



TECHNISCHE  
UNIVERSITÄT  
WIEN

## DIPLOMARBEIT

# Parametrisierung und Analyse von Gitterstrukturzellen Parametrization and Analysis of Lattice Structure Cells

ausgeführt zum Zwecke der Erlangung  
des akademischen Grades eines Diplom-Ingenieurs

eingereicht an der Technischen Universität Wien  
**Fakultät für Maschinenwesen und Betriebswissenschaften**

von  
**Michael Giritsch**  
Matrikelnummer 11911130

unter der Leitung von  
Univ.Prof.in Dr.-Ing.in Stefanie Elgeti  
Mitwirkung Dipl.-Ing. Michael Kofler  
Institut für Leichtbau und Struktur-Biomechanik

Wien, 2025

Unterschrift

# Inhaltsverzeichnis

<b>Kurzfassung</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>1 Einleitung</b>	<b>1</b>
<b>2 Methoden</b>	<b>3</b>
2.1 Parametrisierung der Einheitszellen . . . . .	4
2.1.1 Spline-basierte Geometriebeschreibung . . . . .	4
2.1.2 Parametrisierung der Einheitszelle . . . . .	8
2.1.3 Erstellen des Finite-Elemente-Netzes im Multipatch . . . . .	11
2.2 Berechnung der mechanischen Eigenschaften . . . . .	13
2.2.1 Finite-Elemente-Simulation . . . . .	13
2.2.1.1 Gleichungen der Kontinuumsmechanik im linear elastischen Fall . . . . .	13
2.2.1.2 Bildung der schwachen Form der Differentialgleichung . . . . .	14
2.2.1.3 Finite-Elemente-Diskretisierung . . . . .	15
2.2.1.4 Beispiel zum Aufbau des Gleichungssystems . . . . .	18
2.2.1.5 Periodische Randbedingung . . . . .	23
2.2.2 Berechnung der Spannungen in simulierter 2D Geometrie . . . . .	28
2.2.3 Berechnung der mechanischen Eigenschaften der Einheitszelle . . . . .	30
<b>3 Ergebnisse</b>	<b>32</b>
3.1 Entwurfsprozess der Einheitszellen . . . . .	32
3.1.1 Entwurf parametrisierbarer Einheitszellen mit variablen mechanischen Eigenschaften	32
3.1.2 Sicherstellen der Konnektivität in der Gitterstruktur durch die Definition variabler Parameter innerhalb der Einheitszelle . . . . .	35
3.1.3 Optimierung der Geometrie für eine Finite-Elemente-Simulation . . . . .	39
3.2 Einheitszellendesign . . . . .	40
3.2.1 Chi-Einheitszelle . . . . .	41
3.2.2 Chi-Bar-Einheitszelle . . . . .	45
3.2.3 Chi-Bar-Flex-Einheitszelle . . . . .	48
3.3 Simulationsergebnisse . . . . .	52
3.3.1 Vergleichssimulation in Abaqus . . . . .	55
3.4 Netzfeinheitsstudien . . . . .	59
3.4.1 Netzfeinheitsstudie Chi-Bar . . . . .	59
3.4.2 Netzfeinheitsstudie Chi-Bar-Flex . . . . .	60
3.4.3 Netzfeinheitsstudie Chi . . . . .	61
3.5 Parameterstudien . . . . .	63
3.5.1 Parameterstudie Chi-Bar . . . . .	63
3.5.2 Parameterstudie Chi-Bar-Flex . . . . .	65
3.5.3 Parameterstudie Chi . . . . .	67
<b>4 Ergebnisdiskussion</b>	<b>70</b>
4.1 Einfluss der Parameter auf die mechanischen Eigenschaften . . . . .	70
4.1.1 Einfluss der Parameter in Einheitszelle Chi-Bar . . . . .	70
4.1.2 Einfluss der Parameter in Einheitszelle Chi-Bar-Flex . . . . .	71
4.1.3 Einfluss der Parameter in Einheitszelle Chi . . . . .	71

4.2 Vergleich der realisierbaren Bandbreite der mechanischen Eigenschaften . . . . .	72
4.3 Zusammenfassung und Ausblick . . . . .	73
Abbildungsverzeichnis . . . . .	76
Tabellenverzeichnis . . . . .	78

