



## Diplomarbeit

# Implementierung der Hauptnormalspannungen und -verzerrungen in das Finite Elemente Programm CFS++

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines

Diplom-Ingenieurs (Dipl.-Ing.)

eingereicht an der TU Wien, Fakultät für Maschinenwesen und Betriebswissenschaften

von

**Michael SEIFRIED**

Matrikelnummer: 01026247

unter der Leitung von

Univ. Prof. Dipl.-Ing. Dr. Manfred Kaltenbacher

Mitwirkung

Dipl.-Ing. Dr. Florian Toth

Institut für Mechanik und Mechatronik, E325

begutachtet von

Univ. Prof. Dipl.-Ing. Dr.

Manfred Kaltenbacher

Institut für Mechanik und Mecha-  
tronik

Technische Universität Wien

Univ. Prof. Dipl.-Ing. Dr.

Helmut Böhm

Institut für Leichtbau und  
Struktur-Biomechanik

Technische Universität Wien

Ich nehme zur Kenntnis, dass ich zur Drucklegung meiner Arbeit unter der Bezeichnung

**Diplomarbeit**

nur mit Bewilligung der Prüfungskommission berechtigt bin.

*Eidesstattliche Erklärung*

Ich erkläre an Eides statt, dass die vorliegende Arbeit nach den anerkannten Grundsätzen für wissenschaftliche Abhandlungen von mir selbstständig erstellt wurde. Alle verwendeten Hilfsmittel, insbesondere die zugrunde gelegte Literatur, sind in dieser Arbeit genannt und aufgelistet. Die aus den Quellen wörtlich entnommenen Stellen sind als solche kenntlich gemacht.

Das Thema dieser Arbeit wurde von mir bisher weder im In- noch im Ausland einer Beurteilerin/ einem Beurteiler zur Begutachtung in irgendeiner Form als Prüfungsarbeit vorgelegt. Diese Arbeit stimmt mit der von den Begutachtern beurteilten Arbeit überein.

Wien, Juli 2017

.....

Unterschrift

# Inhaltsverzeichnis

<b>Einleitung</b>	<b>1</b>
<b>1 Theorie</b>	<b>2</b>
1.1 Der Spannungsbegriff . . . . .	2
1.1.1 Symmetrie des Spannungstensors . . . . .	4
1.1.2 Andere Spannungsgrößen . . . . .	4
1.2 Der Verzerrungsbegriff . . . . .	5
1.2.1 Geometrische Bedeutung der Verzerrungen . . . . .	6
1.2.2 Andere Verzerrungsgrößen . . . . .	8
1.2.3 Linearisierter Verzerrungstensor . . . . .	8
1.3 Hooke'sches Gesetz . . . . .	9
1.4 Verträglichkeitsbedingungen . . . . .	9
1.5 Hauptnormalspannungen . . . . .	10
1.5.1 Lagrange-Maximierungsproblem . . . . .	10
1.5.2 Eigenschaften . . . . .	12
1.6 Hauptnormalverzerrungen . . . . .	13
1.7 Zweck der Hauptnormalspannungen . . . . .	14
1.7.1 Versagenshypothesen . . . . .	14
1.8 Lösung des Eigenwertproblems . . . . .	16
1.8.1 Begriffe aus der linearen Algebra . . . . .	17
1.8.2 Ähnlichkeitstransformationen . . . . .	17
1.8.3 QR-Zerlegung . . . . .	17
1.8.4 Transformation auf Hessenbergform . . . . .	18
1.8.5 Algorithmen zur Lösung des Eigenwertproblems . . . . .	19
<b>2 Implementierung</b>	<b>21</b>
2.1 Numerische Methoden in der Entwicklung . . . . .	21
2.2 Aufbau von CFS++ . . . . .	21
2.3 Implementierung in der MechPDE . . . . .	22
2.4 CoefFunctionEigen . . . . .	23
2.4.1 Auswahl des LAPACK-Drivers . . . . .	23
2.5 CoefFunctionCache . . . . .	24
2.5.1 Datentyp des Cache . . . . .	25
2.5.2 Benchmarks . . . . .	26

<b>3 Testfälle</b>	<b>29</b>
3.1 Einachsiger Schub . . . . .	29
3.1.1 Analytische Lösung . . . . .	29
3.1.2 Lösung in CFS++ . . . . .	30
3.2 Würfel unter Zugspannung . . . . .	31
3.2.1 Analytische Lösung . . . . .	31
3.2.2 Lösung in CFS++ . . . . .	32
3.3 Rohr unter Innendruck . . . . .	32
3.3.1 Airy'sche Spannungsfunktion . . . . .	33
3.3.2 Analytische Lösung . . . . .	34
3.3.3 Lösung in CFS++ . . . . .	35
3.4 Lochscheibe . . . . .	37
3.4.1 Analytische Lösung . . . . .	37
3.4.2 Lösung in CFS++ . . . . .	38
3.5 MEMS-Lautsprecher . . . . .	41
3.5.1 MEMS-Technologie . . . . .	41
3.5.2 Piezoelektrischer Effekt . . . . .	41
3.5.3 Modellierung des Lautsprechers . . . . .	42
3.5.4 Vergleich bestehender MEMS Lautsprecher . . . . .	44
3.5.5 Ergebnisse . . . . .	44
<b>Conclusio</b>	<b>50</b>
<b>Abbildungsverzeichnis</b>	<b>51</b>
<b>Tabellenverzeichnis</b>	<b>52</b>
<b>Literaturverzeichnis</b>	<b>53</b>

## Verwendete Formelzeichen

$A$	$\text{m}^2$	Fläche ( <i>area</i> )
$A_{ij}$	1	Almansi'sche Verzerrungen ( <i>Almansi strain</i> )
$\mathbf{b}$	$\frac{\text{m}}{\text{s}^2}$	Beschleunigung ( <i>acceleration</i> )
$[\mathbf{c}^E]$	$\frac{\text{N}}{\text{m}^2}$	Elastizitätstensor ( <i>elasticity tensor</i> )
$E$	$\frac{\text{N}}{\text{m}^2}$	Elastizitätsmodul ( <i>elasticity modulus</i> )
$\mathbf{E}$	$\frac{\text{V}}{\text{m}}$	Elektrische Feldstärke ( <i>electric field intensity</i> )
$\mathbf{e}$	1	Einheitsvektor ( <i>unit vector</i> )
$[\mathbf{e}]$	1	piezoelektrischer Koppeltensor ( <i>piezoelectric coupling tensor</i> )
$\mathbf{F}$	N	mechanische Kraft ( <i>force</i> )
$[\mathbf{F}]$	1	Householder-Reflektor ( <i>Householder reflector</i> )
$[\mathbf{F}_d]$	1	Deformationsgradiententensor ( <i>deformation gradient</i> )
$\mathbf{f}$	$\frac{\text{N}}{\text{m}^3}$	mechanische Kraftdichte ( <i>force density</i> )
$G$	$\frac{\text{N}}{\text{m}^2}$	Schubmodul ( <i>shear modulus</i> )
$[\mathbf{I}]$	1	Einheitsmatrix ( <i>unit matrix</i> )
$ \mathcal{J} $	1	Determinante von $[\mathbf{F}_d]$ ( <i>determinant of <math>[\mathbf{F}_d]</math></i> )
$\mathbf{k}$	$\frac{\text{N}}{\text{m}^3}$	mechanische Volumskraftdichte ( <i>mechanic body force density</i> )
$\mathbf{n}$	1	Normaleneinheitsvektor ( <i>unit normal vector</i> )
$[\mathbf{Q}]$	1	Drehmatrix ( <i>rotary matrix</i> )
$\mathbf{R}$	N	mechanische Volumskraft ( <i>mechanic body force</i> )
$\mathbf{r}$	N	Ortsvektor ( <i>position vector</i> )
$[\mathbf{T}]$	$\frac{\text{N}}{\text{m}^2}$	2. Piola-Kirchhoff Spannungstensor ( <i>2nd Piola-Kirchhoff tensor</i> )
$\mathbf{t}$	$\frac{\text{N}}{\text{m}^2}$	mechanischer Spannungsvektor ( <i>traction vector</i> )
$U^*$	$\frac{\text{N}}{\text{m}^2}$	Verzerrungsenergie pro Volumseinheit ( <i>strain energy density</i> )
$U_V^*$	$\frac{\text{N}}{\text{m}^2}$	Volumsverzerrungsenergie pro Volumseinheit ( <i>volume strain energy density</i> )
$U_g^*$	$\frac{\text{N}}{\text{m}^2}$	Gestaltänderungsenergie pro Volumseinheit ( <i>deformation energy density</i> )
$u, v, w$	m	Verschiebungen ( <i>displacement</i> )
$V$	$\text{m}^3$	Volumen ( <i>volume</i> )
$\epsilon_i$	1	Dehnungen ( <i>stretch</i> )
$\epsilon_{ij}$	1	Green'sche Verzerrungen ( <i>Green's strain</i> )
$\epsilon_1, \epsilon_2, \epsilon_3$	1	Hauptnormalverzerrungen ( <i>principal strain</i> )
$\epsilon_{ij}$	1	Linearisierte Verzerrungen ( <i>linearized strain</i> )
$[\boldsymbol{\epsilon}]$	1	Linearisierter Verzerrungstensor ( <i>linearized strain tensor</i> )
$\nu$	1	Poisson-Zahl, Querkontraktionszahl ( <i>Poisson ratio</i> )
$\rho$	$\frac{\text{kg}}{\text{m}^3}$	Dichte ( <i>density</i> )
$[\boldsymbol{\sigma}]$	$\frac{\text{N}}{\text{m}^2}$	Cauchy Spannungstensor ( <i>Cauchy stress tensor</i> )
$\boldsymbol{\sigma}$	$\frac{\text{N}}{\text{m}^2}$	Spannungsvektor ( <i>stress vector</i> )
$\sigma$	$\frac{\text{N}}{\text{m}^2}$	Normalspannung ( <i>normal stress</i> )
$\sigma_F$	$\frac{\text{N}}{\text{m}^2}$	Streckgrenze ( <i>yield strength</i> )
$\sigma_V$	$\frac{\text{N}}{\text{m}^2}$	Vergleichsspannung ( <i>equivalent stress</i> )
$\sigma_1, \sigma_2, \sigma_3$	$\frac{\text{N}}{\text{m}^2}$	Hauptnormalspannungen ( <i>principal stress</i> )
$[\boldsymbol{\tau}]$	$\frac{\text{N}}{\text{m}^2}$	1. Piola-Kirchhoff Spannungstensor ( <i>1st Piola-Kirchhoff tensor</i> )
$\boldsymbol{\tau}$	$\frac{\text{N}}{\text{m}^2}$	Schubspannung ( <i>shear stress</i> )
$\Phi$	1	Airy'sche Spannungsfunktion ( <i>Airy stress function</i> )

$\frac{\partial x}{\partial t}$	partielle Differentiation von $x$ nach $t$
$dx$	totales Differential von $x$
$\Delta$	Laplace-Operator
$\nabla$	Nabla-Operator
$\bar{x}$	konjugiert Komplexe des Skalars $x$
$ x $	Betrag von $x$
$x$	Vektor $x$
$x^T$	transponierter Vektor $x$
$\bar{x}$	konjugiert Komplexe von $x$
$\ x\ _2$	$L_2$ - Norm von $x$
$[\mathbf{K}]$	Matrix oder Tensor $[\mathbf{K}]$
$[\mathbf{K}]^T$	transponierte Matrix $[\mathbf{K}]$
$[\mathbf{K}]^{-1}$	inverse Matrix von $[\mathbf{K}]$
$\det[\mathbf{K}]$	Determinante von $[\mathbf{K}]$
$\text{Tr}[\mathbf{K}]$	Spur von $[\mathbf{K}]$
$\overline{[\mathbf{K}]}$	konjugiert Komplexe von $[\mathbf{K}]$

# Einleitung

In dieser Diplomarbeit wird die Implementierung der Hauptnormalspannungen und -verzerrungen in das Finite Elemente Programm Coupled Field Simulation (CFS++) beschrieben. Aufbauend auf einer kurzen Wiederholung des Spannungs- und Verzerrungsbegriffes wird im Kapitel „Theorie“ die Herleitung der Hauptnormalspannungen und -verzerrungen rekapituliert. Um eine Lösung des Problems in einem Computerprogramm zu ermöglichen, wird die numerische Lösung des Eigenwertproblems kurz diskutiert.

Im Kapitel „Implementierung“ werden die notwendigen Programmteile, die zur Berechnung der Hauptnormalspannungen- und verzerrungen notwendig sind, erläutert. Diese wurden im Ergebnis-Postprocessing in CFS++ implementiert.

Die Implementierung wurde anhand mehrerer Beispiele ausgiebig getestet. Beginnend am einzelnen Element und weiterführend mit klassischen Problemen der Festigkeitslehre wurde die Plausibilität der Ergebnisse überprüft. Dabei wurden, soweit möglich, analytische Lösungen zur Spannungsberechnung herangezogen und diese mit den berechneten Hauptnormalspannungen verglichen. Abschließend wurde ein MEMS-Lautsprecher modelliert und dessen Hauptnormalspannungsverteilung diskutiert.

# Kapitel 1

## Theorie

### 1.1 Der Spannungsbegriff

Für eine kurze Einführung in den Spannungsbegriff wurden Teile aus [1] übernommen. Der Spannungsbegriff ist ein grundlegender Teil der Dynamik, welcher die Zusammenhänge zwischen Kraft und Bewegung eines Körpers beschreibt. Betrachtet man die inneren Kräfte in einem Körper in einem kartesischen Koordinatensystem, kann man ein kleines Volumenelement aus dem Körper mit den dazugehörigen Kräften ausschneiden. Diese Kräfte sind in Abb. 1.1 gezeigt, wobei  $\sigma_i$  die angreifenden Spannungen an den Oberflächen des Volumenelements darstellen und  $d\mathbf{R}$  die angreifenden Volumskräfte. Bezeichnet man  $\mathbf{k} = \frac{d\mathbf{R}}{dV}$  als die örtliche Volumskraft pro Volumeneinheit,  $\rho$  als die Dichte im betrachteten Körperpunkt und  $\mathbf{b}$  als die Beschleunigung des Körperpunktes in einem Inertialsystem, gilt das Gleichungssystem

$$\begin{aligned}\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{yx}}{\partial y} + \frac{\partial \sigma_{zx}}{\partial z} + k_x &= \rho b_x \\ \frac{\partial \sigma_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \sigma_{zy}}{\partial z} + k_y &= \rho b_y \\ \frac{\partial \sigma_{xz}}{\partial x} + \frac{\partial \sigma_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z} + k_z &= \rho b_z,\end{aligned}\tag{1.1}$$

oder kurz

$$\nabla[\boldsymbol{\sigma}] + \mathbf{k} = \rho \mathbf{b}\tag{1.2}$$

mit

$$[\boldsymbol{\sigma}] = \begin{bmatrix} \sigma_{xx} & \sigma_{yx} & \sigma_{zx} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{zy} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix},\tag{1.3}$$

wobei  $[\boldsymbol{\sigma}]$  den Spannungstensor darstellt. Die Gln. (1.1) und (1.2) werden auch als *dynamisches Grundgesetz* bezeichnet. Die Spannungsvektoren aus Abb. 1.1 lassen sich in ihre Komponenten zerlegen, sodass

$$\boldsymbol{\sigma}_i = \sigma_{ix}\mathbf{e}_x + \sigma_{iy}\mathbf{e}_y + \sigma_{iz}\mathbf{e}_z \quad i = x, y, z\tag{1.4}$$

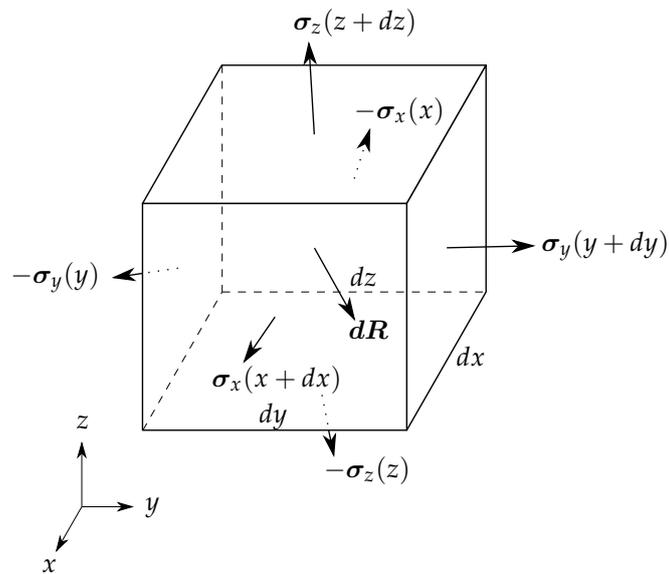


Abbildung 1.1: Volumselement mit resultierenden Spannungsvektoren, nach [1]

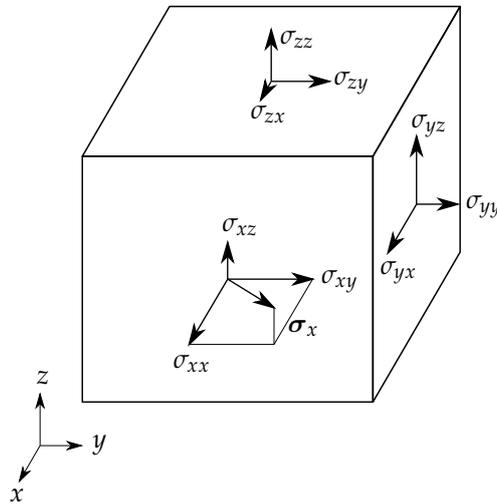


Abbildung 1.2: Komponenten des Spannungstensors am Volumselement, nach [1]

gilt. Die Komponenten werden in dieser Notation mit Doppelindex geschrieben, wobei der erste Index der Spannungskomponente die Normalenrichtung der Schnittfläche beschreibt, während der zweite Index die Richtungskomponente des Spannungsvektors auf die Schnittfläche beschreibt. Siehe auch Abb. 1.2. Betrachtet man beliebige Schnittebenen, kann der resultierende Spannungsvektor auf die Schnittebene aus dem Normalenvektor  $\mathbf{n}$  auf die Schnittebene und dem Spannungstensor gebildet werden. Der Zusammenhang

$$[\boldsymbol{\sigma}] \mathbf{n} = \mathbf{t} \quad (1.5)$$

unter der Nebenbedingung

$$\|\mathbf{n}\|_2 = \sqrt{n_x^2 + n_y^2 + n_z^2} = 1 \quad (1.6)$$

wird auch als *Cauchy'sches Fundamentaltheorem* bezeichnet. Abbildung 1.3 soll den Zusammenhang verdeutlichen.

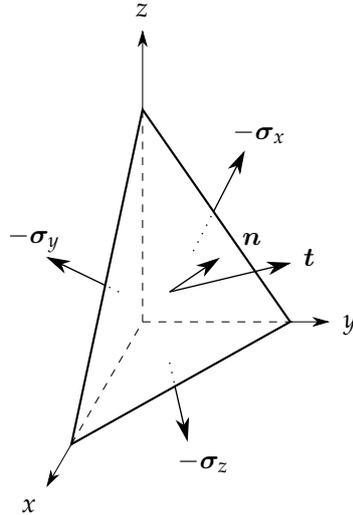


Abbildung 1.3: Spannungstensor in einer beliebigen Schnittebene, nach [1]

### 1.1.1 Symmetrie des Spannungstensors

Wendet man auf Gl. (1.2) einige Umformungen an, führen diese zur Gleichung

$$\int_m \mathbf{r} \times \mathbf{b} dm = \mathbf{M} + \int_V [(\sigma_{zy} - \sigma_{yz}) \mathbf{e}_x + (\sigma_{xz} - \sigma_{zx}) \mathbf{e}_y + (\sigma_{yx} - \sigma_{xy}) \mathbf{e}_z] dV, \quad (1.7)$$

wobei  $\mathbf{r}$  einen beliebigen Ortsvektor und  $\mathbf{M} = \int_V \mathbf{r} \times \mathbf{k} dV$  das resultierende Moment der äußeren Kräfte darstellen [1]. Für einen in Ruhe befindlichen Körper, auf den ein Gleichgewichtssystem von äußeren Kräften einwirkt, muss

$$\sigma_{zy} = \sigma_{yz}, \quad \sigma_{xz} = \sigma_{zx}, \quad \sigma_{xy} = \sigma_{yx} \quad (1.8)$$

gelten [1]. Der Spannungstensor ist somit ein symmetrischer Tensor. Das Boltzmann-Axiom verallgemeinert diesen Satz auf beliebig bewegte Körper [1].

Es muss erwähnt werden, dass die Symmetrie des Spannungstensors nur für klassische Materialien der Kontinuumsmechanik gilt. Lässt man Kontinua mit unabhängigen Volumenmomenten und Flächenmomenten zu, kann die Symmetrie nicht nachgewiesen werden. Nähere Erläuterungen dazu finden sich in [2].

### 1.1.2 Andere Spannungsgrößen

An dieser Stelle sei vor allem der zweite Piola'sche Spannungstensor erwähnt, der in der geometrisch nichtlinearen Analyse eine große Rolle spielt. Da in Gl. (1.2) Größen, welche im deformierten Zustand definiert werden (etwa der Cauchy'sche Spannungstensor  $[\boldsymbol{\sigma}]$ ) mit Größen vermischt werden, welche im undeformierten Zustand definiert werden, etwa der Kraftdichte  $\mathbf{k}$ , darf diese nur in der linearen Mechanik angewandt werden, da hier die Verschiebungen gegenüber den Körperabmessungen als klein angenommen werden, was bei einer nichtlinearen Analyse nicht mehr der Fall ist [3].

Für die Herleitung des zweiten Piola'schen Spannungstensors muss auch der erste Piola'sche Spannungstensor  $[\boldsymbol{\tau}]$  hergeleitet werden. Es gilt mit  $[\mathbf{F}_d]$  als dem Deformationsgradiententensor, welcher in

Gl. (1.15) eingeführt wird, und  $|\mathcal{J}|$  als dessen Determinante

$$\int_A [\boldsymbol{\sigma}] dA = \int_{A_0} |\mathcal{J}| [\boldsymbol{\sigma}] [\mathbf{F}_d]^{-T} dA_0 = \int_{A_0} [\boldsymbol{\tau}] dA_0, \quad (1.9)$$

woraus der Zusammenhang

$$[\boldsymbol{\tau}] = |\mathcal{J}| [\boldsymbol{\sigma}] [\mathbf{F}_d]^{-T} \quad (1.10)$$

leicht abgelesen werden kann [3]. Der erste Piola'sche Spannungstensor verknüpft die Kraft in der deformierten Konfiguration mit dem Flächenelement in der undeformierten Konfiguration und wird somit auch als Nennspannungstensor bezeichnet [2].

$[\boldsymbol{\tau}]$  ist ein nichtsymmetrischer Nennspannungstensor und somit für die Verknüpfung von Spannungs- und Verzerrungstensoren in Konstitutivgleichungen ungeeignet. Daher wird  $[\boldsymbol{\tau}]$  so modifiziert, dass dieser wieder symmetrisch wird [2]. Man führt den zweiten Piola'schen Spannungstensor  $[\mathbf{T}]$  mittels

$$[\mathbf{T}] = [\mathbf{F}_d]^{-1} [\boldsymbol{\tau}] \quad (1.11)$$

ein [2]. Die Symmetrie von  $[\mathbf{T}]$  lässt sich über die Symmetrie des Cauchy-Spannungstensors  $[\boldsymbol{\sigma}]$  leicht zeigen, denn es gilt

$$\frac{[\mathbf{T}]^T}{|\mathcal{J}|} = \left( [\mathbf{F}_d]^{-1} [\boldsymbol{\sigma}] [\mathbf{F}_d]^{-T} \right)^T = \left( [\boldsymbol{\sigma}] [\mathbf{F}_d]^{-T} \right)^T \left( [\mathbf{F}_d]^{-1} \right)^T = [\mathbf{F}_d]^{-1} [\boldsymbol{\sigma}]^T [\mathbf{F}_d]^{-T} = \frac{[\mathbf{T}]}{|\mathcal{J}|}. \quad (1.12)$$

Somit kann das dynamische Grundgesetz Gl. (1.2) in der Form

$$\nabla_X ([\mathbf{F}_d] [\mathbf{T}]) + \mathbf{k} = \rho \mathbf{b} \quad (1.13)$$

angegeben werden, wobei nur mehr Größen vorkommen, die im Ausgangszustand definiert werden. In dieser Form ist es somit für weiterführende Herleitungen, etwa im Rahmen der nichtlinearen Finiten Elemente Analyse, zugänglich [3].

