materials*. International Journal of Solids and Structures 37, pp. 7105-7126, 2000.
- [Forest 01] S. Forest, F. Pradel & K. Sab. *Asymptotic analysis of heterogeneous Cosserat media*. International Journal of Solids and Structures 38, pp. 4585-4608, 2001.
- [Green 64] A. E. Green & R. S. Rivlin. *Multipolar Continuum Mechanics*. Archive for Rational Mechanics and Analysis, 17, pp. 113-147, 1964.
- [Kanit 03] T. Kanit. *Determination of the size of the representative volume element for random composites: statistical and numerical approach*. International Journal of Solids and Structures, vol. 40, pp. 3647-3679, 2003.
- [Kaßbohm 05a] S. Kaßbohm, W. H. Müller & R. Feßler. *Fourier series for computing the response of periodic structures with arbitrary stiffness distribution*. Computational Materials Science 32, pp. 387-391, 2005.
- [Kaßbohm 05b] S. Kaßbohm, W. H. Müller & R. Feßler. *Improved approximations of Fourier coefficients for computing periodic structures with arbitrary stiffness distribution*. Computational Materials Science, accepted for publication 2005.
- [Lakes 81] R.S. Lakes & J. F. C. Yang. *Transient study of couple stress in compact bone: torsion*. Journal of Biomechanical Engineering, 103, pp. 275-279, 1981.
- [Lakes 82] R.S. Lakes & J. F. C. Yang. *Experimental study of micropolar and couple stress elasticity in bone in bending*. Journal of Biomechanics, 15, pp. 91-98, 1982.
- [Lakes 95] R. S. Lakes. *Experimental methods for study of Cosserat elastic solids and other generalized continua*. in Continuum models for materials with micro-structure, ed. H.Mühlhaus, J. Wiley, N. Y. Ch. 1, pp. 1-22, 1995.

LITERATURVERZEICHNIS

- [Mindlin 62] R. D. Mindlin & H.F. Tiersten. *Effects of Couple-stresses in Linear Elasticity*. Archive for Rational Mechanics and Analysis, 11, pp. 415-448, 1962.
- [Mindlin 64] R. D. Mindlin. *Micro-structure in Linear Elasticity*. Archive for Rational Mechanics and Analysis, 16, pp. 51-77, 1964.
- [Mindlin 65] R. D. Mindlin. *Second Gradient of Strain and Surface-Tension in Linear Elasticity*. International Journal of Solids and Structures, 1, 417-438, 1965.
- [Moulinec 94] H. Moulinec & P. Suquet. *A fast numerical method for computing the linear and nonlinear mechanical properties of composites*. Mécanique des solides, t. 318, Série II, pp. 1417-1423, 1994.
- [Moulinec 98] H. Moulinec & P. Suquet. *A numerical method for computing the overall response of nonlinear composites with complex microstructure*. Comput. Methods Appl. Mech. Engrg., 157, pp. 69-94, 1998.
- [Müller 96] W. H. Müller. *Mathematical versus experimental stress analysis of inhomogeneities in solids*. J. Phys., IV 6, pp. C1-139-C1-148, 1996.
- [Müller 04] W. H. Müller. *Morphology changes in solder joints - experimental evidence and physical understanding*. Microelectronics Reliability 44, pp. 1901-1914, 2004.
- [Muschik 01] W. Muschik, C. Papenfuss & H. Ehrentraut. *A sketch of continuum thermodynamics*. J. Non-Newtonian Fluid Mech., 96, pp. 255-290, 2001.
- [Muschik 04] W. Muschik, C. Papenfuss & H. Ehrentraut. *Sketch of the mesoscopic description of nematic liquid crystals*. J. Non-Newtonian Fluid Mech., 119, pp. 91-104, 2004.
- [Nakamura 84] S. Nakamura, R. Benedict & R. Lakes. *Finite Element method for orthotropic micropolar elasticity*. Int. J. Engng. Sci., Vol. 22, No. 3, pp. 319-330, 1984.

LITERATURVERZEICHNIS

- [Nakamura 95] S. Nakamura & R.S. Lakes. *Finite element analysis of Saint Venant end effects in micropolar elastic solids*. Engineering Computations, 12, pp. 571-587, 1995.
- [Oden 70] J. T. Oden, D. M. Rigsby & D. Cornett. *On the numerical solution of a class of problems in a linear first strain-gradient theory of elasticity*. Int. J. for Numerical Meth. in Engng., Vol. 2, pp. 159-174, 1970.
- [Padovan 78] J. Padovan. *Applications of 3-D Finite Element procedures to static and dynamic problems in micropolar elasticity*. Computers & Structures, Vol. 8, pp. 231-236, 1978.
- [Press 92] Press, Teukolsky, Vetterling & Flannery. *Numerical recipes in fortran*. Cambridge University Press, Second Edition, 1992.
- [Spivak 70] M. Spivak. *Differential geometry*, vol. I. Publish or Perish, Inc., 6 Beacon Street, Boston, Mass. 02108 (U.S.A.), 1970.
- [Svendsen 99] B. Svendsen. *On the thermodynamics of thermoelastic materials with additional scalar degrees of freedom*. Continuum Mech. Therm., 4, pp. 247-262, 1999.
- [Svendsen 01] B. Svendsen. *On the continuum modeling of materials with kinematic structure*. Acta Mech., 152, pp. 49-80., 2001.
- [Toupin 64] R. A. Toupin. *Theories of Elasticity with Couple-Stress*. Archive for Rational Mechanics and Analysis, 17, pp. 85-112, 1964.
- [Trostel 80] R. Trostel. *Eine Struktur-Analyse der Materialgleichungen in der Stabtheorie auf der Basis der Cosserat-Modellvorstellung*. Beiträge zur Bautechnik, herausgegeben von Jürgen Bauer, Claus Scheer, Erich Czesielski, Verlag Wilhelm Ernst und Sohn, 1980.