# Kurzfassung

Ziel dieser Arbeit ist es, Gitterstrukturen zu erstellen, in denen die mechanischen Eigenschaften gezielt und lokal angepasst werden können. Die Gitterstruktur wird aus sich wiederholenden Einheitszellen aufgebaut, die aus Splines zusammengesetzt sind. Um die mechanischen Eigenschaften der Gitterstruktur lokal anzupassen, wird die Geometrie der Einheitszelle verändert. Es werden dazu Parameter definiert, die eine Formbarkeit der Geometrie ermöglichen, indem sie Einfluss auf die Form der Splines nehmen. Im Zuge dieser Arbeit werden verschiedene Einheitszellen erstellt.

Nach der Erstellung der Einheitszelle erfolgt die Untersuchung ihre mechanischen Eigenschaften. Zur Bestimmung dieser Eigenschaften wird eine Finite-Elemente-Simulation durchgeführt, in der die Einheitszellen unter Dehnung untersucht werden. Die Randbedingungen sind so gestaltet, dass sie eine Verformung der Einheitszelle im Verbund der gesamten Gitterstruktur simulieren. Mit diesen Simulationen werden Parameterstudien durchgeführt, um den Einfluss der geometrischen Anpassungen auf Steifigkeit und Verformungsverhalten zu quantifizieren. Die Ergebnisse zeigen, dass sich die mechanischen Eigenschaften der Gitterstruktur gezielt steuern lassen, sodass Strukturen mit gewünschten mechanischen Eigenschaften aufgebaut werden können.

# Abstract

The research objective is to create lattice structures in which the mechanical properties can be specifically and locally modified. The lattice structure is constructed from repeating unit cells composed of splines. In order to locally adjust the mechanical properties of the lattice structure, the geometry of the unit cell is adjusted. To this end, parameters are defined that enable the geometry to be shaped by influencing the form of the splines. In the course of this work, various unit cells are created.

Once the unit cell has been created, its mechanical properties are assessed. To determine these properties, a finite element simulation is performed in which the unit cells are analyzed under stress. The boundary conditions are designed to simulate deformation of the unit cell in the composite of the entire lattice structure. These simulations are used to perform parameter studies to quantify the influence of geometric adjustments on stiffness and deformation behavior. The results show that the mechanical properties of the lattice structure can be specifically controlled in such a way that structures with desired mechanical properties can be constructed.

## Verwendete Formelzeichen

$b_{i,d}(\xi)$	1	Bernsteinpolynom des Kontrollpunktes $i$ der ersten Splinedimension
$b_{j,e}(\eta)$	1	Bernsteinpolynom des Kontrollpunktes $j$ der zweiten Splinedimension
$\xi$	1	Splineparameter der ersten Splinedimension
$\eta$	1	Splineparameter der zweiten Splinedimension
$d$	1	Grad der ersten Splinedimension
$e$	1	Grad der zweiten Splinedimension
$S(\xi, \eta)$	1	Splineposition
$P_{i,j}$	1	Position des Kontrollpunktes
$\sigma$	Pa	Spannungstensor
$\Omega$	1	Rechengebiet
$\partial\Omega$	1	Oberfläche des Rechengebiets
$\mathbf{n}$	1	Normalenvektor zur Oberfläche des Rechengebiets
$\mathbf{f}$	N	Vektor der Volumskräfte
$f^e$	N	Kraftvektor auf nur ein Element bezogen
$\mathbf{C}$	Pa	Steifigkeitstensor
$\varepsilon$	1	Verzerrungstensor
$\mathbf{u}$	m	kontinuierliches Verschiebungsfeld
$\mathbf{v}$	m	Testfunktion
$\Gamma_N$	$m^2$	Neumann-Rand
$\mathbf{t}_N$	Pa	Neumann-Randwert
$r$	1	Referenzelement Achse in x-Richtung
$s$	1	Referenzelement Achse in y-Richtung
$\mathbf{K}$	$\frac{N}{m}$	Steifigkeitsmatrix
$\mathbf{K}^e$	$\frac{N}{m}$	Steifigkeitsmatrix auf nur ein Element bezogen
$n$	1	Anzahl der Knotenpunkte
$N_i$	1	Ansatzfunktion des Knotenpunktes $i$
$u_{k,i}$	m	Verschiebung am Knotenpunkt $i$
$\mathbf{u}_k$	m	Verschiebungsvektor aller Knotenpunkten
$B$	$\frac{1}{m}$	Dehnungs-Verschiebungs-Matrix
$IP$	1	Anzahl Integrationspunkte im Element
$\mathbf{J}_i$	m	Jacobi-Matrix des Integrationspunktes $i$
$w_i$	1	Gewicht des Integrationspunktes $i$
$\mathbf{N}$	1	Matrix der Basisfunktionen
$\mathbf{C}$	$\frac{N}{m}$	Zwangsbedingungsmatrix
$\mathbf{q}$	N	Zwangsbedingungskraftvektor
$\lambda$	m	Lagrange-Multiplikator
$\Pi_p$	$N \cdot m$	Potential des Systems
$\lambda$	Pa	erste Lamé-Konstante
$\mu$	Pa	zweite Lamé-Konstante oder Schubmodul
$E$	Pa	E-Modul
$\sigma_{vm}$	Pa	Von-Mises-Spannung
$\nu$	1	Poissonzahl
$\Delta x$	m	Verzerrung der Einheitszelle in x-Richtung
$\Delta y$	m	Verzerrung der Einheitszelle in y-Richtung
$F$	N/m	Benötigte Kraft zur Deformation der Einheitszelle