## 1.2 Der Verzerrungsbegriff

Die Herleitung des Verzerrungsbegriffes und dessen geometrische Bedeutung ist an die Darstellung in [1] angelehnt. Der Verzerrungszustand ist im Gegensatz zum Spannungszustand eine Größe der Kinematik. Dieses Teilgebiet der Mechanik befasst sich mit der Beschreibung der Bewegung und Verformung eines Körpers, ohne auf die dabei wirkenden Kräfte einzugehen. Für die Beschreibung der Verzerrungen betrachtet man zwei beliebige Lagen eines Körpers zum Zeitpunkt  $t_0 = 0$  und zum Zeitpunkt  $t$ , wobei eine Verformung des Körpers zulässig ist. Die erstgenannte Lage des Körpers wird auch Referenzkonfiguration genannt, letztere Momentankonfiguration.

Betrachtet man zur Referenzkonfiguration ein Bogenelement mit der Länge  $dl_0$  im Körper, wird dieses aufgrund der Verformungen die Länge  $dl$  zum Zeitpunkt  $t$  haben, siehe auch Abb. 1.4. Außerdem wird ein beliebiger Körperpunkt  $P$  betrachtet und in kartesischen Koordinaten beschrieben. Die Koordinaten des Punktes in der Ausgangskonfiguration werden mit Großbuchstaben  $X, Y, Z$  bezeichnet, in der Momentankonfiguration mit den Kleinbuchstaben  $x, y, z$ , und die Verschiebung mit den Buchstaben  $u, v, w$ . Dann gilt

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} u \\ v \\ w \end{bmatrix}. \quad (1.14)$$

Der Deformationsgradiententensor  $[\mathbf{F}_d]$  ist mittels

$$[\mathbf{F}_d] = \begin{bmatrix} \frac{\partial x}{\partial X} & \frac{\partial x}{\partial Y} & \frac{\partial x}{\partial Z} \\ \frac{\partial y}{\partial X} & \frac{\partial y}{\partial Y} & \frac{\partial y}{\partial Z} \\ \frac{\partial z}{\partial X} & \frac{\partial z}{\partial Y} & \frac{\partial z}{\partial Z} \end{bmatrix} \quad (1.15)$$

definiert [3]. Für die Länge eines Bogenelementes im unverformten und im verformten Zustand kann

$$dl_0^2 = dX^2 + dY^2 + dZ^2, \quad dl^2 = dx^2 + dy^2 + dz^2 \quad (1.16)$$

geschrieben werden.

Der Verschiebungsvektor wird nun im Sinne einer Lagrange-Formulierung als Funktion der Koordinaten  $X, Y, Z$  und der Zeit  $t$  angegeben. Für die Darstellung in Gl. (1.16) muss nun von den Verschiebungskomponenten das totale Differential gebildet werden, was für die Komponente  $u$

$$dx = dX + du = dX + \frac{\partial u}{\partial X} dX + \frac{\partial u}{\partial Y} dY + \frac{\partial u}{\partial Z} dZ \quad (1.17)$$

ergibt. Es bietet sich an, in diesem Fall die Einstein'sche Summenkonvention zu verwenden, da mit ihr Gleichungen in kompakter Form angegeben werden können. Über doppelt vorkommende Indizes innerhalb eines Produktes oder einer Variable ist zu summieren. Gleichung (1.17) lässt sich somit allgemeiner in der Form

$$dx_i = dX_i + \frac{\partial u_i}{\partial X_j} dX_j \quad (1.18)$$

anschreiben.

Die Metrik eines Raumes ist durch die Angabe des Quadrates des Bogenelementes vollständig bestimmt, eine Winkel - oder Längenmessung ist daher mit einer  $L_2$ -Norm gegeben. Daher wird die Differenz aus  $dl$  und  $dl_0$  gebildet, was

$$\frac{1}{2}(dl^2 - dl_0^2) = \epsilon_{xx} dX^2 + \epsilon_{yy} dY^2 + \epsilon_{zz} dZ^2 + 2(\epsilon_{xy} dX dY + \epsilon_{yz} dY dZ + \epsilon_{xz} dX dZ) \quad (1.19)$$

ergibt, wobei die eingeführten Koeffizienten  $\epsilon_{ij}$  in der Form

$$\epsilon_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} + \frac{\partial u_k}{\partial X_i} \frac{\partial u_k}{\partial X_j} \right) \quad (1.20)$$

angegeben werden können. Aus Gl. (1.20) ist ersichtlich, dass eine Vertauschung der Indizes  $i$  und  $j$  keine Änderung des Ergebnisses bewirkt. Folglich ist der Green'sche Verzerrungstensor, der aus den Koeffizienten nach Gl. (1.20) besteht, wie der Cauchy'sche Spannungstensor ein symmetrischer Tensor.

### 1.2.1 Geometrische Bedeutung der Verzerrungen

Es stellt sich nun die Frage, wie der Verzerrungstensor geometrisch zu interpretieren ist. Dazu wird das Linienelement  $dl_0 = dX, \quad dY = dZ = 0$ , also jenes Linienelement, das in der Ausgangskonfiguration parallel zur  $x$ -Achse liegt, betrachtet. Eingesetzt in Gl. (1.19) erhält man die Beziehung

$$dl = \sqrt{1 + 2\epsilon_{xx}} dX, \quad (1.21)$$

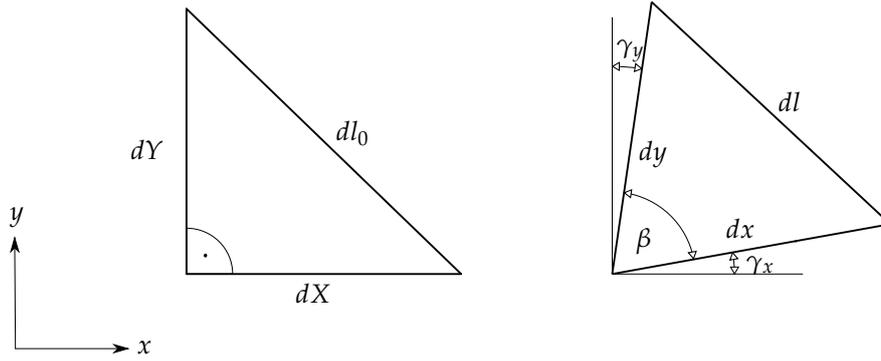


Abbildung 1.4: Geometrische Betrachtung der Verzerrungen, nach [1]

wobei dieses Element nach der Deformation nicht notwendigerweise parallel zur  $x$ -Achse liegen muss. Diese Beziehung lässt sich nun in die Definition der Dehnung einsetzen, welche

$$\epsilon = \frac{\Delta l}{l_0} = \frac{l - l_0}{l_0} = \frac{dl - dl_0}{dl_0} \quad (1.22)$$

lautet. Setzt man Gl. (1.21) in Gl. (1.22) ein und entwickelt in eine Potenzreihe, ergibt sich der Ausdruck

$$\epsilon_x = 1 + \epsilon_{xx} - \underbrace{\frac{1}{2} \epsilon_{xx}^2 + \dots}_{\text{nichtlinearer Anteil}} - 1. \quad (1.23)$$

oder in allgemeiner Form, wenn die Indizes vertauscht werden,

$$\epsilon_i = 1 + \epsilon_{ii} - \underbrace{\frac{1}{2} \epsilon_{ii}^2 + \dots}_{\text{nichtlinearer Anteil}} - 1. \quad (1.24)$$

Für kleine Dehnungen, also wenn  $|\epsilon_{ii}| \ll 1$  hält, gilt daraufhin:  $\epsilon_i = \epsilon_{ii}$ . Die Normalverzerrungen  $\epsilon_{xx}$ ,  $\epsilon_{yy}$  und  $\epsilon_{zz}$  entsprechen den Längenänderungen derjenigen Bogenelemente, die vor der Verformung parallel zu den Koordinatenachsen waren. Sind die Verzerrungen allerdings groß, gilt dieser Zusammenhang nicht mehr.

Es werden nun die Anteile des Verzerrungstensors interpretiert, die nicht in der Diagonale liegen. Dazu wird die geometrische Bedeutung des Elementes  $\epsilon_{xy}$  betrachtet. Abbildung 1.4 zeigt links die Linienelemente  $dX$  und  $dY$ , die in der Referenzkonfiguration mit den Koordinatenachsen  $x$  und  $y$  parallel sind, und rechts die verformten Linienelemente  $dx$  und  $dy$ , die mit den Koordinatenachsen nicht mehr parallel sind. Es gelten die geometrischen Beziehungen

$$dl_0^2 = dX^2 + dY^2 \quad (1.25)$$

und der Cosinussatz

$$dl^2 = dx^2 + dy^2 - 2 dx dy \cos(\beta). \quad (1.26)$$

Aufgrund geometrischer Überlegungen kann der Zusammenhang zwischen Schubwinkel  $\beta$  und der

Schubverzerrung  $\epsilon_{xy}$

$$-\cos(\beta) = -\cos\left(\frac{\pi}{2} - \gamma_x - \gamma_y\right) = \sin(\gamma_{xy}) = 2\epsilon_{xy} \underbrace{\frac{1}{(1 + \epsilon_x)(1 + \epsilon_y)}}_{\text{nichtlinearer Anteil}} \quad (1.27)$$

hergestellt werden. Es ist wiederum eine Linearisierung von Gl. (1.27) möglich. Für kleine Dehnungen  $\epsilon_x \ll 1$ ,  $\epsilon_y \ll 1$  ergibt sich der einfache Zusammenhang

$$\sin(\gamma_{xy}) = 2\epsilon_{xy}. \quad (1.28)$$

Die Verzerrungskomponenten  $\epsilon_{xy}$ ,  $\epsilon_{yz}$  und  $\epsilon_{zx}$  sind also mit den Winkeländerungen im Körper verknüpft, wobei die Verzerrungskomponente jeweils halb so groß ist wie die zugehörige Winkeländerung im Körper [1].

## 1.2.2 Andere Verzerrungsgrößen

In Kapitel 1.2 wurden die Verschiebungen und alle daraus hergeleiteten Größen im Sinne einer Lagrange'schen Betrachtung als Funktion der Referenzkonfiguration angegeben. Der daraus hergeleitete Verzerrungstensor wird als Green-Lagrange-Verzerrungstensors bezeichnet. Gibt man die Größen als Funktion der Momentankoordinaten an, tritt anstelle des Green-Lagrange-Verzerrungstensors der Almansi'sche Verzerrungstensor. Die Komponenten des Almansi'schen Verzerrungstensors lassen sich in der Form

$$A_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{\partial u_k}{\partial x_i} \frac{\partial u_k}{\partial x_j} \right) \quad (1.29)$$

anschreiben [1].

## 1.2.3 Linearisierter Verzerrungstensor

Wie bereits in den Gln. (1.23) und (1.27) gezeigt wurde, lässt sich für kleine Verzerrungen ein einfacher Zusammenhang zwischen den Dehnungen bzw. den Winkeländerungen und den Verzerrungen herstellen. Für kleine Verschiebungen des Körpers verglichen mit den Körperabmessungen lässt sich auch der Verzerrungstensor selbst linearisieren. Dies führt auf die Beziehung

$$2\epsilon_{ij} = \frac{\partial u_i}{\partial X_j} + \frac{\partial u_j}{\partial X_i} + \underbrace{\frac{\partial u_k}{\partial X_i} \frac{\partial u_k}{\partial X_j}}_{\approx 0}. \quad (1.30)$$

Linearisiert man den Almansi'schen Verzerrungstensor, erhält man dasselbe Ergebnis, es wird also nicht mehr zwischen Momentankonfiguration und Ausgangskonfiguration unterschieden. Der linearisierte Verzerrungstensor lautet ausgeschrieben nach Gl. (1.31) also

$$[\epsilon] = \begin{bmatrix} \epsilon_{xx} & \epsilon_{xy} & \epsilon_{xz} \\ \epsilon_{xy} & \epsilon_{yy} & \epsilon_{yz} \\ \epsilon_{xz} & \epsilon_{yz} & \epsilon_{zz} \end{bmatrix} = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & \frac{1}{2} \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \\ \frac{1}{2} \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) & \frac{\partial v}{\partial y} & \frac{1}{2} \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) \\ \frac{1}{2} \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) & \frac{1}{2} \left( \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right) & \frac{\partial w}{\partial z} \end{bmatrix}. \quad (1.31)$$

### 1.3 Hooke'sches Gesetz

Der Spannungs- und Verzerrungszustand in einem Körper sind voneinander abhängig. Ganz allgemein werden phänomenologische Variablen aus der Physik Konstitutivgrößen genannt, wozu etwa die Spannungen, Verzerrungen, Temperatur, Wärmestromvektor, ... zählen [2]. Konstitutivgleichungen verknüpfen gewisse Konstitutivgrößen miteinander und können von unterschiedlichster Form sein, etwa in Form von algebraischen Gleichungen, Differentialgleichungen, Integralgleichungen oder anderen Gleichungsformen.

Als Beispiel für eine Konstitutivgleichung wird hier das Hooke'sche Gesetz besprochen, welches einen linearen, algebraischen Zusammenhang zwischen dem Spannungstensor und dem Verzerrungstensor herstellt. Für den Spezialfall isotroper Materialien, d.h. eine vollständige Richtungsunabhängigkeit der physikalischen Eigenschaften des Materials, kann das Hooke'sche Gesetz ohne Berücksichtigung der Wärmedehnung in der Form

$$\varepsilon_{ij} = \frac{1}{2G} \left( \sigma_{ij} - \frac{\nu}{1+\nu} s \delta_{ij} \right), \quad s = \sigma_{kk} = \sigma_{xx} + \sigma_{yy} + \sigma_{zz} \quad (1.32)$$

$$\sigma_{ij} = 2G \left( \varepsilon_{ij} - \frac{\nu}{1-2\nu} e \delta_{ij} \right), \quad e = \varepsilon_{kk} = \varepsilon_{xx} + \varepsilon_{yy} + \varepsilon_{zz} \quad (1.33)$$

angegeben werden [4], wobei der Zusammenhang zwischen Schubmodul  $G$ , Elastizitätsmodul  $E$  und Querdehnungszahl (Poisson-Zahl)  $\nu$

$$G = \frac{E}{2(1+\nu)} \quad (1.34)$$

lautet. Spezialisiert man das Hooke'sche Gesetz für den ebenen Spannungszustand, das heißt  $\sigma_{zz} = \sigma_{zx} = \sigma_{zy} = 0$ , folgt aus Gl. (1.33)

$$\sigma_{ij} = 2G \left( \varepsilon_{ij} + \frac{\nu}{1-\nu} (\varepsilon_{xx} + \varepsilon_{yy}) \delta_{ij} \right). \quad (1.35)$$

Spezialisiert man das Hooke'sche Gesetz für den ebenen Verzerrungszustand, das heißt  $\varepsilon_{zz} = \varepsilon_{zx} = \varepsilon_{zy} = 0$ , folgt aus Gl. (1.32)

$$\varepsilon_{ij} = \frac{1}{2G} \left( \sigma_{ij} - \frac{\nu}{1+\nu} (\sigma_{xx} + \sigma_{yy}) \delta_{ij} \right). \quad (1.36)$$

Es sind nun alle Gleichungen der linearen Elastizitätstheorie vereint. Dies sind:

- 3 Bewegungsgleichungen (1.1) bzw. (1.2)
- 6 lineare geometrische Beziehungen (1.31)
- 6 Spannungs-Verzerrungs-Beziehungen (1.32) bzw. (1.33)

für die 15 Unbekannten

- 3 Verschiebungskomponenten  $u, v, w$
- 6 Verzerrungskomponenten  $\varepsilon_{ij}$
- 6 Spannungskomponenten  $\sigma_{ij}$  [1].

### 1.4 Verträglichkeitsbedingungen

Für die Komponenten des Verzerrungstensors gibt es noch einen weiteren Satz Gleichungen, der dadurch entsteht, dass man aus den Gln. (1.31) die Verschiebungen eliminiert. Dazu bildet man die dritten

Ableitungen der Verschiebungskomponenten und vertauscht die Differentiationsreihenfolgen, etwa

$$\frac{\partial^2}{\partial y^2} \left( \frac{\partial u}{\partial x} \right) = \frac{\partial^2}{\partial x \partial y} \left( \frac{\partial u}{\partial y} \right) \quad \text{usw.}$$

Es ergeben sich die folgenden Beziehungen, auch *Verträglichkeitsbedingungen* oder *Kompabilitätsbedingungen* genannt [1].

$$\begin{aligned} \frac{\partial^2 \varepsilon_{xx}}{\partial y^2} + \frac{\partial^2 \varepsilon_{yy}}{\partial x^2} &= \frac{\partial^2 \varepsilon_{xy}}{\partial x \partial y} & 2 \frac{\partial^2 \varepsilon_{xx}}{\partial y \partial z} &= \frac{\partial}{\partial x} \left( -\frac{\varepsilon_{yz}}{\partial x} + \frac{\varepsilon_{zx}}{\partial y} + \frac{\varepsilon_{xy}}{\partial z} \right) \\ \frac{\partial^2 \varepsilon_{yy}}{\partial z^2} + \frac{\partial^2 \varepsilon_{zz}}{\partial y^2} &= \frac{\partial^2 \varepsilon_{yz}}{\partial y \partial z} & 2 \frac{\partial^2 \varepsilon_{yy}}{\partial z \partial x} &= \frac{\partial}{\partial y} \left( -\frac{\varepsilon_{zx}}{\partial y} + \frac{\varepsilon_{xy}}{\partial z} + \frac{\varepsilon_{yz}}{\partial x} \right) \\ \frac{\partial^2 \varepsilon_{zz}}{\partial x^2} + \frac{\partial^2 \varepsilon_{xx}}{\partial z^2} &= \frac{\partial^2 \varepsilon_{zx}}{\partial z \partial x} & 2 \frac{\partial^2 \varepsilon_{zz}}{\partial x \partial y} &= \frac{\partial}{\partial z} \left( -\frac{\varepsilon_{xy}}{\partial z} + \frac{\varepsilon_{yz}}{\partial x} + \frac{\varepsilon_{zx}}{\partial y} \right) \end{aligned} \quad (1.37)$$

Dass zwischen den Verzerrungen Beziehungen bestehen müssen, ergibt sich schon rein dadurch, dass die 6 Komponenten des Verzerrungstensors nur von den 3 Verschiebungskomponenten  $u, v, w$  abhängen. Ansonsten käme bei völlig willkürlich wählbaren Verzerrungen kein Kontinuum ohne Lücken oder Selbstdurchdringungen zustande [1]. Auch die Beziehungen in Gl. (1.37) sind nicht völlig voneinander unabhängig. Durch weiteres zweimaliges Differenzieren gelingt es, diese auf drei unabhängige Gleichungen zu reduzieren [1].

## 1.5 Hauptnormalspannungen

Es wurde in Kapitel 1.1 der Spannungsbegriff hergeleitet und festgestellt, dass der Spannungszustand in einem Körperpunkt durch den symmetrischen Tensor 2. Stufe  $[\sigma]$  vollständig festgelegt ist. Der Spannungsvektor  $\mathbf{t}$  lässt sich in eine Normalspannungskomponente  $\sigma$  parallel zum Normalenvektor  $\mathbf{n}$  und in eine Schubspannungskomponente  $\tau$  parallel zur Ebene zerlegen, indem das Skalarprodukt

$$\sigma = \mathbf{t} \cdot \mathbf{n} = \sigma_{xx} n_x^2 + \sigma_{yy} n_y^2 + \sigma_{zz} n_z^2 + 2(\sigma_{xy} n_x n_y + \sigma_{xz} n_x n_z + \sigma_{yz} n_y n_z). \quad (1.38)$$

gebildet wird [1]. Abbildung 1.5 verdeutlicht dies zusätzlich.

Wird nun die Schnittebene gedreht, ändert sich auch der Normalenvektor und damit auch die Normalspannungskomponente nach Gl. (1.38). Es liegt nun die Frage nahe, in welcher Schnittebene die Normalspannungskomponente *maximal* wird.

### 1.5.1 Lagrange-Maximierungsproblem

Für die Beantwortung dieser Frage wird der Normalenvektor in Abhängigkeit des gewählten Koordinatensystemes gesucht, bei dem die linke Seite von Gl. (1.38) maximal wird. Für die Gültigkeit des Cauchy'schen Fundamentaltheorems aus Gl. (1.5) muss Gl. (1.6) gelten. Dies stellt nun ein Optimierungsproblem mit der Nebenbedingung Gl. (1.6) dar, welches üblicherweise mithilfe der Lagrange-Methode gelöst wird.

Nach [5] seien  $f(\mathbf{x}), g(\mathbf{x}): \mathbb{R}^n \rightarrow \mathbb{R}$  stetig differenzierbare Funktionen im  $\mathbb{R}^n$ .  $f$  sei zweimal stetig differenzierbar,  $g$  einmal stetig differenzierbar.  $\mathbf{n}$  ist ein lokales Extremum von  $f$  unter der Restriktion  $g(\mathbf{x}) = c$  und  $\nabla g(\mathbf{x}) \neq 0$ . Dann gibt es ein  $\lambda \in \mathbb{R}$ , dass als Lagrange-Multiplikator bezeichnet wird, mit

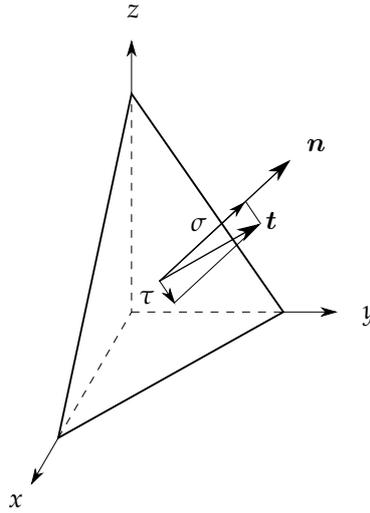


Abbildung 1.5: Spannungsvektor, zerlegt in Normalspannungs- und Schubspannungskomponente, nach [1].

$$\nabla f(\mathbf{x}) = -\lambda \nabla g(\mathbf{x}). \quad (1.39)$$

Führt man nun die Lagrange-Funktion

$$L(\lambda, \mathbf{x}) = f(\mathbf{x}) + \lambda (g(\mathbf{x}) - c) \quad (1.40)$$

ein, erkennt man, dass die notwendige Bedingung Gl. (1.39) zusammen mit der Nebenbedingung  $g(\mathbf{x}) = c$  äquivalent zur Formulierung

$$\nabla L(\lambda, \mathbf{x}) = 0 \quad (1.41)$$

ist [5]. Es wurde also mithilfe der Lagrange-Funktion aus Gl. (1.40) eine Formulierung gefunden, die ohne explizite Nebenbedingung auskommt. Die Lagrange-Funktion für das vorliegende Problem lautet nun

$$L(n_x, n_y, n_z) = \sigma(n_x, n_y, n_z) - \lambda (n_x^2 + n_y^2 + n_z^2 - 1). \quad (1.42)$$

Wendet man nun Gl. (1.41) an, erhält man dadurch ein Gleichungssystem, dessen Lösungen die Extremwerte der Funktion  $f$  darstellen. Angewendet auf das vorliegende Problem ergeben sich die Gleichungen

$$\begin{aligned} \frac{\partial L}{\partial n_x} &= 2\sigma_{xx} n_x + 2\sigma_{xy} n_y + 2\sigma_{xz} n_z - 2\lambda n_x = 0 \\ \frac{\partial L}{\partial n_y} &= 2\sigma_{xy} n_x + 2\sigma_{yy} n_y + 2\sigma_{yz} n_z - 2\lambda n_y = 0 \\ \frac{\partial L}{\partial n_z} &= 2\sigma_{xz} n_x + 2\sigma_{yz} n_y + 2\sigma_{zz} n_z - 2\lambda n_z = 0 \end{aligned} \quad (1.43)$$

$\frac{\partial L}{\partial \lambda}$  ergibt wiederum die Nebenbedingung und liefert keine neue Information. Gl. (1.43) ergibt nun ein homogenes lineares Gleichungssystem, das auch in Matrizenform dargestellt werden kann,

$$\begin{bmatrix} \sigma_{xx} - \lambda & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} - \lambda & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} - \lambda \end{bmatrix} \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad (1.44)$$

welches wiederum äquivalent mit den Schreibweisen

$$[\boldsymbol{\sigma}] \mathbf{n} = \lambda \mathbf{n}, \quad ([\boldsymbol{\sigma}] - \lambda [\mathbf{I}]) \mathbf{n} = \mathbf{0} \quad (1.45)$$

ist.