Index

- GL(3), allgemeine lineare Gruppe, 9
SO(3), spezielle orthogonale Gruppe, 9
 S^1 , 1-Sphäre, 9
 S^2 , 2-Sphäre, 9
- Abminderungsfaktor, 31, 32, 36, 83
Abminderungsfaktor, Berechnung hiervon, 49
Abminderungsfaktoren, 14
axialer Vektor, 17
- Beispiel, akademisches, 34
Beispiel, praktisches, 38
Beispiele, 34
Beispiele, akademische, 34
- Cosserat-Kontinuum, 9, 13, 15, 17
Couple-Stress-Kontinuum, 13
- Dehnungen, thermische, 42
Determinismus, 10
DFT, diskrete Fourier Transformation, 35
Differentialgleichungen, 17
Differentialgleichungen, lineare partielle, 17
Differentialoperator, 17
Diskretisierung, 32, 34
Diskretisierungsgitter, 35
- Eigendehnung, 22, 35
Eigendehnung, thermische Dehnung, 21
Eigendehnungen, 42
- Eigenspannung, 21
Einheit, imaginäre, 28
Elastizitätsmodul, 42
Elastizitätstheorie, lineare, 14
Euklidischer Punktraum, 9
Experimente, numerische, 34
- Faltung, 14
Faltung, periodische, 31
Faser-Matrix Struktur, 20
Fehler, relativer, 79
Fehlervektor, 79
Fejérsche Mittel, 31
Flüssigkristall, 9
Fourier Transformation, diskrete, 30
Fourier Transformation, diskrete inverse, 30
Fourier Transformation, inverse diskrete, 86
Fourierkoeffizient, 25, 28, 35
Fourierreihe, 14, 25, 28, 86
Funktion, periodische, 27
- Gebiet, 20
Gitter, 34
Gleichgewichtsbedingungen, 19
granularer Stoff, 9
- Homogenisierung, 13
- imaginäre Einheit, 28
Indizes, 5
Interpolationspolynom, 35
Interpolationspolynom, trigonometrisches, 30, 35, 86

- Invarianz, 10
- Iteration, 26

- Körper, 9, 10
- klassisches Kontinuum, 10, 13
- Komponenten des Spannungstensors, 38
- Kompressionsmodul, 42
- Konfiguration, 9
- Konfigurationsraum, 9
- Kontinuum mit Mikrostruktur, 9
- Kontinuum, klassisches, 15
- Kreiswellenvektor, 2

- Längen im RVE, 28
- Lösung, explizite, 22
- Lösung, implizite, 22
- Lösungsoperator, 22
- Lage, 10
- Laplace-Operator, 19
- Lasten, 17, 21, 34
- Liegruppe, 9

- Mannigfaltigkeit, 9
- Materialgesetz, 19
- Materialkonstanten, 45
- Mittelwerte, 21
- Momentenspannung, 15
- Momentenspannung, Paarspannung, 13

- Näherung, 32
- Näherung, m -te, 25
- Neumann-Iteration, 22, 62, 79
- noweb, 60
- Nyquist Frequenz, 30

- Objektivität, 10
- Operator, 17

- Ordnung, 9, 10
- Ordnungsparameter, 10
- Originalproblem, 22
- Orthogonalitätsrelationen, 28
- Orthonormalitätsrelationen, 28
- Ortsvektor, 10, 24
- Ortsvektor, dimensionsloser diskreter, 30
- Ortsvektor, dimensionsloser kontinuierlicher, 31

- Paar, geordnetes, 17
- Paarspannung, 13, 15
- poröser Stoff, 9
- Problem, vereinfachtes, 22
- Programm-Dokumentation, 60

- Randbedingungen, 17, 34
- Randbedingungen, periodische, 13, 17, 20
- Randwertproblem, 17
- Rekursion, 22
- rekursive Formel, 32
- RVE, 34

- Schubmodul, 42
- Spannung, 13
- Spin, 9
- Spur (eines Tensors), 60
- Störung, verallgemeinerte, 24
- Startwert, 26
- Statik, 17
- Struktogramm, 62

- Teilchen, 9
- Temperatur, 42
- Theorie, makroskopische, 10
- Theorie, mikroskopische, 10
- transponierter Tensor, 60

INDEX

trigonometrisches Interpolations-
polynom, 36

Tripel, geordnetes, 2

Vergrößerung, 13

Verschiebung, 17

Verschiebung, verallgemeinerte,
17

Verschiebungs-Verzerrungs Zu-
sammenhang, 20

Verschiebungsvektor, 17

Verzerrung, 13

Verzerrung, verallgemeinerte, 19

Volumenelement, repräsentatives,
15

Volumenelement, repräsentatives,
RVE, 14

Wellenvektor, 28

Wellenvektor, dimensionsloser, 28

Zahlenfolge, 30