$n_R$	1	Anzahl der Randelemente
$\sigma_{mi}$	Pa	Gemittelter Spannungstensor über das Element $i$
$l_i$	m	Länge des Elementes am Rand
$L$	m	Viertel der Normlänge einer Referenzeinheit

# Kapitel 1

## Einleitung

Diese Diplomarbeit behandelt die Erstellung von Einheitszellen und die Berechnung ihrer mechanischen Eigenschaften durch eine Finite-Elemente-Simulation. Diese Einheitszellen werden nebeneinander angeordnet und bilden so eine Gitterstruktur. In Abbildung 1.1 wurde eine solche Gitterstruktur aus 4x2 Einheitszellen aufgebaut.

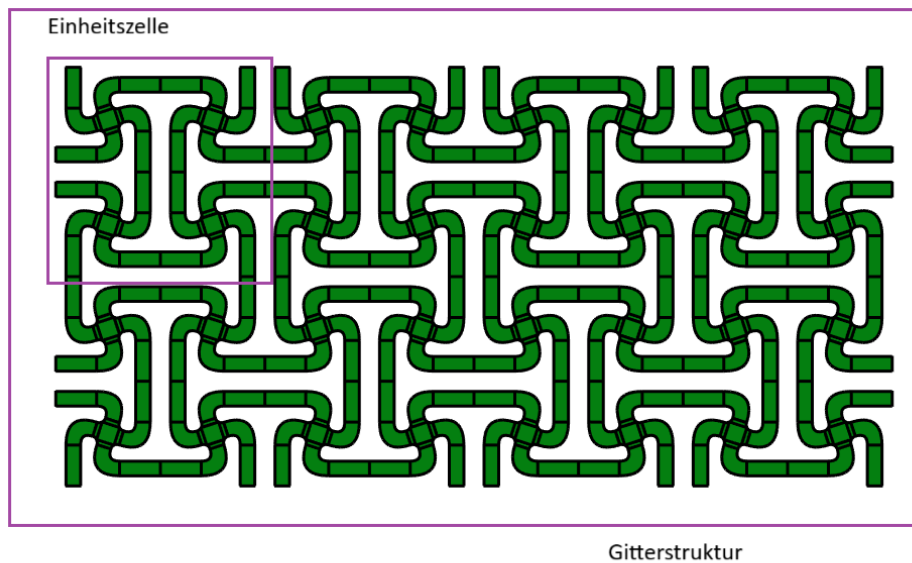


Abbildung 1.1: Gitterstruktur aufgebaut mit Einheitszellen

Gitterstrukturen sind regelmäßig angeordnete Strukturen, die sich netzartig wiederholen. Solche Gitterstrukturen sind häufig in der Natur zu beobachten, welche die ursprüngliche Inspiration für die Anwendungen solcher Strukturen in der Technik bildet. [32] Solche Gitterstrukturen bieten hohe Steifigkeit bei geringem Gewicht. [5]