Gleichung (1.45) stellt ein Eigenwertproblem dar. Man sucht dabei im Allgemeinen komplexe Zahlen  $\lambda$ , genannt Eigenwerte, und im Allgemeinen komplexe Vektoren  $\mathbf{n}$ , genannt Eigenvektoren, so dass (1.45) erfüllt ist. Für eine nichttriviale Lösung dieses Gleichungssystems muss die Bedingung

$$\det([\boldsymbol{\sigma}] - \lambda[\mathbf{I}]) = \begin{vmatrix} \sigma_{xx} - \lambda & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_{yy} - \lambda & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} - \lambda \end{vmatrix} = 0 \quad (1.46)$$

gelten [6]. Die Determinante des Problems in Gl. (1.46) ist ein Polynom dritten Grades in  $\lambda$ , welches analytisch noch lösbar ist. Die dazugehörigen Eigenvektoren findet man durch Lösung des Gleichungssystems, wobei die euklidische Länge der Eigenvektoren bzw. das Vorzeichen dieser nicht gegeben ist, sondern nur die Richtung der Eigenvektoren [6].

Die Eigenwerte des Problems sind genau die Hauptnormalspannungen  $\sigma_1 = \lambda_1, \sigma_2 = \lambda_2, \sigma_3 = \lambda_3$  mit den dazugehörigen Eigenvektoren  $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$ , wobei die Eigenvektoren in Länge und Richtung bis auf das Vorzeichen durch Gl. (1.44) bestimmt sind [1]. Per Konvention sind die Hauptnormalspannungen durch die Ungleichungskette

$$\sigma_1 \geq \sigma_2 \geq \sigma_3 \quad (1.47)$$

sortiert [7]. Transformiert man den Spannungstensor in ein Koordinatensystem, das genau von den Eigenvektoren  $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$  aufgespannt wird, transformiert sich der Spannungstensor zu

$$[\boldsymbol{\sigma}] = \begin{bmatrix} \sigma_{xx} & \sigma_{yx} & \sigma_{zx} \\ \sigma_{xy} & \sigma_{yy} & \sigma_{zy} \\ \sigma_{xz} & \sigma_{yz} & \sigma_{zz} \end{bmatrix} \rightarrow [\boldsymbol{\sigma}]_H = \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix}. \quad (1.48)$$

Es verschwinden also die Schubspannungen bei der Darstellung des Spannungstensors in Hauptachsensform [7].

## 1.5.2 Eigenschaften

Es stellt sich nun die Frage, welche Auswirkungen die Symmetrie einer Matrix auf dessen Eigenschaften hat, besonders im Hinblick auf die Eigenschaften der Hauptnormalspannungen und Hauptnormalspannungsvektoren.

### Reelle Eigenwerte

Wegen der Symmetrie des Spannungstensors bzw. Verzerrungstensors sind die drei Wurzeln der kubischen Gleichung  $\lambda_1, \lambda_2, \lambda_3$  reell, wie etwa in [6] oder [8] gezeigt wird. Dies deckt sich mit der Erfahrung, da eine komplexe (Haupt-) Spannung ja gar nicht definiert werden kann.

### Orthogonale Eigenvektoren

Sind  $\mathbf{n}_1$  und  $\mathbf{n}_2$  Eigenvektoren zu den unterschiedlichen Eigenwerten  $\lambda_1$  und  $\lambda_2$ , dann sind die Eigenvektoren orthogonal zueinander [8].

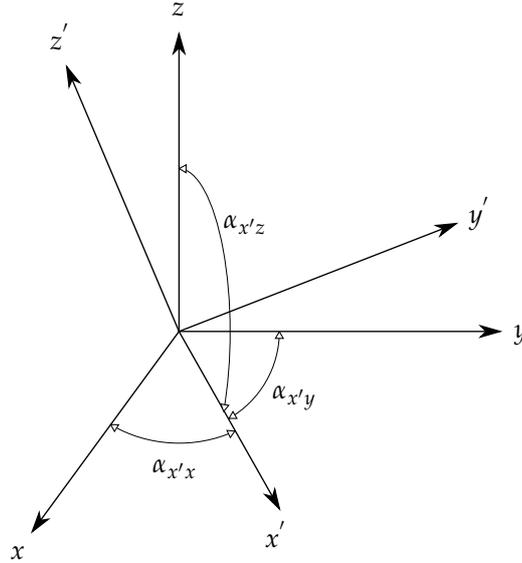


Abbildung 1.6: Rotation des Koordinatensystems

## 1.6 Hauptnormalverzerrungen

Dieses Kapitel wurde in Anlehnung an [7] verfasst. Analog zur Ermittlung der Hauptnormalspannungen stellt sich nun die Frage, in welchen Richtungen die Normalverzerrungen  $\varepsilon_{xx}$ ,  $\varepsilon_{yy}$ ,  $\varepsilon_{zz}$  Extremwerte annehmen. Da es sich hierbei um eine physikalische Eigenschaft des deformierten Körperpunkts handelt, liegt keine Abhängigkeit vom gewählten Koordinatensystem vor.

Die Extremwerte der Normalverzerrungen werden als Haupt(normal)verzerrungen bezeichnet, die Wirkungsrichtungen als (Normal)verzerrungshauptrichtungen. Bei der Bestimmung dieser Richtungen geht man von der Darstellung eines Tensors 2. Stufe in einem anderen Koordinatensystem aus. Die Drehmatrix  $[Q]$  mit

$$[Q] = \begin{bmatrix} n_{xx} & n_{xy} & n_{xz} \\ n_{yx} & n_{yy} & n_{yz} \\ n_{zx} & n_{zy} & n_{zz} \end{bmatrix} = \begin{bmatrix} \cos \alpha_{x'x} & \cos \alpha_{x'y} & \cos \alpha_{x'z} \\ \cos \alpha_{y'x} & \cos \alpha_{y'y} & \cos \alpha_{y'z} \\ \cos \alpha_{z'x} & \cos \alpha_{z'y} & \cos \alpha_{z'z} \end{bmatrix} \quad (1.49)$$

bestimmt sich aus den Winkeln zwischen den Koordinatenachsen  $x'$ ,  $y'$ ,  $z'$  relativ zu den Koordinatenachsen  $x$ ,  $y$ ,  $z$ . Die Drehwinkel  $\alpha_{ij}$  sind dabei nach Abb. 1.6 definiert.

Ein Tensor 2. Stufe, wie der Verzerrungstensor, kann im Koordinatensystem  $x'$ ,  $y'$ ,  $z'$  in der Form

$$[\varepsilon]' = [Q][\varepsilon][Q]^T \quad (1.50)$$

angegeben werden. Weiters müssen die Richtungsosini der Bedingung  $n_{ix}^2 + n_{iy}^2 + n_{iz}^2 = 1, i = x, y, z$  gehorchen.

Es wird nun beispielweise der Maximalwert für  $\varepsilon'_{xx} = f(n_{xx}, n_{xy}, n_{xz})$  gesucht. Aus Gleichung (1.50) ergibt sich

$$\varepsilon'_{xx} = \varepsilon_{xx} n_{xx}^2 + \varepsilon_{yy} n_{xy}^2 + \varepsilon_{zz} n_{xz}^2 + 2(\varepsilon_{xy} n_{xx} n_{xy} + \varepsilon_{xz} n_{xx} n_{xz} + \varepsilon_{yz} n_{xy} n_{xz}). \quad (1.51)$$

Man kann leicht erkennen, dass Gl. (1.51) bis auf die Koeffizienten ident zu Gl. (1.38) ist, wobei auch

die Nebenbedingung identisch ist. Somit stellt die Ermittlung der Hauptnormalverzerrungen ebenfalls ein Lagrange-Maximierungsproblem dar, welches im vorliegenden Fall wieder auf ein Eigenwertproblem

$$[\varepsilon] \mathbf{n} = \lambda \mathbf{n} \quad (1.52)$$

führt. Für die Hauptverzerrungen und Hauptverzerrungsrichtungen gelten somit ebenfalls die genannten Eigenschaften aus Kap. 1.5.2, da auch der Verzerrungstensor ein reeller, symmetrischer Tensor ist.

## 1.7 Zweck der Hauptnormalspannungen

Es wird eine keinesfalls vollständige Auflistung über den Zweck der Hauptnormalspannungen und Hauptnormalverzerrungen wiedergegeben.

### 1.7.1 Versagenshypothesen

Eine entscheidende Rolle spielen die Hauptnormalspannungen bei der Formulierung von Versagenshypothesen. Im Gegensatz zur einachsigen Belastung, wie sie etwa beim klassischen Zugversuch aus der Werkstoffprüfung vorherrscht, kommt es bei der Belastung von Bauteilen im Maschinenbau zumindest in einzelnen Bereichen zu zwei- oder dreiachsigen Belastungen [7]. Über Anstrengungs- bzw. Versagungshypothesen wird versucht, eine Vergleichbarkeit zwischen einachsigen und mehrachsigen Beanspruchungen herzustellen [7]. Erreicht die errechnete Vergleichsspannung im Bauteil den Vergleichsspannungswert beim Versagen durch Bruch oder Fließen im einachsigen Zugversuch, ist auch im Bauteil örtlich mit Fließen oder mit Bruch zu rechnen [7].

Anstrengungshypothesen für isotrope Werkstoffe lassen sich in einem von den Koordinatenachsen  $\sigma_1, \sigma_2, \sigma_3$  aufgespannten Raum, auch Hauptspannungsraum genannt, darstellen. Der Spannungszustand im Körperpunkt lässt sich daraufhin als Spannungspunkt im Hauptspannungsraum eintragen. Bleibt dieser Spannungspunkt innerhalb einer von der Anstrengungshypothese definierten Fläche, der Fließ- oder Bruchfläche, kommt es zu elastischem Materialverhalten. Liegt der Spannungspunkt jedoch auf der Fließ- oder Bruchfläche, kommt es zum Fließen oder Bruch [7]. Ein Spannungspunkt außerhalb der Fließ- oder Bruchfläche ist physikalisch nicht möglich, da es vorher zu einer Spannungsumlagerung durch Fließen oder zum Bruch des Bauteils kommt [7].

Im Folgenden werden einige einfache Versagenshypothesen mit einem Werkstoffkennwert genannt, wobei Teile aus [7] übernommen wurden.

#### Versagenshypothese nach Tresca

Tresca ging in seinen Überlegungen davon aus, dass Fließen im Werkstoff durch das Gleiten der Metallkristalle längs bestimmter Ebenen hervorgerufen wird. Daher wird angenommen, dass für den Fließbeginn die größte Schubspannung im Körper maßgeblich ist [7]. Es kann gezeigt werden, dass die größte Schubspannung  $\tau_{max}$  im Körper durch

$$\tau_{max} = \frac{1}{2} (\sigma_1 - \sigma_3) \quad (1.53)$$

gegeben ist, sofern die Hauptnormalspannungen nach Gl. (1.47) geordnet sind [1]. Dieser Wert wird mit der maximalen Schubspannung beim Zugversuch verglichen, welche

$$\tau_{max} = \frac{1}{2} \sigma_F \quad (1.54)$$

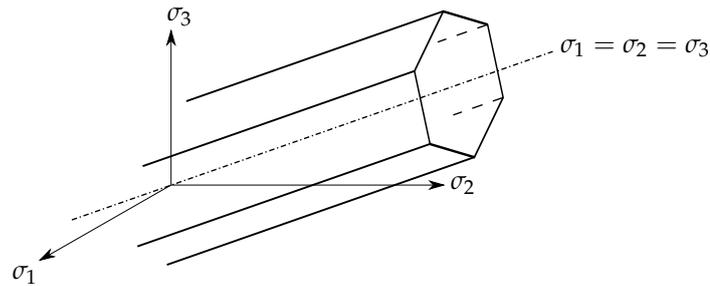


Abbildung 1.7: Versagenskriterium nach Tresca im Hauptspannungsraum, nach [7]

im Moment des Fließbeginns bei der Zugspannung  $\sigma_F$  beträgt [7]. Daher lautet die Fließbedingung nach Tresca

$$\sigma_1 - \sigma_3 = \sigma_F. \quad (1.55)$$

und die Vergleichsspannung nach Tresca

$$\sigma_V = \sigma_1 - \sigma_3 \quad \begin{array}{l} \text{kein Fließen, wenn } \sigma_V < \sigma_F \\ \text{Fließen, wenn } \sigma_V = \sigma_F \end{array} \quad (1.56)$$

[7]. Gleichung (1.55) kann im Hauptspannungsraum nach Abb. 1.7 dargestellt werden, wobei die Ordnung der Vergleichsspannungen nach Gl. (1.47) verworfen wird. Dies führt zu 6 Ebenengleichungen.

Man kann leicht erkennen, dass die Fließfläche bzw. die Fließbedingung, die durch Gl. (1.55) beschrieben wird, parallel zur Achse  $\sigma_1 = \sigma_2 = \sigma_3$  verläuft, was die Unabhängigkeit des Fließbeginns von der Höhe des hydrostatischen Druckanteils sehr gut verdeutlicht [7].

### Versagenshypothese nach Mises

Ein zweites wichtiges Versagenskriterium für zähe Materialien stammt von Richard von Mises. Er geht in seinen Überlegungen davon aus, dass Fließen nur eintritt, wenn die Verformungen des Körpers mit einer Gestaltsänderung verbunden sind. Eine reine Volumsänderung als Resultat eines hydrostatischen Druckes führt zu keinem Fließen im Körper [7].

Die von Mises-Vergleichsspannung in Abhängigkeit der Hauptnormalspannungen beträgt

$$\sigma_V = \sqrt{\frac{1}{2} [(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2]} \quad \begin{array}{l} \text{kein Fließen, wenn } \sigma_V < \sigma_F \\ \text{Fließen, wenn } \sigma_V = \sigma_F \end{array} \quad (1.57)$$

[1]. Die Fließfläche kann wieder im Hauptspannungsdiagramm gezeichnet werden. Diese stellt beim Mises-Kriterium einen Zylindermantel dar, der wie beim Fließkriterium von Tresca eine unendliche Ausdehnung entlang der hydrostatischen Achse besitzt, siehe Abb. 1.8. Für den ebenen Spannungszustand lassen sich die Spannungskriterien von Tresca und von Mises ebenfalls darstellen, siehe Abb. 1.9, wobei die strichlierte Linie das Versagenskriterium nach Mises und die durchgezogene Linie das Versagenskriterium nach Tresca darstellt.

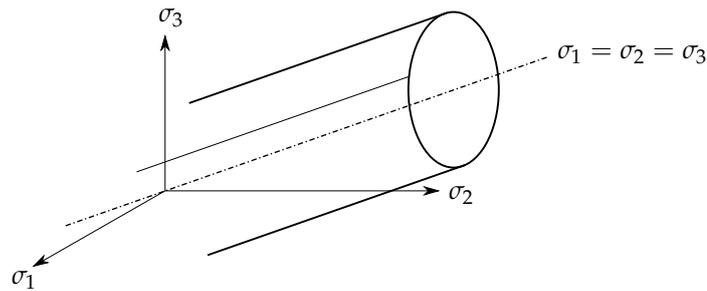


Abbildung 1.8: Versagenskriterium nach Mises im Hauptspannungsraum, nach [7]

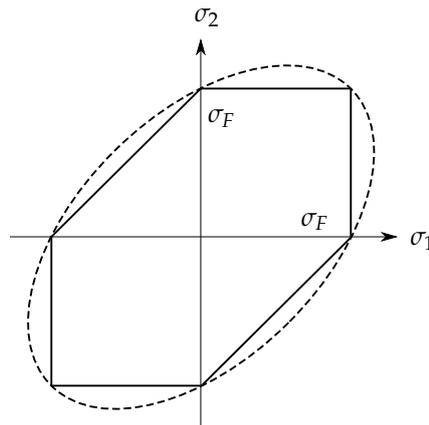


Abbildung 1.9: Versagenskriterium nach Mises und Tresca für den ebenen Spannungszustand, nach [7]

## 1.8 Lösung des Eigenwertproblems

Die Lösung des Eigenwertproblems ist ein klassisches Aufgabengebiet in der Numerik. Obwohl für den genannten Anwendungszweck, die Berechnung der Hauptnormalspannungen und der Hauptnormalverzerrungen, lediglich  $2 \times 2$  oder  $3 \times 3$  Matrizen gelöst werden müssen, wird trotzdem ein kurzer Einblick in die numerische Lösungsmethoden gegeben. Auch deshalb, weil in der Implementierung nach Kap. 2 keine analytischen Lösungsmethoden angewendet werden, sondern bereits vorhandene Lösungsalgorithmen aus LAPACK (Linear Algebra Package). Da diese sehr gut getestet und zuverlässig sind, obwohl sie eigentlich für größere Matrizen entworfen wurden.

Wie bereits in Gl. (1.46) gezeigt wurde, können die Eigenwerte durch Lösung des charakteristischen Polynoms gelöst werden. Für eine  $n \times n$  Matrix ergibt sich daraus ein Polynom  $n$ -ter Ordnung. Problematisch dabei ist allerdings, dass die Nullstellen eines Polynoms schon für ein relativ niedriges  $n$  zu einem schlecht konditionierten Problem führt, das heißt dass schon eine kleine Änderung der Koeffizienten zu einem großen Unterschied in der Lösung führt [9]. Diese Methode wird deswegen im Allgemeinen in der Numerik nicht angewandt [9].

Abel zeigte, dass die Nullstellen eines Polynoms mit Ordnung  $m > 4$  nicht geschlossen als Formel angegeben werden können [10]. Daraus folgt, dass die Eigenwerte nicht exakt, sondern nur iterativ gelöst werden können [9]. Es existieren jedoch iterative Methoden, die recht schnell konvergieren, wie in den nächsten Kapiteln gezeigt wird.

### 1.8.1 Begriffe aus der linearen Algebra

Es sei für die nachfolgende Auflistung eine Matrix  $[A] \in \mathbb{C}^{n \times m}$  gegeben.  $A_{ij}$  ist der Eintrag der Matrix  $[A]$  in der Zeile  $i$  und in der Spalte  $j$ .

- Eine quadratische Matrix  $[A]$  ist hermitesch, wenn  $A_{ij} = \overline{A_{ji}}$  gilt.
- Eine quadratische Matrix ist symmetrisch, wenn  $A_{ij} = A_{ji}$  gilt. Es gilt demnach  $[A] = [A]^T$ , das heißt die Matrix ist gleich ihrer transponierten Matrix.
- Eine quadratische Matrix  $[Q] \in \mathbb{Q}^{n \times n}$  ist unitär, wenn  $\overline{[Q]}^T [Q] = [I]$  gilt.
- Eine quadratische Matrix  $[Q] \in \mathbb{R}^{n \times n}$  ist orthogonal, wenn  $[Q]^T [Q] = [I]$  gilt. Eine orthogonale Matrix ist somit ein Spezialfall der unitären Matrix.
- Dreiecksmatrix: Eine Matrix  $[A] \in \mathbb{C}^{n \times m}$  ist eine obere Dreiecksmatrix, wenn die Einträge der Matrix unterhalb der Hauptdiagonalen Null sind.  $[A]$  ist eine untere Dreiecksmatrix, wenn die Einträge der Matrix oberhalb der Hauptdiagonalen Null sind.
- Hessenbergmatrix: Eine quadratische Matrix  $[A] \in \mathbb{C}^{n \times n}$  ist eine obere Hessenbergmatrix, wenn die Einträge der Matrix unterhalb der ersten (unteren) Nebendiagonalen Null sind. Analog dazu ist die untere Hessenbergmatrix definiert.
- Tridiagonalmatrix: Eine quadratische Matrix  $[A] \in \mathbb{C}^{n \times n}$  ist eine Tridiagonalmatrix, wenn ihre Einträge nur in der Hauptdiagonalen und den ersten beiden Nebendiagonalen von Null abweichen.

### 1.8.2 Ähnlichkeitstransformationen

Viele numerischen Methoden zur Eigenwertberechnung haben gemein, dass mittels Ähnlichkeitstransformationen versucht wird, die Matrix in eine Diagonalmatrix oder in eine Dreiecksmatrix zu überführen. Ähnlichkeitstransformationen haben die Eigenschaft, dass sie die Eigenwerte einer Matrix nicht ändern [11]. Die Eigenwerte können schlussendlich auf der Diagonalen der transformierten Matrix abgelesen werden, denn die Eigenwerte einer Diagonalmatrix bzw. Dreiecksmatrix sind genau die Einträge in der Hauptdiagonalen [11].

Da die Ähnlichkeitstransformation ein zentraler Bestandteil vieler Eigenwertalgorithmen ist, wird diese hier näher betrachtet. Unter einer Ähnlichkeitstransformation einer Matrix  $[A] \rightarrow [B]$  versteht man eine Transformation der Form

$$[B] = [P] \cdot [A] \cdot [P]^{-1} \quad (1.58)$$

mithilfe einer regulären und damit invertierbaren Transformationsmatrix  $[P]$  [11]. Man kann mithilfe des charakteristischen Polynoms beweisen, dass die Eigenwerte von  $[A]$  und  $[B]$  gleich sind, was etwa in [9] gezeigt wird.

### 1.8.3 QR-Zerlegung

#### Householder-Zerlegung

Die folgenden Überlegungen stammen aus [9]. Für die folgenden Kapitel zur numerischen Eigenwertberechnung einer Matrix muss geklärt werden, wie eine Matrix  $[A]$  in eine Dreiecksmatrix  $[R]$  und in eine orthonormale Matrix  $[Q]$  zerlegt werden kann. Dies kann beispielsweise mit der klassischen Gram-Schmidt-Orthogonalisierung durchgeführt werden. Da dieses klassische Verfahren zur QR-Zerlegung aber numerisch instabil ist, wird es im Allgemeinen nicht verwendet.

Eine numerisch stabilere Möglichkeit der QR-Zerlegung stellt die Householder-Transformation dar. Diese transformiert  $[A]$  in eine obere Dreiecksmatrix  $[R]$  mittels einer Folge von Multiplikationen mit unitären Transformationsmatrizen  $[Q_k]$  so, dass sukzessive unterhalb der Hauptdiagonalen Nullen entstehen. Diese können geschrieben werden als

$$[Q_k] = \begin{bmatrix} [I_k] & [0] \\ [0] & [F_k] \end{bmatrix}, \quad (1.59)$$

wobei hier  $[I_k]$  eine  $(k-1) \times (k-1)$  Einheitsmatrix und  $[F_k]$  eine unitäre  $(m-k+1) \times (m-k+1)$  Matrix ist. Der Householder-Reflektor ist jene Matrix, welche den Vektor  $x$  auf  $e_1$  projiziert, dass also  $[F]x = \pm \|x\| e_1$  gilt. Dieser kann als

$$[F_k] = \left( [I] - 2 \frac{v \otimes v}{v \cdot v} \right) \quad \text{mit } v = \text{sign}(x_1) \|x\| e_1 + x \quad (1.60)$$

angeschrieben werden. Ein großer Vorteil der Methode ist, dass der Vektor  $x$  im reellen Fall auf 2 verschiedene und im komplexen Fall auf unendlich viele Arten auf  $e_1$  projiziert werden kann. Dadurch kann man numerisch ungünstigen Fällen aus dem Weg gehen.

Eine Möglichkeit zur Implementierung gibt Algorithmus 1.1, wobei  $[A]$  eine  $m \times n$  Matrix ist und die Notation  $[A_{k:m,k:n}]$  eine Untermatrix von  $[A]$  darstellt, welche aus den Zeilen  $k$  bis  $m$  und den Spalten  $k$  bis  $n$  von  $[A]$  besteht. Nach Durchlauf des Algorithmus ist  $[A]$  zur Dreiecksmatrix  $[R]$  transformiert.

---

**Algorithmus 1.1** Householder-Zerlegung

---

```

for  $k = 1$  to  $n$  do
   $x = [A_{k:m,k}]$ 
   $v_k = \text{sign}(x_1) \|x\|_2 e_1 + x$ 
   $v_k = \frac{v_k}{\|v_k\|_2}$ 
   $[A_{k:m,k:n}] = [A_{k:m,k:n}] - 2 v_k (v_k^T [A_{k:m,k:n}])$ 
end for

```

---

### 1.8.4 Transformation auf Hessenbergform

Zur Transformation einer Matrix  $[A] = [Q]^* [T] [Q]$  in eine Hessenbergform wird eine etwas abgewandelte Variante der Householder-Zerlegung angewandt. Es wird im ersten Schritt ein Householder-Reflektor gewählt, der die erste Zeile bzw. Reihe der Matrix  $[A]$  nicht verändert. Fett gedruckte Einträge werden durch die Multiplikation mit der Matrix  $[Q]$  verändert [9].

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{bmatrix} \xrightarrow{[Q_1]^* \cdot} \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times \end{bmatrix} \xrightarrow{\cdot [Q_1]} \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times \\ \mathbf{0} & \times & \times & \times & \times \end{bmatrix} \quad (1.61)$$

Dieser Schritt wird wiederholt, um in den Spalten unterhalb der ersten Nebendiagonalen sukzessive Nullen zu erzeugen, dabei lässt der Reflektor  $[Q]_i$  die jeweils ersten  $i$  Spalten bzw. Reihen unberührt

[9]. Dieser Vorgang wird  $m - 2$  mal wiederholt, bis sich die Form

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \times & \times \\ & & & & \times & \times \end{bmatrix} \underbrace{[Q_{m-2}]^* \dots [Q_2]^* [Q_1]^* [A]}_{[Q]^*} \underbrace{[Q_1][Q_2] \dots [Q_{m-2}]}_{[Q]} = [H] \quad (1.62)$$

ergibt [9].

Wenn  $[A]$  hermitesch ist, führt der oben genannte Algorithmus zu einer Tridiagonalform [9]. Es wird in Kap. 1.8.5 noch ersichtlich werden, dass die Transformation zu einer Tridiagonalform für die numerische Berechnung der Eigenwerte sehr hilfreich ist [9].

### 1.8.5 Algorithmen zur Lösung des Eigenwertproblems

Moderne Eigenwertalgorithmen arbeiten in einem zweistufigen Verfahren. In einem ersten Schritt wird die Matrix in einer Abfolge von Ähnlichkeitstransformation auf eine Hessenbergform oder eine Tridiagonalform gebracht, während in einem zweiten Schritt iterativ die Lösung des Eigenwertproblems erfolgt.

#### Vorteil einer Hessenbergform oder Tridiagonalform

Die Transformation in eine obere Hessenbergmatrix bzw. eine Tridiagonalform erfolgt aus Performancegründen. Eine Reduktion auf eine obere Hessenbergmatrix erfordert  $\mathcal{O}(m^3)$  Rechenoperationen (*flops*). Die zweite, iterative Phase erfordert üblicherweise  $\mathcal{O}(m)$  Iterationen, bis eine Konvergenz auf Maschinengenauigkeit gegeben ist. Jede Iteration erfordert  $\mathcal{O}(m^2)$  Rechenoperationen. Die gesamte Eigenwertberechnung erfordert somit  $\mathcal{O}(m^3 + m \cdot m^2) = \mathcal{O}(m^3)$  Rechenoperationen [9].

Ist die Matrix, die einer Eigenwertberechnung unterworfen wird, eine hermitesche Matrix, führt die Transformation auf eine Hessenbergform zu einer Tridiagonalmatrix. Dann kann jeder Iterationsschritt in  $\mathcal{O}(m)$  Rechenoperationen ausgeführt werden, was den Algorithmus noch schneller macht [9].

Entfällt jedoch die Transformation in eine obere Hessenbergmatrix, erfordert jeder Iterationsschritt  $\mathcal{O}(m^3)$  Rechenoperationen, wobei wiederum  $\mathcal{O}(m)$  Iterationen ausgeführt werden müssen, um Konvergenz zu erreichen. Somit würde die Eigenwertberechnung  $\mathcal{O}(m \cdot m^3) = \mathcal{O}(m^4)$  Rechenoperationen benötigen, was klar im Nachteil zur oben genannten Variante steht [9].

#### QR-Algorithmus

Die einfachste Methode zur numerischen Berechnung der Eigenwerte ist der QR-Algorithmus. Bei diesem Verfahren wird die Matrix  $[A]$  wiederholt einer Ähnlichkeitstransformation mittels unitärer Transformationsmatrizen unterworfen. Die Iteration lautet

$$[A_{k+1}] = [Q_k]^{-1} \cdot [A_k] \cdot [Q_k], \quad (1.63)$$

beziehungsweise in der Form des Algorithmus 1.2 [9]. Die Analogie zu Gl. (1.63) lässt sich erkennen, wenn man die QR-Faktorisierung zu  $[R_k] = [Q_k]^{-1} \cdot [A_{k-1}]$  umformt und dies in die Rekombinationsgleichung einsetzt, was eben genau auf Gl. (1.63) führt [6].

Wie man schnell erkennen kann, ist dies in Widerspruch zu Kap. 1.8.4, da durch die Multiplikation mit  $[Q_k]$  von rechts die Diagonalform wieder zerstört wird [9]. Jedoch zeigt sich, dass bei einer ite-

---

**Algorithmus 1.2** QR-Algorithmus

---

```
[A0] = [A]
for k = 1, 2, ... do
    [Qk] · [Rk] = [Ak-1]    QR-Faktorisierung
    [Ak] = [Rk] · [Qk]    Rekombination
end for
```

---

rativen Anwendung von Algorithmus 1.2 die Matrix  $[A]$  gegen eine Diagonalmatrix konvergiert. Der Beweis dazu ist recht aufwendig und wird etwa in [9] geführt.

**QR-Algorithmus mit Shifts**

Eine verbesserten Algorithmus zur Eigenwertberechnung ist der QR-Algorithmus mit Shifts. Die Idee des QR-Algorithmus mit Shift ist, den QR-Schritt mit einer veränderten Matrix  $[A]$  auszuführen, die aus der um den geschätzten Eigenwert reduzierten Originalmatrix besteht, mit dem Ziel, eine rasche Konvergenz eines Eigenwerts (dem letzten) zu erreichen [12]. Sobald ein Eigenwert gefunden wurde, wird die entsprechende Zeile und Spalte der Matrix gestrichen. Man spricht dabei von *Deflation* [9], wobei hier nicht näher darauf eingegangen wird. Der QR-Algorithmus mit Shifts kann nun in der Form des Algorithmus 1.3 geschrieben werden.

---

**Algorithmus 1.3** QR-Algorithmus mit Shifts

---

```
for k = 1, 2, ... do
    [Qk] · [Rk] = [Ak-1] - μk[I]    QR-Faktorisierung von [Ak-1] - μk[I]
    [Ak] = [Rk] · [Qk] + μk[I]    Rekombination
end for
```

---

Der Rayleigh-Quotient  $\mu_k$  ist eine Schätzung für einen Eigenwert, je besser die Schätzung ist desto schneller konvergiert der Algorithmus [9].

**Weitere Algorithmen**

Weitere Algorithmen zur Eigenwertberechnung einer Matrix sind:

- Multishiftverfahren nach Bai und Demmel [13]
- Divide-and-Conquer-Algorithmus
- Jacobi-Algorithmus
- Bisektions-Methode

Für sehr große, dünnbesetzte Matrizen stehen weitere, iterative Algorithmen wie die *Lanczos*-Iteration oder die *Arnoldi*-Iteration zur Verfügung. Unter bestimmten Umständen benötigen diese weniger als  $\mathcal{O}(m^3)$  Rechenoperationen [9] [14].

# Kapitel 2

## Implementierung

### 2.1 Numerische Methoden in der Entwicklung

In der Industrie und der Wissenschaft werden zunehmend numerische Verfahren bei der Entwicklung mechatronischer Systeme eingesetzt, welche die Simulation solcher Komponenten und Systeme ermöglichen. Weiters sind Entwickler aufgrund der Komplexität der Systeme oft nicht mehr in der Lage, mit analytischen Methoden die im System vorherrschenden Größen zu berechnen beziehungsweise sie messtechnisch zu ermitteln [15].

Generell sind die Größen, welche die physikalischen Effekte in mechatronischen Systemen beschreiben, eine Funktion von Ort und Zeit und führen auf ein System von partiellen Differentialgleichungen. Diese können im Allgemeinen nicht analytisch gelöst werden und erfordern die Anwendung von numerischen Verfahren. Dabei stehen etwa das Finite Differenzen Verfahren, das Finite Volumen Verfahren oder das Finite Elemente Verfahren zur Verfügung [15].

### 2.2 Aufbau von CFS++

Für das Simulationsprogramm *Coupled Field Simulation (CFS++)* wurde das Finite Elemente (FE) - Verfahren gewählt, da es sich durch numerische Effizienz, Möglichkeit zur Diskretisierung von komplexen Geometrien und vielfältigen Analysemöglichkeiten auszeichnet. Seit dem Jahr 2000 wird das Programm CFS++ entwickelt, wobei eine Behandlung von Mehrfeldproblemen bei der Entwicklung ein zentrales Ziel darstellt [15].

CFS++ beinhaltet keine Programmteile zum Preprocessing, also etwa zum Erstellen des Meshes oder zum Postprocessing in Form einer grafischen Darstellung der gewünschten Ergebnisse. Die Steuerung des FE-Programms geschieht mittels Eingabe-Dateien im xml-Format, Mesh-Files von verschiedenen externen Preprozessoren können zur Verfügung gestellt werden [15].

Das Programm und somit auch der implementierte Teil wurde in der Programmiersprache C++ verfasst, das sich durch hohe Performance und eine hohe Objektorientierung auszeichnet. Es ist darauf ausgelegt, möglichst flexibel zu sein, um Erweiterungen im Zuge von Forschungsprojekten, etwa durch Implementierung weiterer partieller Differentialgleichungen oder Ergebnistypen, zu realisieren [15].

Für die Berechnung der Hauptnormalspannungen und Hauptnormalverzerrungen wurde der Fokus auf das Ergebnis-Postprocessing gelegt, wobei hier die Berechnung abgeleiteter Ergebnisse aus einer mechanischen Finite-Elemente-Analyse gemeint ist (und nicht die grafische Darstellung der berechneten Ergebnisse in externen Programmen, wie etwa *Paraview*). Dieses findet in CFS++ mithilfe sogenannter *Coefficient Functions* statt, die aus dem primären Ergebnis einer mechanischen FE-Analyse,

dem Verschiebungsvektor  $u$ , die abgeleiteten Ergebnisse berechnet.

Wie in Abb. 2.1 ersichtlich, wurden für die Berechnung der Hauptnormalspannungen neue sogenannte *Coefficient Functions* implementiert und bereits vorhandene angewendet.

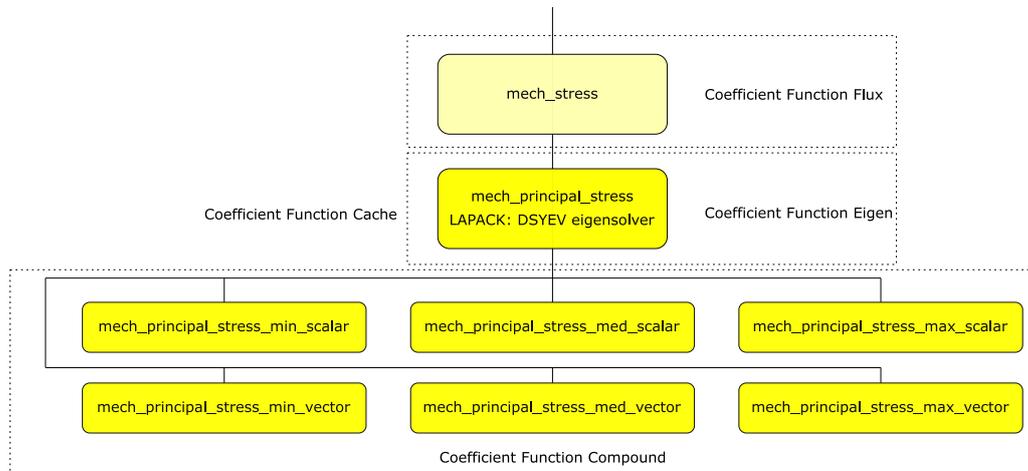


Abbildung 2.1: Schema zur Implementierung der Hauptnormalspannungen

Ausgangspunkt zur Berechnung der Hauptnormalspannungen war die Ermittlung des Spannungstensors  $[\sigma]$ , welcher bereits in Form der *Coefficient Function Flux* implementiert war. In der neu implementierten *Coefficient Function Eigen* findet die Berechnung der Eigenwerte und Eigenvektoren des Spannungstensors statt, wie in Kap. 1 ausführlich erklärt wurde. In den jeweiligen *Coefficient Function Compound* findet eine Zuweisung der Eigenwerte und Eigenvektoren zur gewünschten Hauptnormalspannung, der maximalen, der minimalen oder der mittleren (im 3D-Fall), statt. Um eine korrekte Darstellung im Postprocessing zu ermöglichen, existiert von jeder Hauptnormalspannung bzw. -verzerrung eine Variante, die einen skalaren oder vektoriellen Wert zurückgibt. Dies ist zumindest notwendig, um eine eindeutige Unterscheidung zwischen positiven und negativen Hauptnormalspannungen bzw. -verzerrungen zu ermöglichen, da diese Information verlorengeht, wenn nur die Länge eines Hauptnormalspannungsvektors im Postprocessing ausgegeben wird.

## 2.3 Implementierung in der MechPDE

Zentraler Ort der Implementierung der Hauptnormalspannungen und -verzerrungen in CFS++ ist die MechPDE, also die Implementierung der mechanischen Differentialgleichung. In dieser sei besonders auf die beiden Funktionen `DefinePrimaryResults()` und `DefinePostProcResults()` hingewiesen.

In der erstgenannten wird der Ergebnistyp einer mechanischen Finite-Elemente-Analyse, der Vektor der Verschiebungscomponenten der Knoten  $u$ , festgelegt. Außerdem werden zusätzlich notwendige Angaben zum Lösungstyp, wie die Anzahl und Benennung der Freiheitsgrade, in einem Pointer des Datentyps `ResultInfo` hinterlegt.

Alle anderen Ergebnistypen einer mechanischen Analyse sind in der Funktion `DefinePostProcResults()` definiert. Dazu zählen beispielsweise bei einer nichtstatischen Analyse der Geschwindigkeitsvektor  $v = \frac{\partial u}{\partial t}$  oder der Beschleunigungsvektor  $a = \frac{\partial^2 u}{\partial t^2}$ , weiters die abgeleiteten Ergebnistypen mechanische Spannungen und Verzerrungen.

Wie bereits in Kapitel 2.2 erwähnt wurde, erfolgt die eigentliche Berechnung der abgeleiteten Ergebnistypen in eigenen *Coefficient Functions*. Diese stellen verschiedene Operationen dar, sind im Sinne einer Finiten Elemente Analyse definiert und können je nach geforderter physikalischer Größe auf das

Primärresultat oder ein abgeleitetes Resultat angewendet werden.

Als einfaches Beispiel ist die Berechnung des Verzerrungstensors genannt. Diese berechnet sich aus einer Multiplikation mit der Differentialoperatormatrix  $\mathcal{B}$

$$S = \mathcal{B} u, \quad (2.1)$$

wobei hier der Verzerrungstensor  $S$  in Voigt-Notation gegeben ist [3]. Die `CoefFunctionB0p` berechnet zu diesem Zweck die Differentialoperatormatrix im geforderten lokalen Punkt im Element und wendet diese auf den Verschiebungsvektor an.

## 2.4 CoefFunctionEigen

Für die Lösung des Eigenwertproblems wurde die *Coefficient Function Eigen*, oder kurz `CoefFunctionEigen` angelegt. Zentraler Bestandteil dieser Coefficient Function ist die Berechnung der Eigenwerte und Eigenvektoren des Spannungs- oder Verzerrungstensors durch einen Eigenwertlöser aus dem *Linear Algebra Package (LAPACK)*. Dies ist eine Sammlung von Algorithmen zur Lösung linearer Gleichungssysteme, linearer Ausgleichsprobleme und eben Eigenwertproblemen. Die dazu notwendigen Matrixzerlegungen wie etwa LU-, Cholesky-, QR-, SVD- und Schur-Zerlegung sind in LAPACK ebenso enthalten.

Wie bereits in Kap. 1.5.1 gezeigt wurde, sind die Eigenwerte des Spannungstensors die Hauptspannungen und die Eigenvektoren die Hauptspannungsrichtungen. Wird der Spannungstensor in einem Koordinatensystem angeschrieben, das mit den Hauptspannungsrichtungen zusammenfällt, fallen die Schubspannungskomponenten des Spannungstensors weg, man spricht von der Darstellung in Hauptnormalform. Im Folgenden ist bei der Implementierung der Hauptnormalspannungen immer auch implizit die Implementierung der Hauptnormalverzerrungen gemeint. Diese geht vom Verzerrungstensor  $[\varepsilon]$  aus, der Berechnungsweg bleibt aber gleich.

### 2.4.1 Auswahl des LAPACK-Drivers

Die Auswahl des richtigen Solvers in LAPACK erforderte einige Überlegungen. Alle LAPACK-Treiber und Routinen sind nach einem fixen Schema der Form `XYZZZZ` benannt. Dieses lässt sich wie folgt entschlüsseln [16]:

- Der erste Buchstabe, X, gibt den Mengenbereich und die Maschinenpräzision des zu lösenden Problems an.
  - S reelle Zahlen, einfache Präzision
  - D reelle Zahlen, doppelte Präzision
  - C komplexe Zahlen, einfache Präzision
  - Z komplexe Zahlen, doppelte Präzision
- Die nächsten zwei Buchstaben, YY, geben die Eigenschaften der Matrix wieder. Beispielsweise steht das Kürzel SY für symmetrische Matrizen.
- Die Buchstaben ZZZ stehen für das zu lösende Problem, wobei Routinen zur Lösung des Eigenwertproblems das Kürzel EV tragen.

Eine für das vorliegende Problem geeignete Routine ist somit die LAPACK-Routine `DSYEV`.

Jeder LAPACK-Treiber ruft wiederum die für das Problem nötigen Routinen auf, die aus der Numerik heraus erforderlich sind.

- DSYTRD tridiagonalisiert die Matrix mithilfe von Ähnlichkeitstransformationen. Damit bleiben die Eigenwerte der Matrix erhalten, siehe Kap. 1.8.2
- DORGTR erstellt die orthogonale Reflektormatrix  $[Q]$  aus Gl. (1.62), die zur Berechnung der Eigenvektoren benötigt wird.
- DSTEQR berechnet alle Eigenwerte und Eigenvektoren der Matrix mithilfe des Shifted QR-Algorithmus aus Kap. 1.8.5.

DSYEV gibt die Eigenwerte und die normalisierten Eigenvektoren zurück. Das Eigenwertproblem bringt es mit sich, dass die Länge der Eigenvektoren grundsätzlich nicht bestimmt ist - auch bei einer Normalisierung auf Länge 1 gibt es zwei entgegen gerichtete richtige Darstellungsmöglichkeiten pro Eigenvektor. Dies gilt es bei der Interpretation der Ergebnisse zu beachten, siehe Kap. 3.

Die Coefficient Function Eigen gibt bei Aufruf der vererbten Funktion GetVector() ein generisches Objekt zurück, welches die normalisierten Eigenvektoren multipliziert mit den Eigenwerten und die Eigenwerte allein enthält. Dieses Array-Objekt hat physikalisch keinen Sinn und muss später weiter aufbereitet werden, siehe Codeausschnitt 2.1.

Codeausschnitt 2.1: Implementierung des Eigenwertlösers

```

1 //lp_a: Spannungs- oder Verzerrungstensor

3 dsyev(&lp_jobz, &lp_uplo, &lp_n, &*lp_a.begin(), &lp_lda, &*lp_w.begin()
      , &*lp_work.begin(), &lp_lwork, &lp_info);

5 //lp_w: Eigenwerte
  //lp_a: normierte Eigenvektoren
7 //lp_n: Dimension des Problems

9 //Konvertierung in ein generisches Array
  for(int i = 0; i < lp_n; ++i) {
11   for (int j = 0; j < lp_n; ++j){
      solVec[i*lp_n + j] = lp_w[i] * lp_a[i*lp_n + j];
13   }
  }
15 for(int i = 0; i < lp_n; ++i) {
      solVec[lp_n * lp_n + i] = lp_w[i];
17 }

```

## 2.5 CoefFunctionCache

Wie in Abb. 2.1 gezeigt, erfolgt nach der Berechnung der Eigenwerte und Eigenvektoren, die gesammelt übergeben werden, in den neu definierten Ergebnisobjekten (*results*) *mech\_principal\_stress\_min\_vector* (kleinste Hauptnormalspannung), *mech\_principal\_stress\_med\_vector* (mittlere Hauptnormalspannung) und *mech\_principal\_stress\_max\_vector* (größte Hauptnormalspannung) bzw. deren skalaren Varianten eine Zuordnung der Ergebnisse zum richtigen Ergebnistyp. Dies geschieht durch

den in CFS++ integrierten Math-Parser in der Coefficient Function Compound, welcher einfache Rechenoperationen beherrscht und im vorliegenden Fall nur jene Komponenten aus dem generischen Lösungsarray übernimmt, welche auf den angeforderten Lösungstyp passen.

Aufgrund der Programmarchitektur ist es nicht möglich, beim Aufruf eines Ergebnistyps auf die Ergebnisse eines anderen Ergebnistyps zuzugreifen. Dies ist im vorliegenden Fall insofern ein Nachteil, da beim Aufruf einer Hauptnormalspannung erst das Eigenwertproblem gelöst werden muss, was vorher wiederum die Berechnung des Spannungstensors erfordert. Somit werden beim Aufruf aller Hauptnormalspannungen diese Probleme wiederholt gelöst, was zu einem unnötigen Rechenaufwand führt.

Aus diesem Grund wurde eine *Coefficient Function Cache* eingeführt, die bereits ausgewertete Resultate speichern kann. Abb. 2.2 zeigt ein Schema der Implementierung dieser. Es soll angedeutet werden, dass das Objekt des Datentyps Coefficient Function Cache im vorliegenden Fall das Objekt des Datentyps Coefficient Function Eigen ersetzen soll. Die Coefficient Function Cache wird mit der zu zwischenspeichernden Coefficient Function als Parameter in der MechPDE instanziiert.

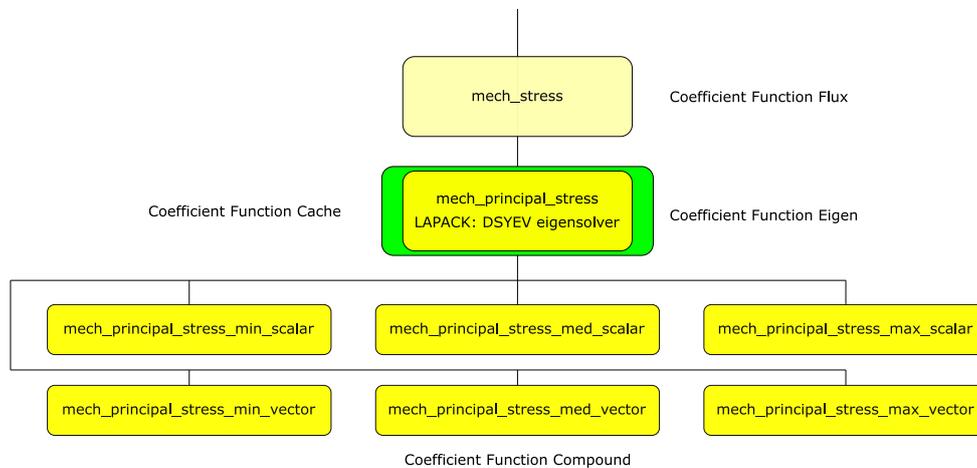


Abbildung 2.2: Schema zur Implementierung der CoefFunctionCache

Mittels einer Abfrage wird ermittelt, ob im betreffenden Element und an den lokalen Elementkoordinaten bereits Ergebnisse ausgewertet wurden. Wenn nicht, werden diese in einem geeigneten Objekt abgelegt. Sollten diese bereits vorhanden sein, werden die Ergebnisse nicht neu ausgewertet, sondern direkt aus dem Cacheobjekt übergeben.