Die Fertigung solcher Gitterstrukturen ist erst durch die Fortschritte im Gebiet der additiven Fertigung möglich. [31] Mit der additiven Fertigungsmethode des selektiven Laserschmelzens können stabbasierete Gitterstrukturen, dreifach periodische Minimalflächen-Einheitszellen oder Schalen-Gittereinheitszellen

gefertigt werden. [16]

Die Arbeit ist in zwei Teile gegliedert. Einerseits der Entwurf der Einheitszelle, die zu einer Gitterstruktur zusammengesetzt wird, und andererseits die Berechnung der mechanischen Eigenschaften dieser Einheitszellen. Ziel ist es, bei der Durchführung nur Open-Source-Software zu verwenden, um eine freie Zugänglichkeit für weiterführende Anwendungsbereiche zu gewährleisten. [29]

Es gibt verschiedene Möglichkeiten zur parametrischen Beschreibung der Einheitszellen. Eine Möglichkeit, die hier genutzt wird, ist die Verwendung von Splines für den Aufbau der Einheitszelle. [3] Dabei ist die Erstellung dieser Splines so gestaltet, dass das Ändern von Parametern die Form der Gitterstruktur variiert. Hier werden diese Parameter so gewählt, dass eine Änderung der Parameter die mechanischen Eigenschaften beeinflusst. [11,12]

Diese Einheitszellen werden genutzt, um durch das Aneinanderreihen dieser Einheitszellen eine Gitterstruktur zu erstellen. Dabei kann diese Gitterstruktur eine beliebige Form annehmen. Durch die Definition der Parameter in der Einheitszelle wird dabei jede einzelne Einheitszelle individuell angepasst. Dadurch ist es möglich, eine Makrostruktur aufzubauen, welche bei gegebener Belastung das Gewicht minimiert oder eine spezifische Deformation anstrebt.

Im Zuge dieser Arbeit wird auch eine Berechnung der mechanischen Eigenschaften dieser Einheitszellen durchgeführt. Zu diesem Zweck kommt eine Finite-Elemente-Simulation zum Einsatz. Die Einheitszelle wird in Elemente aufgeteilt, die ein Finite-Elemente-Netz bilden. Von Interesse sind die mechanischen Eigenschaften der Einheitszelle, wenn sie als eine von vielen Zellen in einer Gitterstruktur verwendet werden. Es gibt verschiedene Möglichkeiten, diese Eigenschaften zu simulieren. In den hier durchgeführten Simulationen wird eine periodische Randbedingung an den Einheitszellenrändern definiert und eine Deformation aufgebracht. [19, 26] Aus dieser Simulation werden dann Eigenschaften wie Poissonzahl, Spannungsspitze und Steifigkeit errechnet.

# Kapitel 2

## Methoden

Diese Arbeit wird in zwei Teilbereiche eingeteilt. Zum einen die Erstellung der Einheitszellen und zum anderen die Berechnung der Eigenschaften dieser.

In dieser Arbeit werden ausschließlich zweidimensionale Einheitszellen erstellt. Zusammengesetzt werden diese Einheitszellen aus Splines. Die Splines sind dabei variabel definiert, sodass die Form der Einheitszelle mithilfe von Parametern beeinflusst wird. Ziel ist es, eine Gitterstruktur zu erstellen, die sich rein aus diesen sich wiederholenden Einheitszellen zusammensetzt.

Ein weiteres Ziel ist die Berechnung der mechanischen Eigenschaften der Einheitszellen. Es wird dafür eine Finite-Elemente-Berechnung durchgeführt. Diese ist so aufgebaut, dass periodische Randbedingungen aufgezwungen werden. Diese stellen sicher, dass die nebeneinander angeordneten Einzelzellen bei der Verformung der Gitterstruktur nicht überlappen oder auseinanderklaffen [19, 26].