## 2.5.1 Datentyp des Cache

Auf die Wahl des optimalen Datentyps zur Ablage der ausgewerteten Ergebnisse wurde Wert gelegt. Dazu wurden einige Benchmark-Tests mit verschiedenen Konfigurationen der Implementierung durchgeführt. Als Kandidaten zur Speicherung der bereits ausgewerteten Ergebnisse wurden Objekte der Datentypen `std::map` und `boost::unordered_map` ausgewählt.

### Maps

Maps sind assoziative Container, dessen Elemente aus einem Schlüssel (key) und einem dazugehörigen Wert (mapped value) bestehen [17]. Beide können dabei grundsätzlich von beliebigem Datentyp sein, der Schlüssel muss jedoch, mit Ausnahme von Sonderformen von Maps, eindeutig sein [17].

## **std::map**

Eine `std::map` ist eine Sequenz von `std::pair` - Elementen, wobei der erste Wert des Wertepaares immer den Schlüssel darstellt und das zweite Element den zum Schlüssel gehörigen Wert [17]. Die Elemente werden dabei in einer nach dem Schlüssel geordneten Reihenfolge gespeichert [17]. In der vorliegenden Implementierung ist der Schlüssel die globale Elementnummer und ist als Integer implementiert, somit kann der bereits implementierte Iterator problemlos verwendet werden. Die Kosten für das Schreiben und Suchen eines Wertepaares liegen bei  $\mathcal{O}(\log N)$ , wobei  $N$  die Länge der Map beschreibt [17].

## **boost::unordered\_map**

Eine Alternative zur `std`-Bibliothek stellt die C++ Bibliothek `boost` [18] dar, die bereits in CFS++ in vielen Implementierungen verwendet wird. Der Container `unordered_maps` basiert auf einem Hashverfahren und sortiert die Schlüssel im Container nicht. Durch eine geeignete Hashfunktion, die auf den Schlüssel angewandt wird, wird die Position des dazugehörigen Wertes in der Hashtabelle bestimmt [17]. Damit entfällt der Iterationsschritt beim Suchen eines Wertes - beim erneuten Aufrufen des Schlüssels wird durch die Berechnung des Hashwertes sofort die richtige Position in der Hashtabelle gefunden [17]. Die Kosten des Einfügens und Suchens in der Tabelle hängen von der Qualität der Hashfunktion ab - im Idealfall sind diese konstant, auch bei großen Containern. Für Schlüssel von primitiven Datentypen trifft dies auch im Allgemeinen zu. Der Speicherbedarf kann jedoch höher sein als bei sortierten Containern [19].

## **2.5.2 Benchmarks**

Da die tatsächliche Performance einer Map von recht vielen Faktoren abhängt, wurde als Benchmark in CFS++ ein Kragbalken unter mechanischer Punktlast am äußeren Ende des Balkens mit verschiedenen Elementzahlen simuliert. Die im Folgenden genannten Zahlenwerte sollten dabei nur relativ zueinander betrachtet werden. Die Tests wurden in folgender Konfiguration ausgeführt:

- Kragbalken, 2D-Problem im ebenen Spannungszustand mit verschiedener Elementanzahl
- linear statisches Problem
- Debug-Modus in CFS++
- Berechnungen mit einem Thread auf einem Intel i5-Prozessor
- Berechnete Ergebnisse: Knotenverschiebung, mechanische Spannung, kleinste Hauptnormalspannung, größte Hauptnormalspannung

In einem ersten Test wurde die Performance von `std::map` und `boost::unordered_map` festgestellt. Dabei wurde die Simulation mit verschiedenen Elementzahlen durchgeführt und die gesamte Rechenzeit des Simulationslaufs gemessen. Es wurden die Ergebnisse der Coefficient Function Eigen, also die berechneten Hauptnormalspannungen mithilfe der Coefficient Function Cache zwischengespeichert, jedoch nicht die der Coefficient Function Flux, welche den Spannungstensor berechnet.

Tabelle 2.1: Messwerte des Benchmarks `std::map` mit `boost::unordered map`

Datentyp	Elementanzahl			
	2000	20000	50000	100000
<code>std::map</code>	2.35	16.36	40.97	81.73
<code>boost::unordered map</code>	2.53	16.42	39.87	80.74

Wie zu erwarten war, ist bei einer kleinen Elementanzahl die Performance bei der Implementierung von `std::map` etwas schneller. Bei steigender Elementanzahl, also bei mehr als 20000 Elementen, wird der Vorteil einer Schreib- und Lesezeit der `boost::unordered_map` in der Ordnung  $\mathcal{O}(const)$  gegenüber einer Schreib- und Lesezeit der `std::map` in der Ordnung  $\mathcal{O}(\log N)$  ersichtlich. Daher wurde für die Implementierung zum Cachen einer ausgewerteten Coefficient Function ein Container des Datentyps `boost::unordered_map` verwendet.

In einem zweiten Benchmark wurde ermittelt, welche Performanceunterschiede durch das Cachen verschiedener Coefficient Functions auftreten. Bei der Berechnung der Hauptnormalspannungen wurden wechselweise der Spannungstensor, berechnet durch die Coefficient Function Flux und die Hauptspannungen, berechnet durch die Coefficient Function Eigen, zwischengespeichert. Die Ergebnisse sind in Tab. 2.2 ersichtlich, wobei wiederum die CPU time der gesamten Simulation gemessen wurde. Die angeforderten Ergebnistypen entsprechen jenen aus dem ersten Benchmark.

- Variante 1: Kein Cachen
- Variante 2: Cachen der `CoeffFunctionFlux`
- Variante 3: Cachen der `CoeffFunctionEigen`
- Variante 4: Cachen der `CoeffFunctionEigen` und der `CoeffFunctionFlux`

Variante 3 entspricht somit der getesteten Variante aus Benchmark 1.

Tabelle 2.2: Benchmark zum Cache verschiedener Coefficient Functions

Datentyp	Elementanzahl			
	2000	20000	50000	100000
Variante 1	2.54	16.86	41.69	83.65
Variante 2	2.25	16.44	39.98	79.70
Variante 3	2.53	16.42	39.87	80.74
Variante 4	2.31	16.63	39.72	80.03

Aus Tab. 2.2 ist ersichtlich, dass die implementierte Variante gänzlich ohne Cache-Funktion die schlechteste Performance aufweist. Alle weiteren Implementierungsvarianten weisen in etwa die gleiche Performance auf. Es wurde deshalb bei der Implementierung Variante 3 gewählt, also ein Cachen der berechneten Hauptnormalspannungen, während auf ein Cachen des Spannungstensors verzichtet wurde. Es ist außerdem zu erwarten, dass im Einsatz von CFS++ die Ergebnisse der Hauptspannungsberechnung weniger oft angefordert werden als die Ergebnisse der Spannungsberechnung, wodurch eine Implementierung, in der die Ergebnisse der Berechnung des Spannungstensors immer zwischengespeichert werden, nicht sinnvoll erscheint. Zur Verdeutlichung ist ein Teil der Implementierung im Codeabschnitt 2.2 wiedergegeben.

## Codeausschnitt 2.2: Teile der Implementierung der Coefficient Function Cache

```
1  template<class TYPE> void CoefFunctionCache<TYPE>::GetVector(Vector<TYPE
    >& coefVec, const LocPointMapped& lpm) {
    //Iterator für key-Werte
3  it_vec_ = cached_data_vec_.find(lpm.ptEl->elemNum);

5  if (it_vec_ != cached_data_vec_.end()) {
    //Elementnummer in Cache gefunden
7  coefVec = cached_data_vec_[lpm.ptEl->elemNum].second;

9  //Check, ob das Element auch im gleichen Punkt im lokalen
    Koordinatensystem ausgewertet wurde
    Vector<Double> actlp = lpm.lp.coord;
11  Double dist = actlp.NormL2(cached_data_vec_[lpm.ptEl->elemNum].first);
    if (dist > 0.001) {
13      WARN("Cached Results and requested results of GetVector do not have
        the same local point. Local distance: " << dist);
    }
15 }
    else {
17  //Elementnummer im Cache nicht gefunden. Auswerten der CoefFunction
        und Einfügen in Cache.
        cached_coefFct_->GetVector(coefVec, lpm);
19  //Struktur der Map
        //Key: Elementnummer
21  //Value: std::pair (Wertepaar) bestehend aus
        // - lokale Koordinaten
23  // - ausgewertetes Resultat
        cached_data_vec_.insert(std::make_pair(lpm.ptEl->elemNum, std::
            make_pair(lpm.lp.coord, coefVec)));
25 }
    }
```

# Kapitel 3

## Testfälle

Die Implementierung der Hauptnormalspannungen und Hauptnormalverzerrungen, dargelegt in Kap. 2, wurde anhand einiger Testbeispiele verifiziert. Essentiell für die Auswahl der Testbeispiele ist die Zugänglichkeit einer analytischen Berechnung der Hauptnormalspannungen bzw. -verzerrungen. Dies ist am einfachsten möglich, wenn auch der Spannungs- oder Verzerrungstensor für das Problem analytisch angegeben werden kann. Die Berechnung der Hauptnormalspannungen und -verzerrungen ist daraufhin, wie in Kap. 1.5.1 dargestellt, möglich.

Sollten in den folgenden Testbeispielen keine Einheiten für Geometrie-, Material- und Lastdaten angegeben worden sein, sind diese für eine konsistente Angabe als SI-Einheiten bzw. deren direkt davon abgeleiteten Einheiten zu verstehen.

### 3.1 Einachsiger Schub

#### 3.1.1 Analytische Lösung

In einem ersten Test wurde eine isotrope Scheibe so belastet, dass sich in ihr ein konstanter Schubspannungszustand einstellt. Dies ist möglich durch Aufbringen von Schubkräften am Element, Abb. 3.1(a). Für eine nachfolgende Formulierung in der FE-Methode müssen Starrkörperrotationen ausgeschlossen werden. Diese wurden nach Abb. 3.1(b) durch eine geeignete Beschränkung der Freiheitsgrade des Systems verhindert.

Der sich in der Scheibe einstellende Spannungszustand lautet

$$[\boldsymbol{\sigma}] = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \quad (3.1)$$

Um die Eigenwerte des Spannungstensors aus Gl. (3.1) zu erhalten, ist die charakteristische Gleichung des Spannungstensors aufzustellen und gleich Null zu setzen und lautet daher

$$\det([\boldsymbol{\sigma}] - \lambda[\mathbf{I}]) = (\sigma_{xx} - \lambda)(\sigma_{yy} - \lambda) - \sigma_{xy}^2 = 0. \quad (3.2)$$

Aus Gl. (3.2) lassen sich die Eigenwerte leicht zu  $\lambda = \pm\sigma_{xy}$  bestimmen. Die Hauptnormalspannungen sind somit

$$\sigma_1 = \lambda_1 = 1, \quad \sigma_2 = \lambda_2 = -1. \quad (3.3)$$

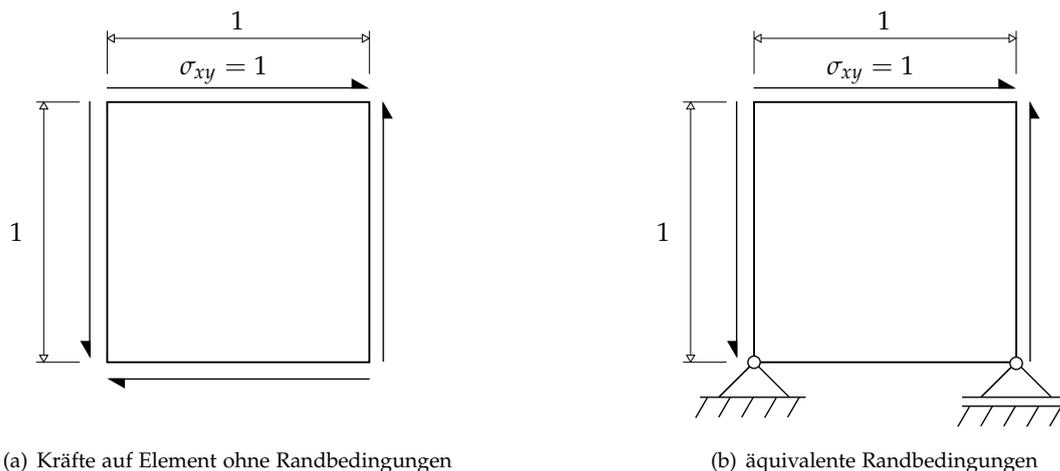


Abbildung 3.1: 2D Element

Die dazugehörigen Eigenvektoren lassen sich aus der Eigenwertgleichung berechnen, also

$$([\boldsymbol{\sigma}] - \lambda_i [\mathbf{I}]) \mathbf{n}_i = \mathbf{0}, \quad (3.4)$$

woraus sich

$$\mathbf{n}_1 = \alpha_1 \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mathbf{n}_2 = \alpha_2 \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad (3.5)$$

mit  $\alpha_i$  beliebig errechnet. Es wird  $\alpha_i$  jedoch so gewählt, dass  $\|\mathbf{n}_i\|_2 = 1$  ist, also

$$\mathbf{n}_1 = \pm \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mathbf{n}_2 = \pm \frac{1}{\sqrt{2}} \begin{bmatrix} -1 \\ 1 \end{bmatrix} \quad (3.6)$$

### 3.1.2 Lösung in CFS++

Da die Scheibe einer konstanten reinen Schubbeanspruchung ausgesetzt ist, kann aus dem Hooke'schen Gesetz Gl. (1.35) geschlossen werden, dass auch der Verzerrungszustand in der Scheibe mit einer Schubverzerrung konstant ist. Die Scheibe kann daher als einzelnes Element (single element) modelliert werden, da diese die Geometrieänderung vollständig abbildet. Die errechneten Hauptnormalspannungen des Finite-Elemente-Programms CFS++ lassen sich am anschaulichsten als Vektorplot darstellen, wobei

$$\boldsymbol{\sigma}_i = \sigma_i \mathbf{n}_i \quad (3.7)$$

geschrieben werden kann.

Die Modellierung erfolgte als linear statisches 2D Modell im ebenen Spannungszustand. Die aufgebrachten Kräfte wurden als Spannungen (traction) modelliert, wobei eine Umrechnung auf konsistente Knotenkräfte im Sinne der virtuellen Arbeit in CFS++ passiert.

Das Ergebnis der Berechnung ist in Abb. 3.2 ersichtlich, wobei alle Plots aus dem Postprocessing-Tool Paraview stammen. Die Ergebnisse aus der Implementierung in CFS++ und der analytischen Rechnung in Kap. 3.1.2 stimmen sowohl in Richtung und Betrag genau überein.

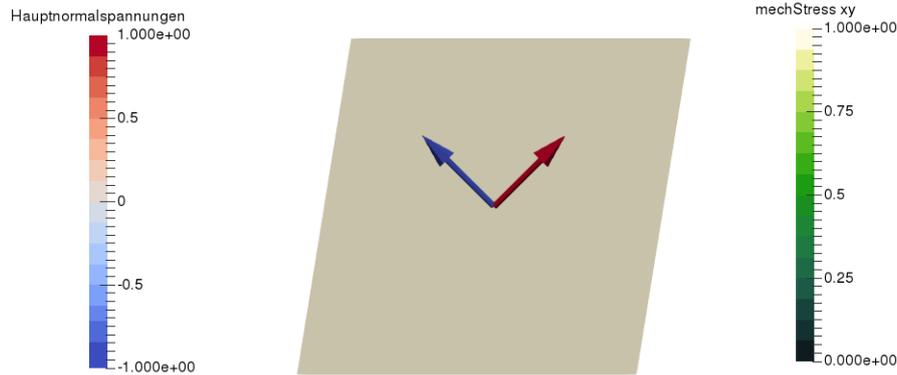


Abbildung 3.2: Vektorplot Hauptnormalspannungen Single Element 2D

## 3.2 Würfel unter Zugspannung

### 3.2.1 Analytische Lösung

In einem weiteren Testfall wurde das Verhalten der Implementierung an einem dreidimensionalen Würfel getestet. Dieser wurde einer uniaxialen Zugspannung unterworfen. Der im gesamten Körper konstante Spannungstensor lautet

$$[\boldsymbol{\sigma}] = \begin{bmatrix} \sigma_{xx} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \quad (3.8)$$

Aus dem Hooke'schen Gesetz für linear elastisches Material aus Gl. (1.32) kann der Verzerrungszustand im Körper berechnet werden. Es gilt also

$$\varepsilon_{xx} = \frac{\sigma_{xx}}{E}, \quad \varepsilon_{yy} = \varepsilon_{zz} = -\nu \frac{\sigma_{xx}}{E}. \quad (3.9)$$

Mit den Zahlenwerten  $\sigma_{xx} = 1$ ,  $E = 10^6$ ,  $\nu = 0.2$  ergibt sich der konstante Verzerrungstensor im Körper von

$$[\boldsymbol{\varepsilon}] = \frac{1}{10^6} \begin{bmatrix} 1 & 0 & 0 \\ 0 & -0.2 & 0 \\ 0 & 0 & -0.2 \end{bmatrix}. \quad (3.10)$$

Die Nullstellen des charakteristischen Polynoms  $\det([\boldsymbol{\varepsilon}] - \lambda[\mathbf{I}])$  ergeben die Eigenwerte

$$\lambda_1 = \varepsilon_1 = 10^{-6}, \quad \lambda_2 = \lambda_3 = \varepsilon_2 = \varepsilon_3 = -2 \cdot 10^{-7}. \quad (3.11)$$

Der normalisierte Eigenvektor zum ersten Eigenwert lautet

$$\mathbf{n}_1 = \pm \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T. \quad (3.12)$$

Da der zweite Eigenwert die algebraische Vielfachheit 2 besitzt, lässt sich dessen geometrische Richtung nicht mehr eindeutig angeben und jeder der in Gl. (3.13) angegebenen Eigenvektoren ist korrekt. Allerdings müssen die Eigenvektoren orthogonal bleiben,  $\alpha$  kann daher nicht für jeden Eigenvektor

einzel frei gewählt werden, es gilt also

$$\mathbf{n}_2 = \begin{bmatrix} 0 \\ \cos \alpha \\ \sin \alpha \end{bmatrix}, \quad \mathbf{n}_3 = \begin{bmatrix} 0 \\ -\sin \alpha \\ \cos \alpha \end{bmatrix}, \quad (3.13)$$

mit  $\alpha$  beliebig.

### 3.2.2 Lösung in CFS++

Wie schon im Testfall Kap. 3.1, kann auch diese Geometrie in einem Element abgebildet werden, da dieses die Verschiebungen und Verzerrungen vollständig modelliert. Die Analyse in CFS++ wurde an einem mechanischen linear statischen Modell in 3D-Formulierung durchgeführt, wobei das Mesh aus einem 3D-Element mit der Kantenlänge 1 besteht. Die Randbedingungen wurden so gesetzt, dass die Querkontraktion durch die angelegte Zugspannung nicht behindert ist.

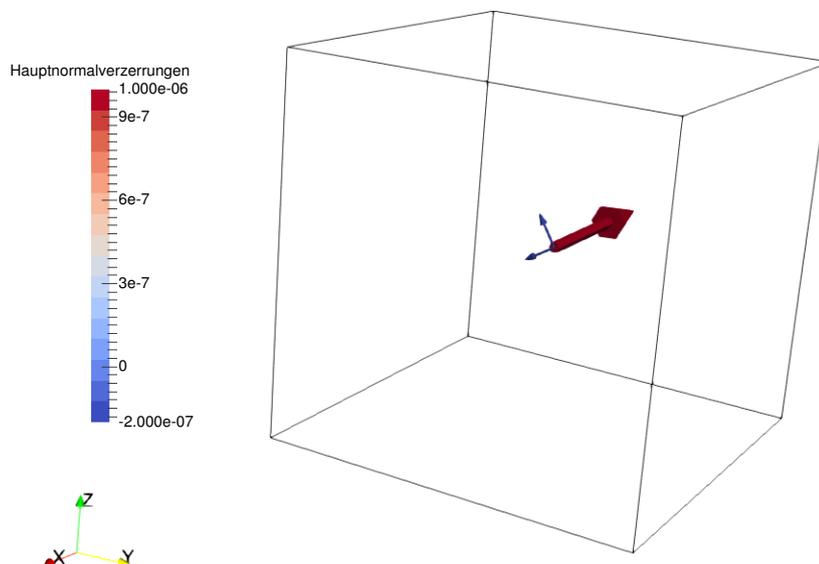


Abbildung 3.3: Vektorplot Hauptnormalverzerrungen Single Element 3D

Abbildung 3.3 zeigt den Vektorplot der Hauptnormalverzerrungen aus CFS++. Wie erwartet, sind die Ergebnisse ident mit der analytischen Lösung. Hier zeigt sich auch der Vorteil eines bereit vorimplementierten Eigenwertlösers, der etwaige Spezialfälle wie doppelte Eigenwerte etc. reibungslos behandelt. Ein selbst programmierter, spezialisierter Löser für kleine Matrizen wäre vielleicht schneller, allerdings wäre es schwierig an die Qualität der bereits vorhandenen Löser hinsichtlich Ausnahmenbehandlung, Behandlung mehrfacher Eigenwerte etc. heranzukommen.

### 3.3 Rohr unter Innendruck

In einem weiteren Testfall wurde eine Spannungsanalyse eines Rohres unter Innendruck durchgeführt und dessen Hauptspannungen betrachtet. Dazu wurde wiederum zuerst die analytische Lösung des Spannungszustandes hergeleitet. Dies ist auf mehreren Wegen möglich, hier wurde die Airy'sche Spannungsfunktion gewählt.

### 3.3.1 Airy'sche Spannungsfunktion

Die folgenden Ausführungen zur Airy'schen Spannungsfunktion stammen aus [20]. Diese stellt für ebene Flächenträger, die nur durch Kräfte in ihrer Ebene belastet werden und deren Mittelfläche bei der Formänderungen eben bleibt (sogenannte Scheiben), eine Möglichkeit der analytischen Berechnung des Spannungs- und Verschiebungsfeldes dar. Die Anwendung dieser ist auf statische, lineare Probleme mit isotropen Materialien, die dem Hooke'schen Gesetz gehorchen, begrenzt.

In Gl. (1.37) ist der Zusammenhang

$$\frac{\partial^2 \varepsilon_{xy}}{\partial x \partial y} = \frac{\partial^2 \varepsilon_{xx}}{\partial^2 y} + \frac{\partial^2 \varepsilon_{yy}}{\partial^2 x} \quad (3.14)$$

als Kompatibilitätsbedingung eingeführt worden. Setzt man in diese für die Verzerrungskomponenten die Beziehungen nach dem Hooke'schen Gesetz in der Formulierung für den ebenen Spannungszustand Gl. (1.35) ein, ergibt sich

$$2(1 + \nu) \frac{\partial^2 \sigma_{xy}}{\partial x \partial y} = \frac{\partial^2 \sigma_{xx}}{\partial y^2} - \nu \frac{\partial^2 \sigma_{yy}}{\partial y^2} + \frac{\partial^2 \sigma_{yy}}{\partial x^2} - \nu \frac{\partial^2 \sigma_{xx}}{\partial x^2}. \quad (3.15)$$

Die Airy'sche Spannungsfunktion ist über die partiellen Ableitungen

$$\sigma_{xx} = \frac{\partial^2 \Phi}{\partial y^2}, \quad \sigma_{yy} = \frac{\partial^2 \Phi}{\partial x^2}, \quad \sigma_{xy} = -\frac{\partial^2 \Phi}{\partial x \partial y} \quad (3.16)$$

definiert. Mit Gl. (3.16) in Gl. (3.15) eingesetzt ergibt sich

$$\frac{\partial^4 \Phi}{\partial x^4} + 2 \frac{\partial^4 \Phi}{\partial x^2 \partial y^2} + \frac{\partial^4 \Phi}{\partial y^4} = 0. \quad (3.17)$$

Diese lineare partielle Differentialgleichung 4. Ordnung wird *Scheibengleichung* genannt. Mit dem Laplace-Operator

$$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad (3.18)$$

kann Gl. (3.17) auch in der Form

$$\Delta \Delta \Phi = 0 \quad (3.19)$$

geschrieben werden. Mit Gl. (3.16) folgt aus Gl. (3.19) auch

$$\Delta(\sigma_{xx} + \sigma_{yy}) = 0. \quad (3.20)$$

Die Spannungssumme  $\sigma_{xx} + \sigma_{yy}$  stellt somit eine harmonische Funktion von  $x$  und  $y$  dar, wobei keine Materialkennwerte in der Gleichung auftreten. Die Beziehungen Gl. (3.19) und (3.20) können auch bei Volumskräften und Wärmedehnungen konstanter Größe hergeleitet werden und führen zum selben Ergebnis. Auch für den ebenen Verzerrungszustand lässt sich Gl. (3.20) herleiten, dazu muss in Gl. (3.15) anstatt der Beziehung für den ebenen Spannungszustand jener für den ebenen Verzerrungszustand eingesetzt werden.

Oftmals ist die Formulierung des Problems in Polarkoordinaten vorteilhaft. Dabei bestehen zwischen den Polarkoordinaten  $r, \varphi$  und den kartesischen Koordinaten  $x, y$  die Zusammenhänge

$$x = r \cos \varphi, \quad y = r \sin \varphi. \quad (3.21)$$

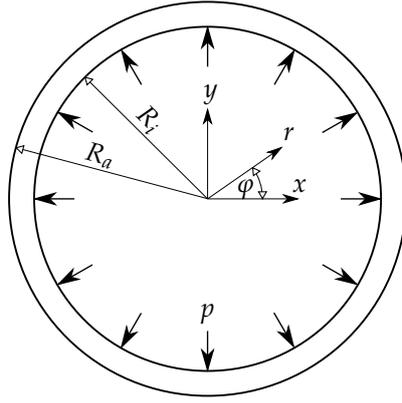


Abbildung 3.4: Rohr unter Innendruck

Die Scheibengleichung transformiert dadurch nach längerer Rechnung zu

$$\Delta\Delta\Phi = 0 \quad \text{mit} \quad \Delta = \frac{\partial^2}{\partial r^2} + \frac{1}{r^2} \frac{\partial^2}{\partial \varphi^2} + \frac{1}{r} \frac{\partial}{\partial r}, \quad (3.22)$$

wobei die Zusammenhänge zwischen den Spannungskomponenten in Polarkoordinaten und der Airy'schen Spannungsfunktion

$$\sigma_{rr} = \frac{1}{r^2} \frac{\partial^2 \Phi}{\partial \varphi^2} + \frac{1}{r} \frac{\partial \Phi}{\partial r}, \quad \sigma_{\varphi\varphi} = \frac{\partial^2 \Phi}{\partial r^2}, \quad \sigma_{r\varphi} = -\frac{\partial}{\partial r} \left( \frac{1}{r} \frac{\partial \Phi}{\partial \varphi} \right) \quad (3.23)$$

lauten.

Mit der Einführung der Airy'schen Spannungsfunktion ist es möglich, die Gleichgewichtsbedingungen, die geometrischen Bedingungen und die Spannungs-Dehnungs-Beziehung in einer einzigen Differentialgleichung  $\Delta\Delta\Phi = 0$  zusammenzufassen, in der nur die Spannungsfunktion  $\Phi$  als Unbekannte auftritt. Gelingt es, diese Differentialgleichung unter Berücksichtigung der Randbedingungen zu integrieren, gewinnt man damit die strenge Lösung der betreffenden Scheibenaufgabe. Die Differentialgleichung  $\Delta\Delta\Phi = 0$  besitzt unendlich viele Lösungen, es ist jedoch nicht trivial eine Lösung zu finden, die auch den jeweils geltenden Randbedingungen genügt.

### 3.3.2 Analytische Lösung

Für die analytische Lösung des Beispiels wurden Anleihen aus [20] entnommen. An einem Rohr greift ein Druck  $p$  am Innenrand an, Abb. 3.4. Diese Belastung ruft einen drehsymmetrischen Spannungszustand hervor, da sowohl die Geometrie als auch die Belastung drehsymmetrisch sind. Die Normalspannungen an den Rändern lassen sich somit als

$$\sigma_{rr}(r = R_i) = -p, \quad \sigma_{rr}(r = R_a) = 0 \quad (3.24)$$

angeben. Da der Spannungszustand vom Drehwinkel unabhängig ist, verschwinden die partiellen Ableitungen der Spannungsfunktion nach  $\varphi$ , die Scheibengleichung (3.22) nimmt die Form

$$\Delta\Delta\Phi = \left( \frac{d^2 \Phi}{dr^2} + \frac{1}{r} \frac{d\Phi}{dr} \right) \cdot \left( \frac{d^2 \Phi}{dr^2} + \frac{1}{r} \frac{d\Phi}{dr} \right) = 0 \quad (3.25)$$

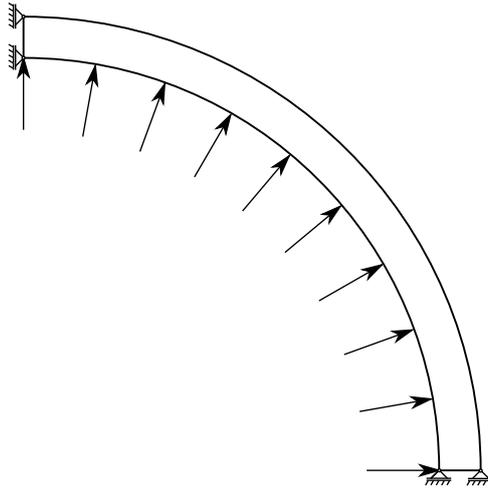


Abbildung 3.5: Geometrie des Testfalles Rohr im FE-Modell

an. Die Komponenten des Spannungstensors lassen sich also zu

$$\sigma_{rr} = \frac{1}{r} \frac{d\Phi}{dr}, \quad \sigma_{\varphi\varphi} = \frac{d^2\Phi}{dr^2}, \quad \sigma_{r\varphi} = 0 \quad (3.26)$$

vereinfachen. Die allgemeine Lösung der Gl. (3.25) lautet

$$\Phi(r) = c_0 + c_1 \ln r + c_2 r^2 + c_3 r^2 \ln r. \quad (3.27)$$

Nach Einsetzen der Randbedingungen errechnet sich schließlich der Spannungszustand im Rohr zu

$$\sigma_{rr} = -\frac{p}{r^2} \frac{R_i^2 R_a^2}{R_a^2 - R_i^2} + p \frac{R_i^2}{R_a^2 - R_i^2}, \quad \sigma_{\varphi\varphi} = \frac{p}{r^2} \frac{R_i^2 R_a^2}{R_a^2 - R_i^2} + p \frac{R_i^2}{R_a^2 - R_i^2}. \quad (3.28)$$

Für dünnwandige Rohre, also  $R_i \approx R_a$ , lässt sich die rechte Gleichung in (3.28) vereinfachen. Es gilt dann

$$\sigma_{\varphi\varphi} = \frac{p}{2} \frac{R_a + R_i}{R_a - R_i}, \quad (3.29)$$

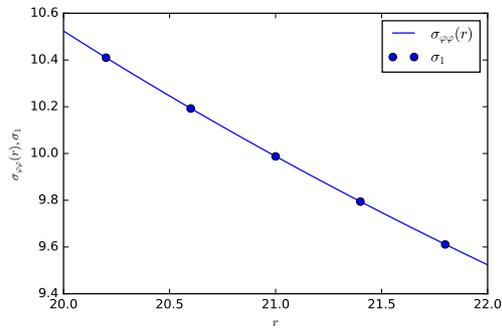
welche auch als Kesselformel bekannt ist.

Da die Schubspannungskomponente  $\sigma_{r\varphi}$  sich zu 0 berechnet, sind die Spannungskomponenten  $\sigma_{rr}$  und  $\sigma_{\varphi\varphi}$  auch gleichzeitig Hauptspannungen.

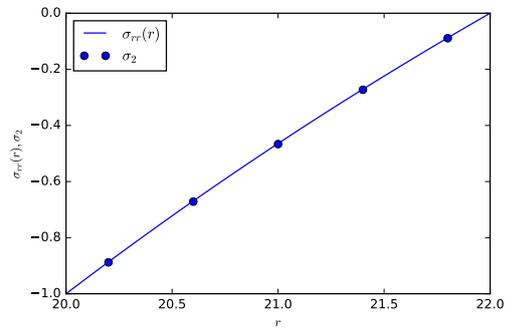
### 3.3.3 Lösung in CFS++

Die FE-Modellierung erfolgte als linear statisches 2D-Modell in der Formulierung für den ebenen Verzerrungszustand, wobei nur ein Viertel des Rohrquerschnittes modelliert wurde, da sowohl Geometrie als auch Belastung um die vertikale und die horizontale Achse symmetrisch sind, siehe Abb. 3.5.

Die gewählten Geometriedaten sind  $R_i = 20$ ,  $R_a = 22$  sowie die Drucklast  $p = 1$ . Daraus lässt sich das Spannungsfeld nach Gl. (3.28) berechnen. Dieses kann verglichen werden mit den errechneten Ergebnissen in CFS++. Abb. 3.6 zeigt diesen Vergleich. Wie zu erwarten war, entsprechen die Hauptnormalspannungen, welche in CFS++ in den Elementmittelpunkten ausgewertet wurden, genau den analytischen Lösungen aus Gl. (3.28). Abbildung 3.8 zeigt einen Vektorplot der Hauptnormalspannungen, während Abb. 3.7(a) und Abb. 3.7(b) Fringeplots der Hauptnormalspannungen darstellen.

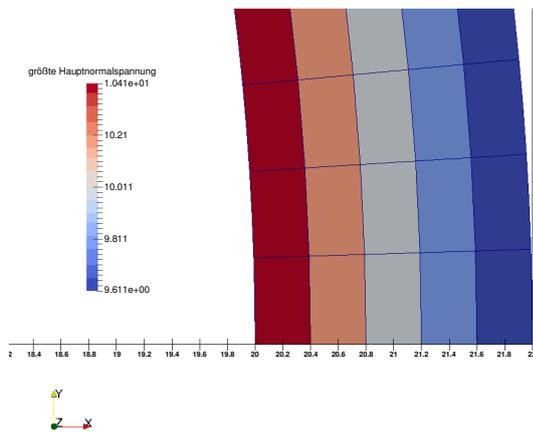


(a) Vergleich  $\sigma_{\varphi\varphi}$  mit  $\sigma_1$

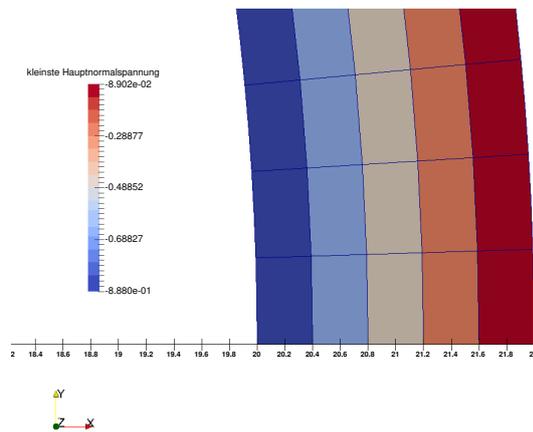


(b) Vergleich  $\sigma_{rr}$  mit  $\sigma_2$

Abbildung 3.6: Vergleich der analytischen Lösung und der FE-Lösung für den Testfall Rohr



(a)  $\sigma_1$



(b)  $\sigma_2$

Abbildung 3.7: Fringeplots Hauptnormalspannungen Rohr unter Innendruck

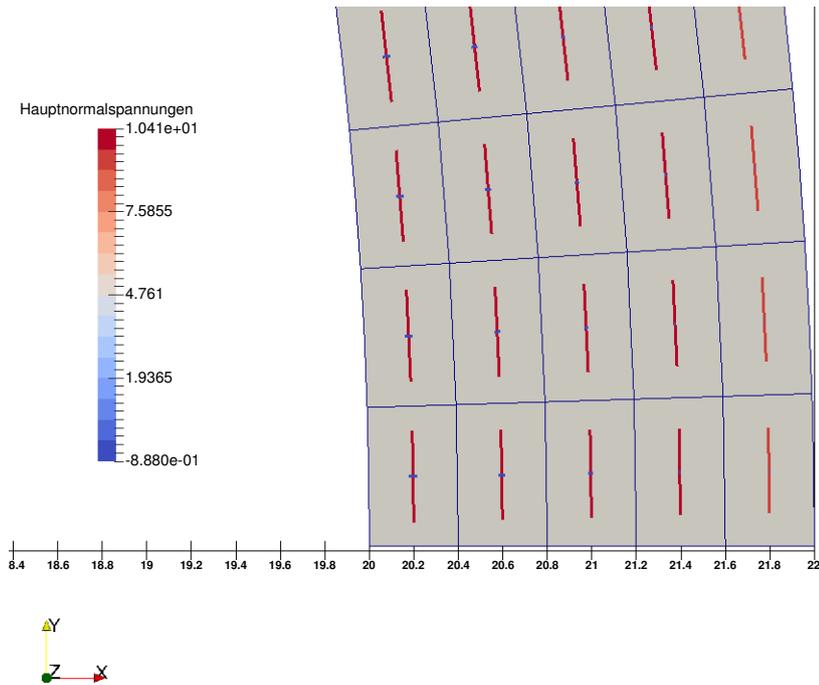


Abbildung 3.8: Vektorplot Hauptspannungen Rohr unter Innendruck. Dargestellt ist ein kleiner Ausschnitt des Rohres nahe der Randbedingung entlang der  $x$ -Achse.

### 3.4 Lochscheibe

Ein klassisches Problem der Festigkeitslehre stellt die unendlich ausgedehnte Scheibe mit einem kreisförmigen Loch mit dem Radius  $a$  dar, die in  $x = \pm\infty$  durch eine konstante Zugspannung  $p$  belastet wird, Abb. 3.9. Durch das Loch tritt eine Störung des ansonsten einachsigen, konstanten Spannungsfeldes  $\sigma_{xx} = p, \sigma_{xy} = \sigma_{yy} = 0$  auf.

#### 3.4.1 Analytische Lösung

Für die analytische Lösung des Beispiels wurde [20] herangezogen. Die Airy'sche Spannungsfunktion für dieses Beispiel lautet

$$\Phi = \frac{p}{4} \left[ r^2 - 2a^2 \ln r - \frac{(r^2 - a^2)^2}{r^2} \cos 2\varphi \right]. \quad (3.30)$$

Gleichung (3.30) erfüllt die Scheibengleichung (3.25). Die Spannungskomponenten lassen sich nun in der Form

$$\begin{aligned} \sigma_{rr} &= \frac{p}{2} \left[ 1 - \frac{a^2}{r^2} + \left( 1 - 4\frac{a^2}{r^2} + 3\frac{a^4}{r^4} \right) \cos 2\varphi \right], \\ \sigma_{\varphi\varphi} &= \frac{p}{2} \left[ 1 + \frac{a^2}{r^2} - \left( 1 + 3\frac{a^4}{r^4} \right) \cos 2\varphi \right], \\ \sigma_{r\varphi} &= \frac{p}{2} \left( -1 - 2\frac{a^2}{r^2} + 3\frac{a^4}{r^4} \right) \sin 2\varphi \end{aligned} \quad (3.31)$$

ausdrücken. Man kann sich leicht davon überzeugen, dass die Spannungsfunktion den Randbedingungen des Problems entspricht. Da im Loch keine äußeren Kräfte angreifen, müssen dort die Spannungskomponenten  $\sigma_{rr}$  und  $\sigma_{r\varphi}$  verschwinden, was für  $r = a$  in Gl. (3.31) auch der Fall ist. Für  $r = \infty$  muss

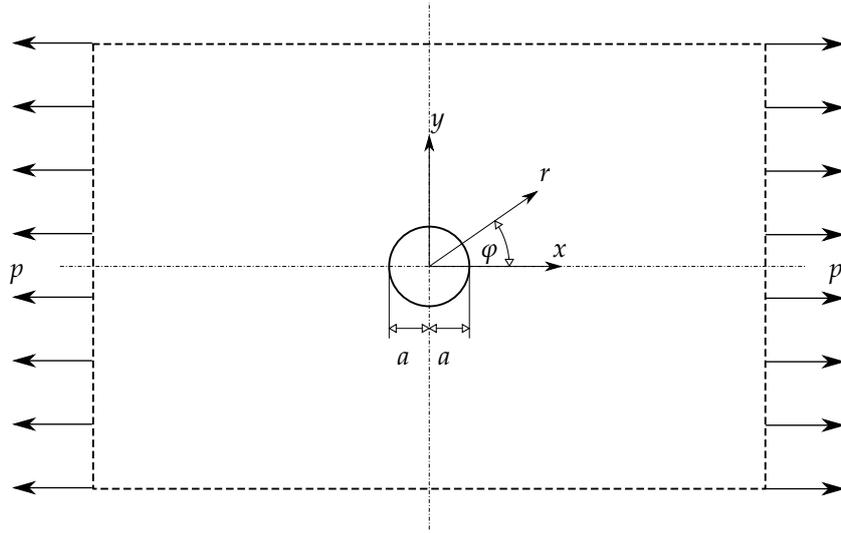


Abbildung 3.9: Geometrie des Testfalles Lochscheibe

$\sigma_{xx} = p, \sigma_{xy} = \sigma_{yy} = 0$  sein. Mit  $r = \infty$  lauten die Gl. (3.31)

$$\begin{aligned}\sigma_{rr}|_{r=\infty} &= \frac{p}{2} (1 + \cos 2\varphi) \\ \sigma_{\varphi\varphi}|_{r=\infty} &= \frac{p}{2} (1 - \cos 2\varphi) \\ \sigma_{r\varphi}|_{r=\infty} &= -\frac{p}{2} \sin 2\varphi.\end{aligned}\quad (3.32)$$

Transformiert man den Spannungstensor  $[\sigma] = \begin{bmatrix} \sigma_{rr} & \sigma_{r\varphi} \\ \sigma_{r\varphi} & \sigma_{\varphi\varphi} \end{bmatrix}$  mit den Spannungskomponenten aus Gl. (3.32) mithilfe der für den zweidimensionalen Fall abgewandelten Gl. (1.49) wieder auf das kartesische Koordinatensystem ist ersichtlich, dass diese Spannungskomponenten tatsächlich der vorausgesetzten einachsigen Zugbeanspruchung entsprechen. Die Drehmatrix lautet in diesem Fall  $[Q] = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{bmatrix}$ .

Die betragsmäßig höchsten Spannungen treten entlang der Bohrung auf. Aus der zweiten Gleichung in Gl. (3.31) erhält man für  $r = a$

$$\sigma_{\varphi\varphi}|_{r=a} = p (1 - 2 \cos 2\varphi), \quad (3.33)$$

welche die Extremwerte  $\sigma_{\varphi\varphi} = -p$  für  $\varphi = 0$  und  $\pi$ , und  $\sigma_{\varphi\varphi} = 3p$  für  $\varphi = \pm \frac{\pi}{2}$  liefert. An den Punkten entlang des Lochkreises ist  $\sigma_{\varphi\varphi}$  auch Hauptspannung, wobei die andere Hauptspannung aufgrund des einachsigen Spannungszustandes 0 ist. In den Punkten, die nicht entlang des Lochkreises liegen, kann diese Annahme wegen  $\sigma_{r\varphi} \neq 0$  nicht getroffen werden.

### 3.4.2 Lösung in CFS++

Abbildung 3.10 zeigt einen Fringeplot der Spannungsfeldes  $\sigma_{xx}$ . Dieser zeigt deutlich den Kerbwirkungsfaktor bei Zugbelastung. Da es in einem FE-Programm nicht ohne spezielle Elemente möglich ist, unendliche Ausdehnungen zu modellieren, wurde eine Scheibe modelliert, deren Seitenlänge im Vergleich zum Lochdurchmesser groß ist, der dadurch induzierte Fehler ist jedoch vernachlässigbar, siehe Abb. 3.11. Für die Simulation in CFS++ wurde ein linear statisches 2D-Modell im ebenen Spannungszustand verwendet. Die gewählten Geometrieparameter nach Abb. 3.11 sind  $a = 1, l = 10$ , die aufgebrauchte Zugspannung beträgt  $p = 1$ . Da sowohl Geometrie als auch Belastung sowohl um die x-Achse als auch um die y-Achse symmetrisch sind, kann eine Analyse an der Viertelscheibe durchgeführt werden. In Abb. 3.12 ist die Lösung des Hauptnormalspannungsproblems aus CFS++ zu sehen.

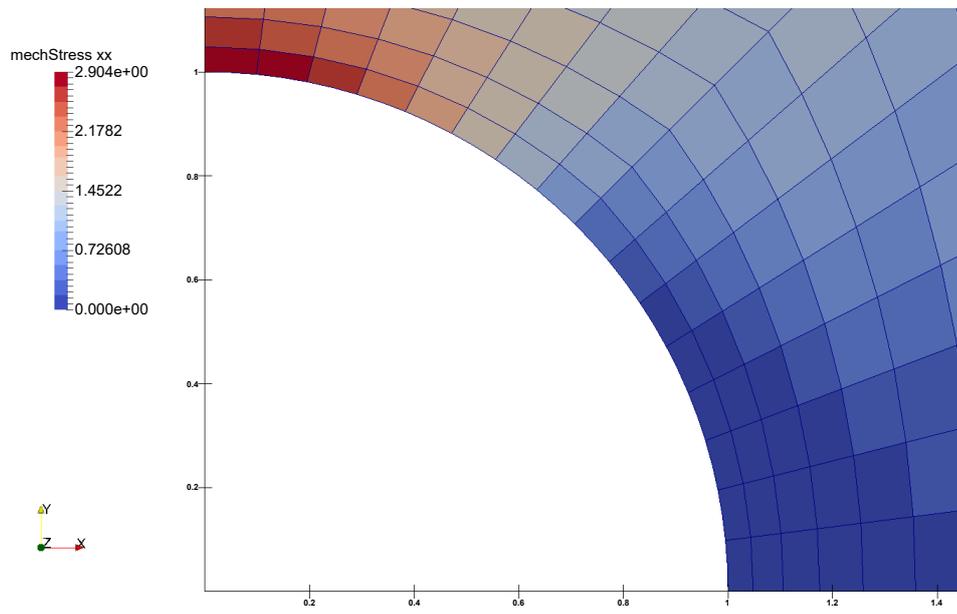


Abbildung 3.10: Fringeplot Spannungsfeld  $\sigma_{xx}$

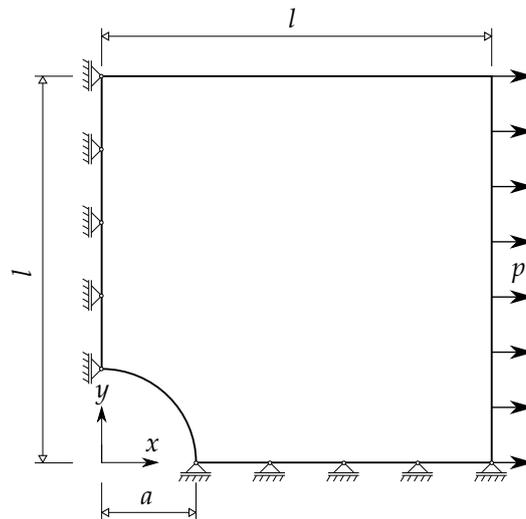


Abbildung 3.11: Geometrie des Testfalles Lochscheibe im FE-Modell

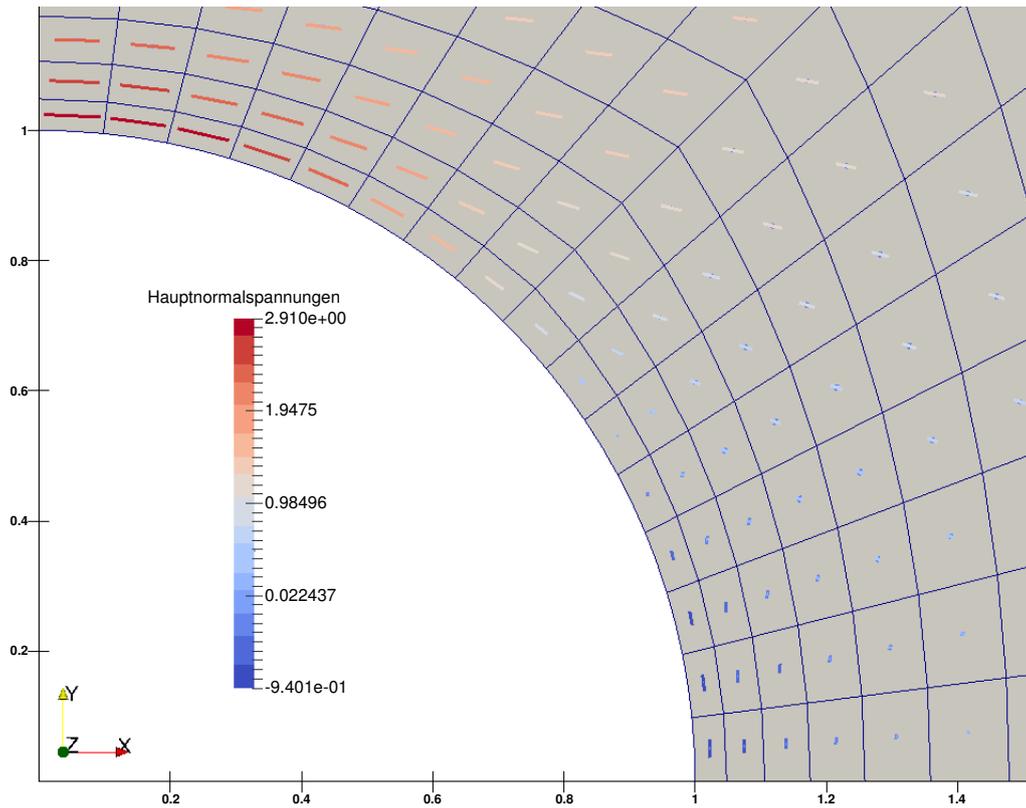


Abbildung 3.12: Vektorplot Hauptspannungen Lochscheibe

Es ist deutlich die Spannungsspitze im Punkt  $x = 0, y = a$  ersichtlich, wobei der Spitzenwert der Spannungsüberhöhung auf den Faktor 3 in der Simulation nicht erreicht wurde. Grund dafür ist die Auswertung des Spannungstensors im Mittelpunkt des Elementes und nicht direkt am Rand, wie in Kap. 3.3.3 bereits erwähnt wurde.