Um den Rahmen dieser Diplomarbeit einzuhalten, wird nur eine reine Zugbelastung der Einheitszelle simuliert. Dieser Lastfall entspricht einem einachsigen Zugversuch. Im Allgemeinen weisen komplexe Einheitszellen jedoch anisotropes Verhalten auf [10, 13]. Die Bestimmung der mechanischen Eigenschaften mit einem Zugversuch ist eine Vereinfachung, die hier gewählt wurde.

Ein weiteres Ziel dieser Arbeit ist es, die gesamte Simulation mit frei zugänglicher Software, idealerweise Open-Source, durchzuführen. Die dafür genutzte Software sind zum einen die Python-Bibliotheken Splinepy und Gustaf für die Erstellung der Einheitszellen [17, 18]. Zum anderen wird MFEM, beziehungsweise der Python-Wrapper PyMFEM, zur Berechnung der mechanischen Eigenschaften der Einheitszelle genutzt [15].

Die Vorteile, die sich aus der Nutzung von Open-Source-Software ergeben, liegen einerseits darin, dass die Erstellung beliebiger Gitterstrukturen, die sich aus den hier erarbeiteten Einheitszellen aufbauen lassen, für alle möglich und frei zugänglich ist. Andererseits ist es vorteilhaft, dass sich die Schnittstelle der beiden Arbeitsteile (Erstellung der Einheitszelle und Berechnung der mechanischen Eigenschaften) vollständig automatisieren und sehr effizient gestalten lässt.

Bei der hier genutzten Open-Source-Software gibt es keine Benutzeroberfläche. Aus diesem Grund wird darauf geachtet, dass die benötigten Eingaben, die für eine erfolgreiche Erstellung der Gitterstruktur und die Berechnung der mechanischen Eigenschaften der einzelnen Einheitszellen erforderlich sind, so gering wie möglich gehalten werden.

## 2.1 Parametrisierung der Einheitszellen

Zur geometrischen Beschreibung der Einheitszelle kommen Splines zum Einsatz. Die Form dieser Splines soll durch Parameter gesteuert werden. Das wird erreicht, indem die formgebenden Kontrollpunkte der Splines nicht einzeln verändert, sondern durch gemeinsame Parameter verschoben werden. Dadurch wird die Definition dieser Splines wesentlich komplexer.

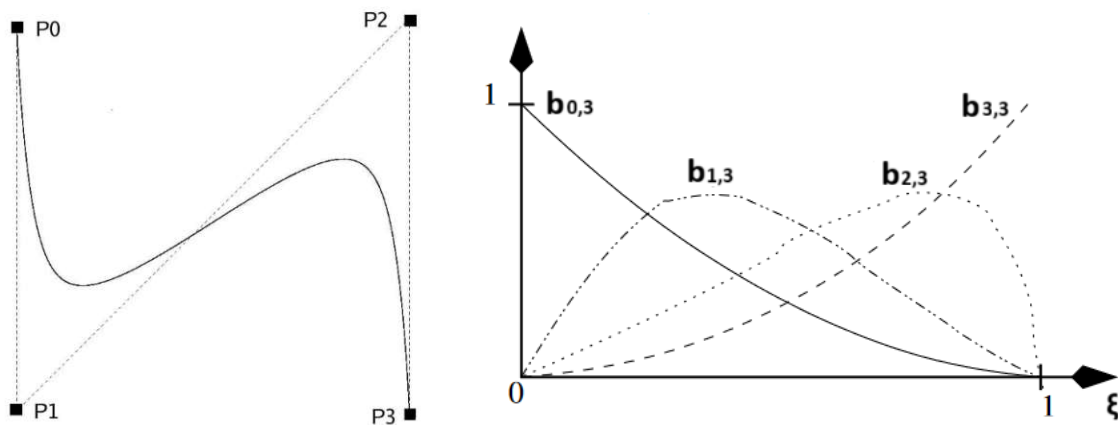
### 2.1.1 Spline-basierte Geometriebeschreibung

Splines sind ein grundlegendes mathematisches Werkzeug zur Erstellung von Geometrien. Die Geometrien können komplex und von beliebiger Dimension sein [23]. Die Ursprünge der Splines liegen in der Anwendung im Computer-Aided Design, kurz CAD [3]. Es gibt verschiedene Arten von Splines. In modernen CAD-Anwendungen werden NURBS (Non-Uniform Rational B-Splines) verwendet. Diese komplexere Definition eines Splines ermöglicht es zum Beispiel, auch Kreise darzustellen [23]. Die Darstellung eines Kreises ist mit weniger komplexen Splinearten wie B-Splines oder Bézier-Splines nicht exakt möglich.