Wie aus Gl. (3.33) zu erwarten war, ist entlang des Randes der Bohrung ein einachsiger Spannungszustand  $\sigma_{\varphi\varphi}$  vorhanden, welcher somit auch gleichzeitig ein Hauptspannungszustand ist.

Im Punkt  $r = a, \varphi = \frac{\pi}{6}$  verschwindet die Umfangsspannung  $\sigma_{\varphi\varphi}$ , wie Gl. (3.33) beschreibt. Somit liegt hier ein völlig spannungsfreier Zustand vor, was auch aus der Berechnung der Hauptspannungen in der Simulation hervorgeht.

Mit zunehmender Entfernung von der Bohrung verschwinden die Spannungsanteile, die durch die Bohrung hervorgerufen werden, und es verbleibt der einachsige Zug  $p$  als einzige Komponente des Spannungstensors.

## 3.5 MEMS-Lautsprecher

### 3.5.1 MEMS-Technologie

Als mikroelektromechanische Systeme (micro electro mechanical systems, MEMS) werden Systeme bezeichnet, die eine charakteristische Länge von weniger als 1  $\mu\text{m}$  bis mehreren mm besitzen, elektrische und mechanische Komponenten aufweisen und in integrierten Schaltkreisen verbaut sind [21]. Die Einsatzgebiete im Bereich der Sensor- und Aktortechnologie sind vielfältig, wie die folgende, bei Weitem nicht vollständige, Auflistung zeigen soll:

- Lautsprecher
- Drucksensoren, z.B. nach [22]
- Mikromotoren, z.B. nach [23]
- MEMS zur Energiegewinnung, z.B. nach [24]
- Akustiksensoren, z.B. nach [25]

Für die Entwicklung von MEMS Lautsprechern werden verschiedene physikalische Prinzipien angewandt. Diese können laut [26] nach dem elektrodynamischen Prinzip, nach dem elektrostatischen Prinzip oder nach dem piezoelektrischen Prinzip entwickelt werden, wobei die beiden letzteren eine relativ neue Entwicklung darstellen.

#### Herstellung MEMS Lautsprecher

Der Produktionsprozess piezoelektrischer MEMS Lautsprecher wird beispielsweise in [27] detailliert erläutert. Dabei werden auf ein Substrat, meist ein Si-Wafer, nacheinander mehrere Schichten verschiedener Materialien aufgedampft, etwa mittels chemischer Gasphasenabscheidung oder Sputtern. Mittels verschiedener Blenden und Behandlungszeiten kann die Form und die Dicke der aufgetragenen Schichten genau eingestellt werden. Die Membran, welche im Allgemeinen möglichst dünn sein soll, wird daraufhin durch Ätzen des Substrats freigelegt. Aus Herstellungsgründen sind die aufgetragenen Schichten im Durchmesser etwas größer als der freigeätzte Bereich, siehe Abb. 3.13(a).

### 3.5.2 Piezoelektrischer Effekt

Der piezoelektrische Effekt beschreibt den Zusammenhang zwischen der elektrischen Feldstärke  $\mathbf{E}$  und der elektrischen Induktion mit der mechanischen Spannung und Verzerrung piezoelektrischer Materialien [3]. Beim direkten piezoelektrischen Effekt entsteht beim Aufbringen einer mechanischen Spannung durch die elektrische Polarisierung des Materials eine Ladung im Material, während beim inversen piezoelektrischen Effekt durch das Anlegen einer Spannung an den gegenüberliegenden Oberflächen bzw. Elektroden des Materials eine mechanische Verzerrung entsteht [3]. Dieses kann in der Form

$$\boldsymbol{\sigma} = [\mathbf{c}^E]\boldsymbol{\varepsilon} - [\mathbf{e}]^T \mathbf{E} \quad (3.34)$$

angegeben werden, wobei  $\boldsymbol{\sigma}$  und  $\boldsymbol{\varepsilon}$  hier in Voigt-Notation dargestellt sind,  $[\mathbf{c}^E]$  den Elastizitätstensor und  $[\mathbf{e}]$  den piezoelektrischen Koppeltensor ausdrücken [3].

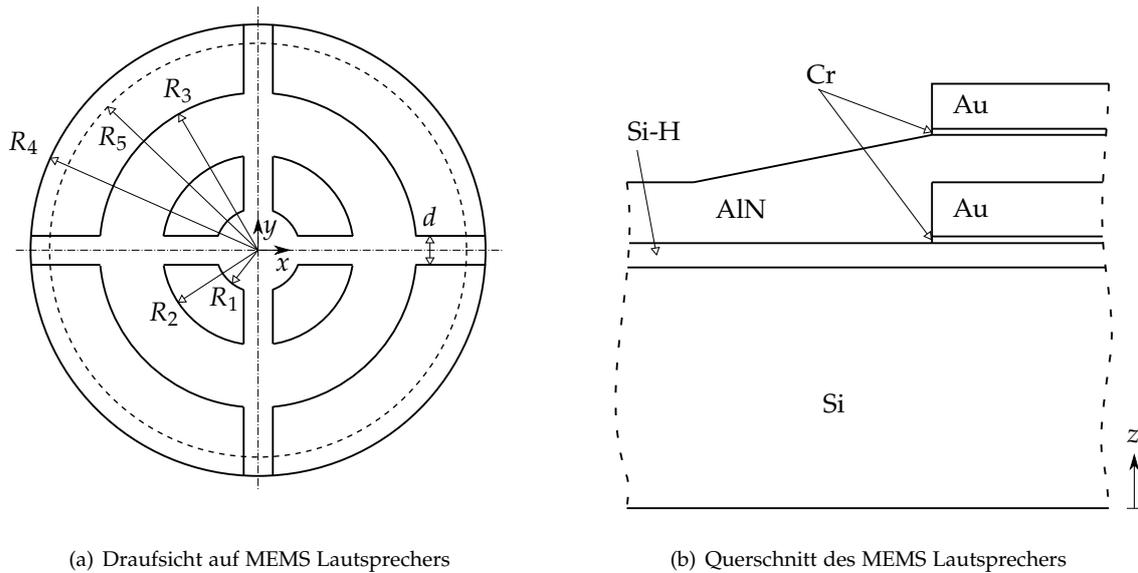


Abbildung 3.13: Geometriedaten des MEMS-Lautsprechers

### 3.5.3 Modellierung des Lautsprechers

#### Geometrie

Der im Folgenden besprochene MEMS Lautsprecher nach dem piezoelektrischen Prinzip ist eine derzeitige Forschungsarbeit am Institut für Mechanik und Mechatronik an der TU Wien, wobei weitere Forschungsarbeiten zum Thema in [28] zu finden sind. Die Geometrie des Modells der Lautsprechermembran ist Abb. 3.13 zu entnehmen, während die dazugehörigen Abmaße in Tab. 3.1 und 3.2 eingetragen sind. Die dargestellte Variante in Abb. 3.13 besitzt 4 Segmente mit Elektroden im Innenbereich und Außenbereich. Abbildung 3.13(b) zeigt den Querschnitt der Platte. Diese ist in jenem Bereich, wo die Elektroden im Produktionsprozess aufgebracht werden, dicker als im übrigen Bereich. Es kann über die Elektroden bestehend aus Gold (Au) eine elektrische Spannung angelegt werden, welche ein elektrisches Feld zwischen den Elektroden erzeugen. Dadurch erfährt die piezoelektrische Aluminiumnitridschicht (AlN) eine Verformung, was wiederum eine Verformung der gesamten Membran zur Folge hat. Die Siliziumschicht (Si) dient als Trägerschicht für die, im Produktionsprozess aufgedampften, weiteren Schichten. Die Chromschichten (Cr) dienen als Haftverbesserer.

Tabelle 3.1: Bemaßungen der Lautsprechermembran

Abmaß	Länge in $\mu\text{m}$
$R_1$	30
$R_2$	85
$R_3$	155
$R_4$	315
$R_5$	300
$d$	40

Tabelle 3.2: Schichtdicken der Lautsprechermembran

Schicht	Dicke in $\mu\text{m}$
Si-Schicht	2
Si-H-Schicht	0.2
Cr-Schichten	0.05
untere Au-Elektrode	0.45
obere Au-Elektrode	0.4
AlN-Schicht	0.4

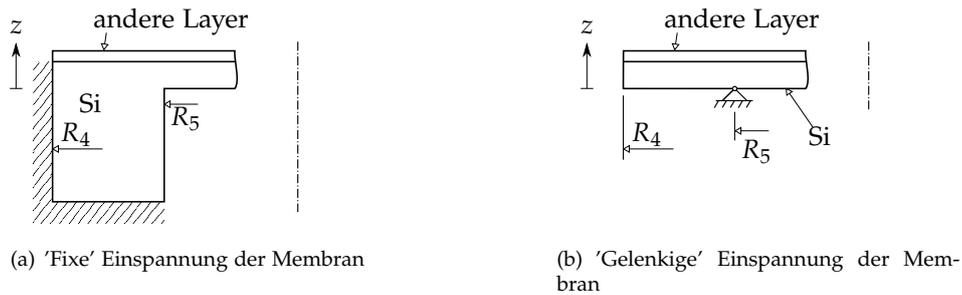


Abbildung 3.14: Modellierung Randbedingung MEMS Lautsprecher

### Materialdaten

Für die Modellierung wurden die nachfolgend genannten Materialdaten verwendet.

**Si** In [29] wird die Steifigkeitsmatrix von orthotropen Silizium angegeben, wobei die  $x$ -Achse mit der kristallographischen  $[110]$  Richtung im Siliziumgitter, die  $y$ -Achse mit der  $[\bar{1}10]$  Richtung und die  $z$ -Achse mit der  $[001]$  Richtung zusammenfällt. Die „gedopte“ Si-H-Schicht wurde mit derselben Steifigkeitsmatrix modelliert.

**Cr, Au** Chrom wurde nach [30] als isotropes Material mit  $E = 279$  GPa und  $\nu = 0.21$  modelliert. Der E-Modul von isotropen Gold beträgt nach [31]  $77.2$  GPa, die Querdehnungszahl  $\nu$  ist  $0.42$ .

**AlN** Die Elastizitätsmatrix für AlN wurde nach [32] verwendet. Für die piezoelektrischen Eigenschaften wurde [33] herangezogen.

### Randbedingungen, Beschaltung der Elektroden

Da das Siliziumsubstrat sehr viel größer und dicker ist als die schwingende Membran, wurde der Berechnungsraum der Simulation in radialer Richtung und in  $z$ -Richtung begrenzt und durch eine fixe mechanische Einspannung ersetzt, siehe Abb. 3.14(a). Alternativ dazu wurde auch eine gelenkige Einspannung modelliert, wie in Abb. 3.14(b) dargestellt wird.

Das elektrische Potential aller unteren Elektroden wurde geerdet, das elektrische Feld wird mittels der oberen Elektroden eingestellt. Mit unterer Elektrode ist jene gemeint, die näher zum Ursprung der  $z$ -Koordinatenachse liegt. Im Modell wurde das elektrische Potential an den Oberflächen (surfaces) zwischen den Elektroden und der AlN-Schicht angegeben. Für alle Testfälle wurde eine statische Elektrodenbeschaltung nach Tab. 3.3 angenommen.

Tabelle 3.3: Beschaltung Elektroden

	unten	oben
Innenelektroden	0 V	-1 V
Außenelektroden	0 V	+1 V

### 3.5.4 Vergleich bestehender MEMS Lautsprecher

In der Literatur sind bereits einige MEMS Lautsprecher nach dem piezoelektrischen Prinzip beschrieben. Tabelle 3.1 gibt die Ergebnisse dieser Forschungsarbeiten wieder, wobei  $SPL_{\max}$  die maximal gemessene Schalldruckleistung bei der Elektrodenversorgungsspannung  $U_V$  und der Signalfrequenz  $f$  im Abstand von  $d_{\text{Messung}}$  zur Membran ist.

Tabelle 3.4: Vergleich verschiedener MEMS Lautsprecher nach dem piezoelektrischen Prinzip

Eigenschaft	Einheit	[34]	[35]	[36]	[27]
Form der Membran		Halbsphäre	quadratisch	rechteckig	quadratisch
Größe der Membran	mm	$\varnothing 3.8$	5x5	18x20	3x3
Dicke der Membran	$\mu\text{m}$	3.5	2.7	80	3
Material der piezoelektr. Schicht		ZnO <sup>1</sup>	ZnO	PZT <sup>2</sup>	ZnO
Dicke der piezoelektr. Schicht	$\mu\text{m}$	0.5	0.5	40	1.5
$u_{\max}$	$\mu\text{m}$	-	-	11.8 <sup>3</sup>	1 <sup>4</sup>
$\frac{u_{\max}^5}{d_{\max}}$		-	-	$5.9 \cdot 10^{-4}$	$3.33 \cdot 10^{-4}$
$SPL_{\max}$	dB	88.2	92	95	83.1
@ $U_V$	V	30	6	16	15
@ $f$	kHz	18	3	0.45	13.3
@ $d_{\text{Messung}}$	mm	5	2	10	10

### 3.5.5 Ergebnisse

Die folgenden Unterkapitel zeigen die Ergebnisse der linear statischen Analysen des MEMS Lautsprechers. Die AlN-Schicht wurde mit der mechanisch-piezoelektrisch gekoppelten partiellen Differentialgleichungen (PDE) modelliert, alle anderen Schichten mit der mechanischen PDE. Die Randbedingungen des elektrischen Feldes wurde in den Oberflächenelementen zwischen Elektroden und AlN-Schicht gesetzt.

#### Variation der Elementanzahl

Aufgrund der Geometrie des Modells, welche eine kleine Dicke im Vergleich zu den übrigen Abmessungen aufweist, lag eine Modellierung mit Schalenelementen nahe. Diese sind allerdings in CFS++ nicht implementiert, weswegen eine Modellierung mit 3D Volumselementen notwendig wurde. Außerdem mussten die verschiedenen Lagen der Membran im Finite Elemente Netz abgebildet werden, um eine Zuweisung der verschiedenen Materialeigenschaften zu ermöglichen.

Beim Einsatz von Volumselementen, die eine sehr geringe Dicke im Vergleich zu den übrigen Abmessungen besitzen, ist das Phänomen des *lockings* zu beachten [3]. Aus diesem Grund wurde die FE-Analyse für verschiedene Elementanzahlen und verschiedene Ordnungen der Ansatzfunktionen getestet, um ein eventuelles Auftreten erkennen und vermeiden zu können. Abb. 3.15 zeigt die Ergebnisse dieser Tests.

<sup>1</sup>Zinkoxid

<sup>2</sup>Blei-Zirkonat-Titanat

<sup>3</sup>@1 V 492Hz

<sup>4</sup>@15 V, 7.3 kHz

<sup>5</sup> $d_{\max}$  = größte Abmessung

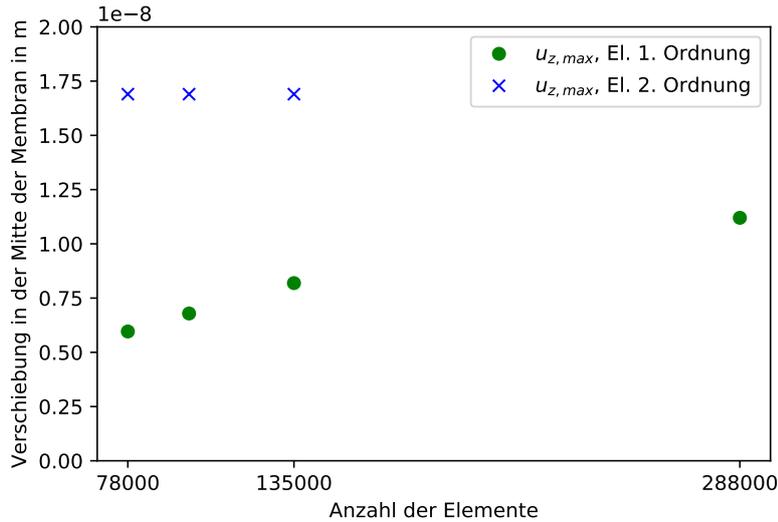


Abbildung 3.15: Konvergenzanalyse des MEMS Lautsprechers für verschiedene Ordnungen der Ansatzfunktion und der Elementanzahl. Einspannung nach 3.14(a).

Für 78000 Elemente beträgt die durchschnittliche Elementkantenlänge in radialer bzw. tangentialer Richtung ca.  $8 \mu\text{m}$ , für 288000 Elemente ca.  $4 \mu\text{m}$ . Es ist deutlich zu erkennen, dass bei der Verwendung von Ansatzfunktionen 1. Ordnung ein starker *locking*-Effekt auftritt und die berechnete maximale Verschiebung mit steigender Elementanzahl nur langsam zum physikalisch korrekten Ergebnis konvergiert. Bei Verwendung von Ansatzfunktionen 2. Ordnung tritt der *locking*-Effekt nicht auf, die Verschiebung bleibt mit steigender Elementanzahl nahezu konstant. Aufgrund dieser Ergebnisse wurde für alle weiteren Berechnungen das größte Netz aus Abb. 3.15 mit Ansatzfunktionen 2. Ordnung gewählt.

### Variation der Elektrodenzahl

Weiters wurde untersucht, welchen Einfluss die Anzahl der Elektroden bzw. der Segmente (siehe Abb. 3.13(a) mit 4 Segmenten) auf die maximale Auslenkung der Membran hat. Für eine möglichst großen Schalldruckpegel ist eine große Auslenkung erstrebenswert. In Tab. 3.5 ist das Ergebnis dieser Berechnungen wiedergegeben.

Tabelle 3.5: Maximale Verschiebung  $u_{z,max}$  für verschiedene Anzahl an Segmenten. Einspannung nach 3.14(a).  $d_4 = 2 R_4$

Segmente	$ u_{z,max} $ in nm	$\frac{ u_{z,max} }{d_4}$
4	16.9	$2.82 \cdot 10^{-5}$
5	16.6	$2.76 \cdot 10^{-5}$
6	16.3	$2.71 \cdot 10^{-5}$

Es kann somit davon ausgegangen werden, dass für eine quasistatische Ansteuerung der Membran die maximale Verschiebung der Membran nahezu unverändert bleibt. Die leicht sinkende maximale Auslenkung kann mit dem ebenfalls sinkenden Verhältnis von Elektrodenfläche zu Gesamtfläche bei steigender Segmentanzahl erklärt werden. Möglicherweise hat die Anzahl der Segmente jedoch

einen Einfluss auf die Eigenfrequenzen und die dazugehörigen Schwingungsformen der Platte. Dieser Einfluss wurde in dieser Arbeit jedoch nicht untersucht.

### Variation der modellierten Einspannung

In einem weiteren Test wurde der Einfluss der Einspannung auf das Verformungsverhalten der Membran untersucht. Dabei wurden eine fixe Einspannung nach Abb. 3.14(a) und eine gelenkige Einspannung nach Abb. 3.14(b) getestet. Es zeigte sich, dass die Modellierung der Randbedingungen einen entscheidenden Einfluss auf das Verformungsverhalten der Membran hat, wie Abb. 3.16 und Abb. 3.17 zeigen. Für die geometrische Modellierung nach Abb. 3.14(a) wurde nur ein kleiner Bereich des Wafers mitmodelliert. Der im Modell fix eingespannte Rand ist jedoch weit genug vom deformierten Bereich weg, sodass die Vernachlässigung der Nachgiebigkeit des (theoretisch) unendlich ausgedehnten Siliziumwafers nach dem St. Venant'schen Prinzip nicht ins Gewicht fällt. Dieses äquivalente Modell sollte daher die Realität besser abbilden als jenes nach Abb. 3.14(b).

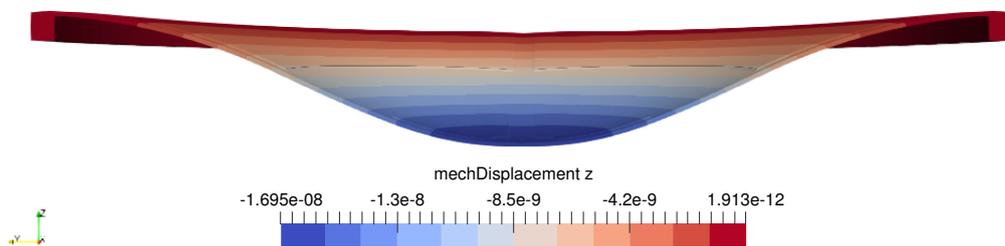


Abbildung 3.16: Deformationsplot MEMS Lautsprecher, fixe Einspannung

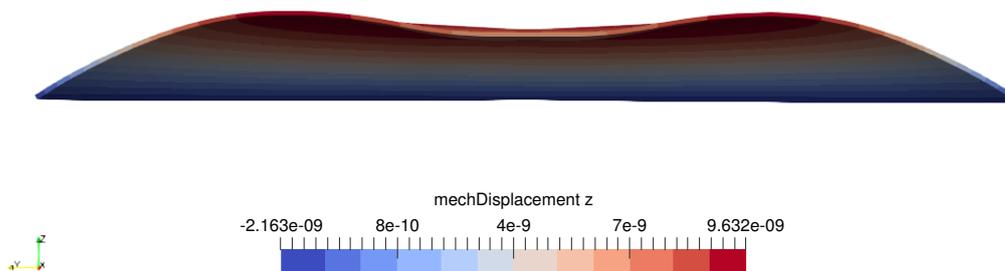
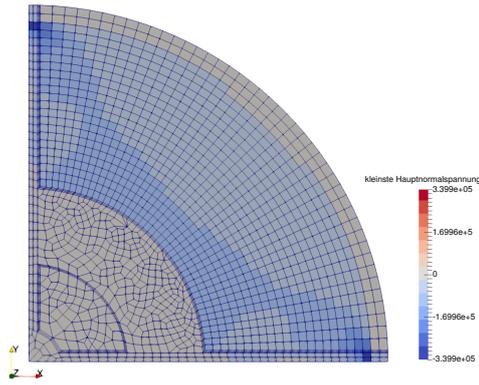


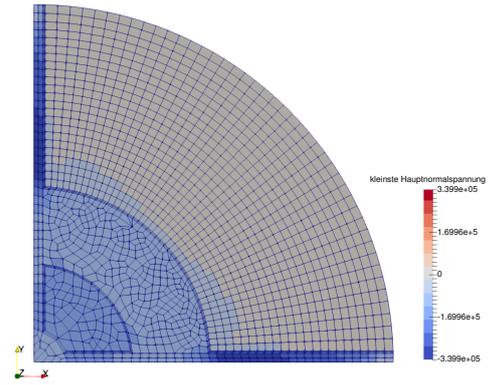
Abbildung 3.17: Deformationsplot MEMS Lautsprecher, gelenkige Einspannung

### Hauptnormalspannungen im Si-Layer

Für die nachfolgenden Plots wurde ein Modell mit fixer Einspannung betrachtet. In den folgenden Abbildungen wird nur ein Segment betrachtet, da sowohl Geometrie als auch Belastung für das 4-Segment-Modell um die x-Achse, als auch die y-Achse symmetrisch sind. Abbildungen 3.18 bis Abb. 3.20 zeigen die Hauptnormalspannungsverteilung des Silizium-Layers nahe der Unterseite für  $z = 0.33 \mu\text{m}$  und nahe der Oberseite für  $z = 1.67 \mu\text{m}$ . Dies entspricht den z-Koordinaten der untersten und obersten Elemente des Si-Layers im Elementmittelpunkt, and denen CFS++ den Spannungstensor auswertet.

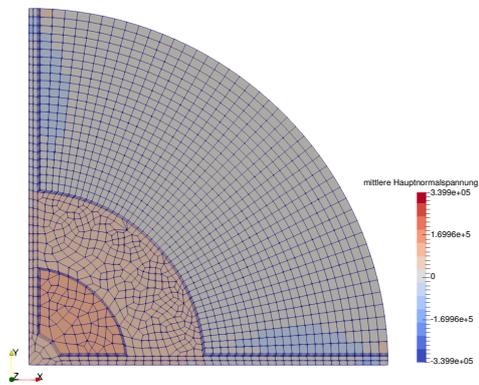


(a)  $z = 0.33 \mu\text{m}$

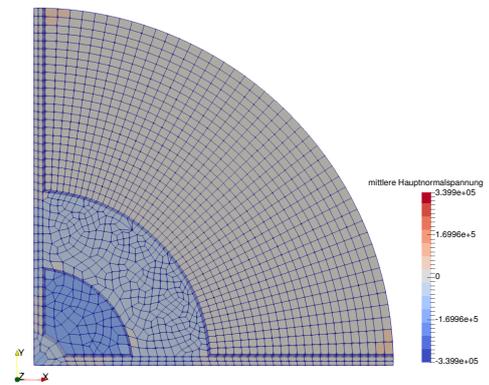


(b)  $z = 1.67 \mu\text{m}$

Abbildung 3.18: Fringeplots kleinste Hauptnormalspannungen im Si-Layer

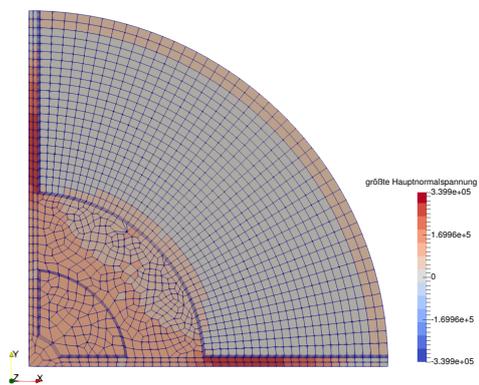


(a)  $z = 0.33 \mu\text{m}$

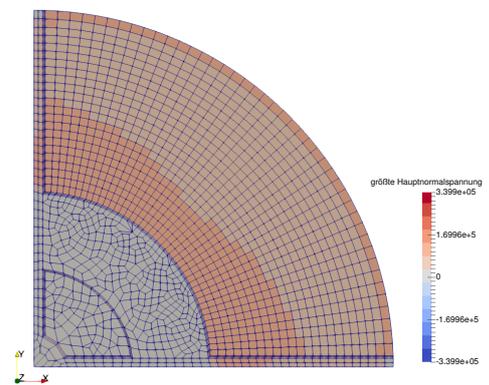


(b)  $z = 1.67 \mu\text{m}$

Abbildung 3.19: Fringeplots mittlere Hauptnormalspannungen im Si-Layer



(a)  $z = 0.33 \mu\text{m}$



(b)  $z = 1.67 \mu\text{m}$

Abbildung 3.20: Fringeplots größte Hauptnormalspannungen im Si-Layer

Die Verformungen der Piezoschichten, welche bei der angegebenen Beschaltung für die äußeren Schichten eine Ausdehnung in der Ebene und für die inneren Schichten eine Kontraktion in der Ebene sind, kann analog zu einem wechselweise aufgetragenen Temperaturmoment auf die Platte betrachtet werden, was auch die Deformationsfigur in Abb. 3.17 deutlich zeigt.

Für eine genauere Betrachtung der Randbereiche der Außenelektroden wurde ein Vektorplot der Hauptnormalspannungen für  $z = 1.67 \mu\text{m}$  angefertigt, siehe Abb. 3.21. Es gilt zu beachten, dass die Hauptspannungsvektoren in der Nähe des Überganges zur Elektrode nicht genau in der Ebene liegen, da diese Abweichungen aber relativ gering sind und schnell abklingen wird auf die Darstellung hier verzichtet.

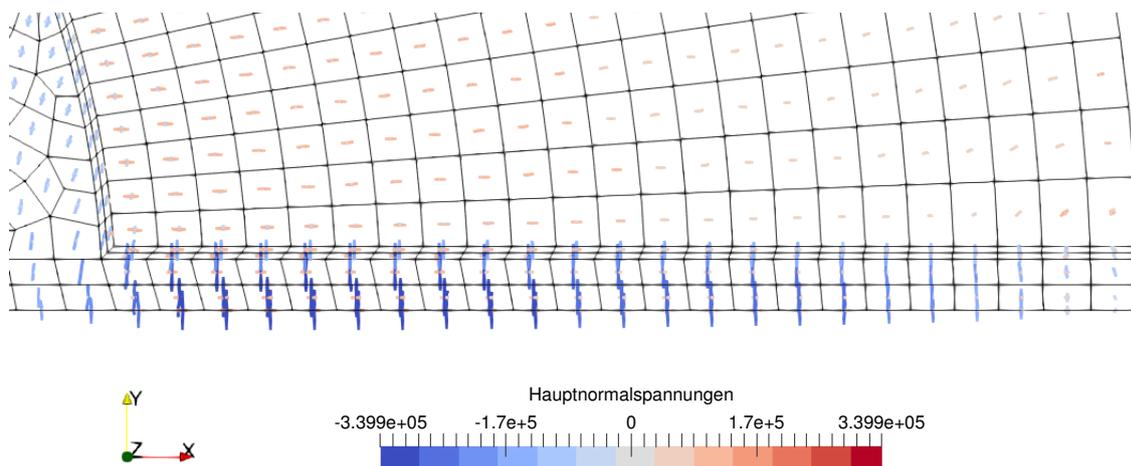


Abbildung 3.21: Vektorplot der Hauptnormalspannungen im Randbereich der Außenelektrode,  $z = 1.67 \mu\text{m}$

Um die geometrische Symmetriebedingung aufrechtzuerhalten, muss sich an der Oberseite der Si-Schicht eine Druckspannung in y-Richtung einstellen, während sich an der Unterseite eine Zugspannung einstellen muss, was von Abb. 3.22 gezeigt wird. Diese zeigt auch, dass diese Annahme für den Außenbereich der Elektrode jedoch nicht mehr gilt, da der Spannungszustand durch das Krümmungsverhalten der Platte und eventuell auftretende Randeffekte durch den Übergang in den ungeätzten, dicken Bereich bestimmt wird. Weiters ist nahe des Außenrandes die Ausdehnung des Piezo-Layers durch den dicken Rand eher behindert.

Abbildung 3.23 zeigt einen Vektorplot der Hauptnormalspannungen im Si-Layer,  $z = 1.67 \mu\text{m}$  im Bereich der Innenelektrode. Man kann erkennen, dass der Spannungszustand, wie es bei einem rein radialsymmetrischen Problem der Fall sein müsste ( $r = 0 : \sigma_{rr} = \sigma_{\varphi\varphi}$ , siehe [1]), im Bereich der Innenelektrode massiv gestört wird. Es stellt sich durch die Kontraktion der piezoelektrischen Schicht in der x-y-Ebene ein zweiachsiger, in der Ebene hydrostatischer Druckspannungszustand ( $\sigma_2 = \sigma_3$ ) an der Oberseite des Si-Layers ein, der keine ausgeprägte radiale Orientierung mehr besitzt. Die unregelmäßige Anordnung der Hauptspannungsvektoren resultiert aus dem Eigenwertproblem, welche bei doppelten Eigenwerten keine Eindeutigkeit der Richtung der Eigenvektoren ergibt, wie schon in Kap. 3.2 diskutiert. Der Randeffect, welcher in den Abb. 3.21 und 3.22 gezeigt ist, tritt bei den Innenelektroden nicht auf.

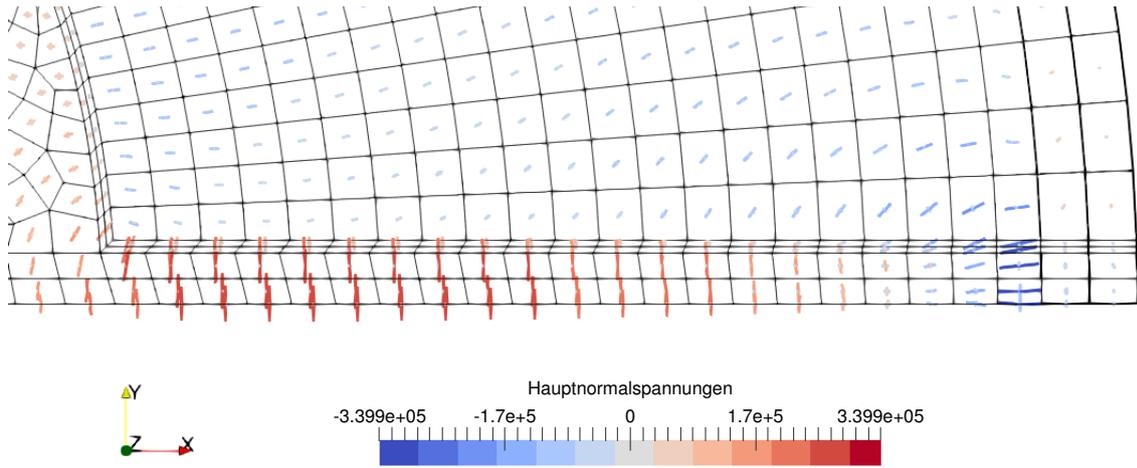


Abbildung 3.22: Vektorplot der Hauptnormalspannungen im Randbereich der Außenelektrode,  $z = 0.33 \mu\text{m}$

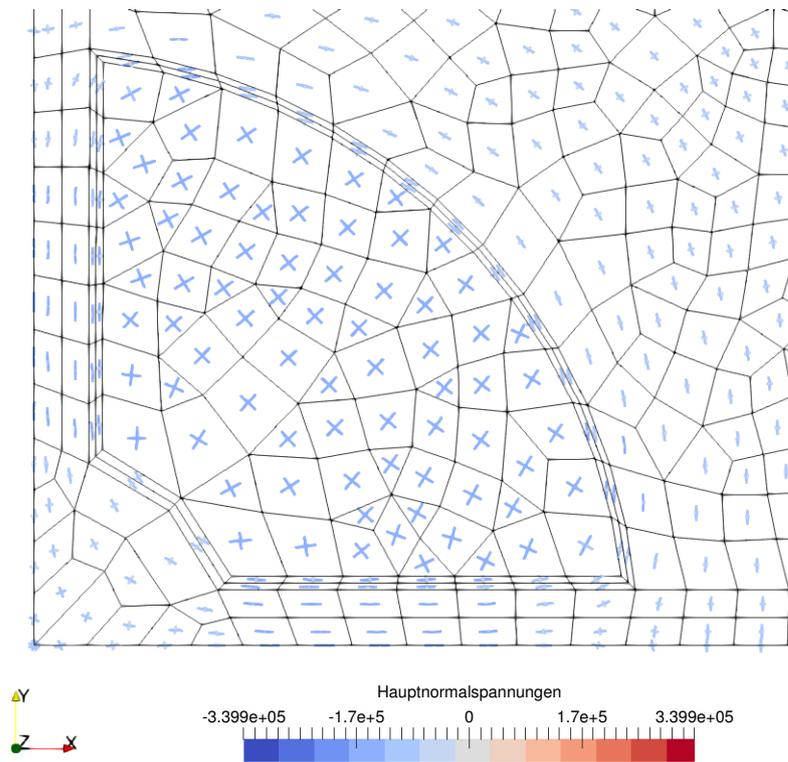


Abbildung 3.23: Vektorplot der Hauptnormalspannungen im Bereich der Innenelektrode,  $z = 1.67 \mu\text{m}$

# Conclusio

Diese Arbeit beschäftigte sich mit der Implementierung der Hauptnormalspannungen und -verzerrungen in das Finite Elemente Programm CFS++. Aufbauend auf Überlegungen zur Berechnung der Hauptnormalspannungen aus dem Spannungstensor und der Hauptnormalverzerrungen aus dem Verzerrungstensor wurde der wohlbekanntere Berechnungsweg in das Programm implementiert. Um eine wiederholte Lösung des Eigenwertproblems zu vermeiden, wurde außerdem eine Cache-Funktion in das Programm implementiert.

In mehreren Testfällen konnte die Berechnung der Hauptnormalspannungen und -verzerrungen validiert werden. Dies gelang mithilfe analytisch lösbarer Probleme aus der Festigkeitslehre. Zum Abschluss wurde ein Modell eines MEMS-Lautsprechers aufgebaut und auch in Hinblick auf den Spannungszustand diskutiert. Dabei zeigte sich auch, dass eine Visualisierung mithilfe von Vektorplots die Darstellung der Spannungs- und Verzerrungszustände sehr erleichtert und dem Entwickler einen besseren Einblick in den Spannungs- und Verzerrungszustand komplexer Modelle gibt.

# Abbildungsverzeichnis

1.1	Volumselement mit resultierenden Spannungsvektoren . . . . .	3
1.2	Komponenten des Spannungstensors am Volumselement . . . . .	3
1.3	Spannungsvektor in einer beliebigen Schnittebene . . . . .	4
1.4	Geometrische Betrachtung der Verzerrungen . . . . .	7
1.5	Spannungsvektor, zerlegt in Normalspannungs- und Schubspannungskomponente . . .	11
1.6	Rotation des Koordinatensystems . . . . .	13
1.7	Versagenskriterium nach Tresca im Hauptspannungsraum . . . . .	15
1.8	Versagenskriterium nach Mises im Hauptspannungsraum . . . . .	16
1.9	Versagenskriterium nach Mises und Tresca für den ebenen Spannungszustand . . . . .	16
2.1	Schema zur Implementierung der Hauptnormalspannungen . . . . .	22
2.2	Schema zur Implementierung der CoeffFunctionCache . . . . .	25
3.1	2D Element . . . . .	30
3.2	Vektorplot Hauptnormalspannungen Single Element 2D . . . . .	31
3.3	Vektorplot Hauptnormalverzerrungen Single Element 3D . . . . .	32
3.4	Rohr unter Innendruck . . . . .	34
3.5	Geometrie des Testfalles Rohr im FE-Modell . . . . .	35
3.6	Vergleich der analytischen Lösung und der FE-Lösung für den Testfall Rohr . . . . .	36
3.7	Fringeplots Hauptnormalspannungen Rohr unter Innendruck . . . . .	36
3.8	Vektorplot Hauptnormalspannungen Rohr unter Innendruck . . . . .	37
3.9	Geometrie des Testfalles Lochscheibe . . . . .	38
3.10	Fringeplot Spannungsfeld $\sigma_{xx}$ . . . . .	39
3.11	Geometrie des Testfalles Lochscheibe im FE-Modell . . . . .	39
3.12	Vektorplot Hauptspannungen Lochscheibe . . . . .	40
3.13	Geometriedaten des MEMS-Lautsprechers . . . . .	42
3.14	Modellierung Randbedingung MEMS Lautsprecher . . . . .	43
3.15	Konvergenzanalyse des MEMS Lautsprechers . . . . .	45
3.16	Deformationsplot MEMS Lautsprecher, fixe Einspannung . . . . .	46
3.17	Deformationsplot MEMS Lautsprecher, gelenkige Einspannung . . . . .	46
3.18	Fringeplots kleinste Hauptnormalspannungen im Si-Layer . . . . .	47
3.19	Fringeplots mittlere Hauptnormalspannungen im Si-Layer . . . . .	47
3.20	Fringeplots größte Hauptnormalspannungen im Si-Layer . . . . .	47
3.21	Vektorplot der Hauptnormalspannungen im Randbereich der Außenelektrode, $z = 1.67 \mu\text{m}$	48
3.22	Vektorplot der Hauptnormalspannungen im Randbereich der Außenelektrode, $z = 0.33 \mu\text{m}$	49
3.23	Vektorplot der Hauptnormalspannungen im Bereich der Innenelektrode, $z = 1.67 \mu\text{m}$ . .	49

# Tabellenverzeichnis

2.1	Messwerte des Benchmarks <code>std::map</code> mit <code>boost::unordered map</code> . . . . .	27
2.2	Benchmark zum Cache verschiedener Coefficient Functions . . . . .	27
3.1	Bemaßungen der Lautsprechermembran . . . . .	42
3.2	Schichtdicken der Lautsprechermembran . . . . .	42
3.3	Beschaltung Elektroden . . . . .	43
3.4	Vergleich verschiedener MEMS Lautsprecher nach dem piezoelektrischen Prinzip . . . . .	44
3.5	Maximale Verschiebung $u_{z,max}$ in Abhängigkeit der Segmentanzahl . . . . .	45

# Literatur

- [1] Heinz Parkus. *Mechanik der festen Körper*. Springer Wien, 1966.
- [2] Holm Altenbach. *Kontinuumsmechanik*. Springer Berlin Heidelberg, 2012.
- [3] Manfred Kaltenbacher. *Numerical simulation of mechatronic sensors and actuators*. Springer Berlin Heidelberg, 2007.
- [4] Franz G. Rammerstorfer. *Skriptum zur Vorlesung Leichtbau*. Institut für Leichtbau und Struktur-Biomechanik, TU Wien, 2013.
- [5] Karl-Heinz Schild. *Skriptum zur Vorlesung Dynamische Optimierung*. Universität Marburg, 2015.
- [6] Hendrik Kuhlmann. *Numerische Methoden der Ingenieurwissenschaften*. Institut für Strömungsmechanik und Wärmeübertragung, 2014.
- [7] Herbert A. Mang und Günther Hofstetter. *Festigkeitslehre*. Springer Berlin-Heidelberg, 2013.
- [8] Ilja Nikolajewitsch Bronstein u. a. *Taschenbuch der Mathematik*. Harri Deutsch Verlag, 2008.
- [9] Lloyd N. Trefethen und David Bau. *Numerical linear algebra*. Society for Industrial und Applied Mathematics, 1997.
- [10] Peter Pesic. *Abels Beweis*. Springer Berlin, 2005.
- [11] Wolfgang Dahmen und Arnold Reusken. *Numerik für Ingenieure und Naturwissenschaftler*. Springer-Lehrbuch. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2006.
- [12] Hans Hansen. *Die Top 10 der Algorithmen. Der QR-Algorithmus*. TU Chemnitz, 2005.
- [13] Zhaojun Bai und James Demmel. »On a block implementation of Hessenberg multishift QR iteration«. In: *International Journal of High Speed Computing* 1.01 (1989), S. 97–112.
- [14] Klaus-Jürgen Bathe. *Finite element procedures*. Prentice Hall, 1996.
- [15] Manfred Kaltenbacher und Andreas Hüppe. »Finite Elemente Verfahren für mechatronische Systeme«. In: *e & i Elektrotechnik und Informationstechnik* 132.8 (2015), S. 448–455. URL: <http://dx.doi.org/10.1007/s00502-015-0371-9>.
- [16] Edward Anderson u. a. *LAPACK Users' Guide*. Society for Industrial und Applied Mathematics, 1999.
- [17] Bjarne Stroustrup. *Die C++ Programmiersprache*. Addison-Wesley, 2010.
- [18] *C++ Boost Bibliothek*. URL: <http://www.boost.org/> (besucht am 16.05.2017).
- [19] Arindam Mukherjee. *Learning Boost C++ Libraries*. Packt Publishing, 2015.
- [20] Karl Girkmann. *Flächentragwerke*. Springer Wien-New York, 1978.
- [21] Mohamed [Hrsg.] Gad-el-Hak. *The MEMS handbook*. The mechanical engineering handbook series. Boca Raton, Fla. [u.a.]: CRC Press, 2002.

- [22] Heng-Chung Chang u. a. »Wireless magnetostrictive type inductive sensing CMOS-MEMS pressure sensors«. In: *IEEE 29th International Conference on Micro Electro Mechanical Systems (MEMS)*. Jan. 2016, S. 218–221.
- [23] Long-Sheng Fan, Yu-Chong Tai und Richard S. Müller. »IC-Processed Electrostatic Micro-Motors«. In: *IEEE International Electron Device Meeting* (1988), S. 666–669.
- [24] Min-Ho Seo u. a. »An electrostatic energy harvester exploiting variable-area water electrode by respiration«. In: *28th IEEE International Conference on Micro Electro Mechanical Systems (MEMS)*. Jan. 2015, S. 126–129.
- [25] Yipeng Lu u. a. »Short-range and high-resolution ultrasound imaging using an 8 MHz aluminum nitride PMUT array«. In: *28th IEEE International Conference on Micro Electro Mechanical Systems (MEMS)*. Jan. 2015, S. 140–143.
- [26] David Tumpold. *Modeling and optimization methods of an electrostatically actuated MEMS speaker*. Shaker Verlag, 2014.
- [27] Sang Choon Ko u. a. »Micromachined piezoelectric membrane acoustic device«. In: *Sensors and Actuators A: Physical* 103 (2003), S. 130–134. URL: <http://www.sciencedirect.com/science/article/pii/S0924424702003102>.
- [28] Florian Toth u. a. »Nonlinear analysis of a prestressed piezoelectric multilayer plate«. In: *8th ECCOMAS Thematic Conference on Smart Structures and Materials*. 2017, S. 1307–1315.
- [29] Matthew A. Hopcroft, William D. Nix und Thomas W. Kenny. »What is the Young’s Modulus of Silicon?«. In: *Journal of Microelectromechanical Systems* 19.2 (Apr. 2010), S. 229–238.
- [30] Vera M. Marx u. a. »The influence of a brittle Cr interlayer on the deformation behavior of thin Cu films on flexible substrates: Experiment and model«. In: *Acta Materialia* 89 (2015), S. 278–289. URL: <http://www.sciencedirect.com/science/article/pii/S1359645415000609>.
- [31] *MatWeb*. URL: <http://www.matweb.com/> (besucht am 24.05.2017).
- [32] Laurie E. McNeil, Marcos Grimsditch und Roger H. French. »Vibrational Spectroscopy of Aluminum Nitride«. In: *Journal of the American Ceramic Society* 76.5 (1993), S. 1132–1136. URL: <http://dx.doi.org/10.1111/j.1151-2916.1993.tb03730.x>.
- [33] Michael Schneider, Achim Bittner und Ulrich Schmid. »Improved piezoelectric constants of sputtered aluminium nitride thin films by pre-conditioning of the silicon surface«. In: *Journal of Physics D: Applied Physics* 48.40 (2015), S. 405301. URL: <http://stacks.iop.org/0022-3727/48/i=40/a=405301>.
- [34] Yongkui Tang u. a. »Micromachined piezoelectric ultrasonic transducer based on dome-shaped diaphragm supported by flat square diaphragm«. In: *IEEE 29th International Conference on Micro Electro Mechanical Systems (MEMS)*. 2016, S. 1110–1111.
- [35] Seung Hwan Yi und Eun Sok Kim. »Micromachined Piezoelectric Microspeaker«. In: *Japanese Journal of Applied Physics* 44.6R (2005), S. 3836. URL: <http://stacks.iop.org/1347-4065/44/i=6R/a=3836>.
- [36] Hye Jin Kim u. a. »High performance piezoelectric microspeakers and thin speaker array system«. In: *ETRI journal* 31.6 (2009), S. 680–687.