Zurückzuführen ist der Ursprung der Splines auf den französischen Ingenieur Pierre Bézier [23]. Nach ihm ist die simpelste Art der Splines, die Bézier-Splines, benannt. Die Vorteile der Splines sind, dass glatte und beliebig stetige Geometrien erstellt werden können. Außerdem ist die Konstruktion von Objekten beliebig hoher Dimension und eingebetteter Geometrien, wie einer dreidimensionalen Fläche, möglich [3].

In der hier angewandten spline-basierten Geometriebeschreibung der Einheitszellen kommen ausschließlich Bézier-Splines zur Anwendung. Aus diesem Grund wird im Folgenden nur auf die Funktionsweise dieser Bézier-Splines eingegangen.

Um einen Bézier-Spline zu erstellen, ist die Definition von Kontrollpunkten erforderlich. Die Positionen dieser Kontrollpunkte definieren die Form des Splines [3]. In der Abbildung 2.1 (a) ist ein Bézier-Spline dargestellt (durchgezogene Linie), der mit vier Kontrollpunkten definiert wurde. Die gestrichelte Linie ist die Verbindungslinie zwischen den Kontrollpunkten und heißt Kontrollpolygon [3].



(a) Bézier-Spline mit Kontrollpunkten und Kontrollpolygon

(b) Basisfunktionen der Kontrollpunkte

Abbildung 2.1: Bézier-Spline dritten Grades [3]

Der Zusammenhang zwischen Kontrollpunktposition und Spline-Kurve wird durch die Bernsteinpolynome

$$b_{i,d}(\xi) = \frac{d!}{i!(d-i)!} \xi^i (1-\xi)^{d-i} \quad (2.1)$$

beschrieben. Dabei ist  $d$  der Grad des Bézier-Splines. Die Bernsteinpolynome  $b_{i,d}(\xi)$  sind im Fall der Bézier-Splines ihre Basisfunktionen [3]. Es gibt für jeden Kontrollpunkt eines Splines eine Basisfunktion, und  $i$  gibt an, zu welchem Kontrollpunkt das Bernsteinpolynom gehört.  $\xi$  ist der Splineparameter. Der Splineparameter ist ein über die Länge des Splines fortlaufender Wert, der von null bis eins über die gesamte Länge des Splines läuft. Im Beispiel in Abbildung 2.1 (a) wäre bei P0  $\xi = 0$  und bei P4  $\xi = 1$ . Durch diesen Parameter ist jeder Punkt auf dem Spline definiert [3].

In der Abbildung 2.1 (b) sind die Basisfunktionen der vier Kontrollpunkte aus Abbildung 2.1 (a) dargestellt. Mithilfe dieser Basisfunktionen wird die Position des Splines mit

$$\mathbf{S}(\xi) = \sum_{i=0}^d \mathbf{P}_i b_{i,d}(\xi) \quad (2.2)$$

berechnet, wobei  $\mathbf{P}_i$  die Koordinaten des Kontrollpunkts  $i$  enthalten. Der Bézier-Spline kann mit beliebigem Grad  $d$  erstellt werden. Eine Eigenschaft der Bézier-Splines ist, dass sie immer aus  $d + 1$  Kontrollpunkten aufgebaut werden [3].

Die resultierenden Eigenschaften von eindimensionalen Bézier-Splines sind:

- Ein Bézier-Spline ist immer in der konvexen Hülle der Kontrollpunkte enthalten.
- Alle Basisfunktionen sind über den gesamten Spline größer oder gleich null. Dadurch bewirkt eine Verschiebung eines Kontrollpunktes eine Formänderung im gesamten Bézier-Spline.
- Der Bézier-Spline berührt die äußeren Kontrollpunkte.
- An den äußeren Kontrollpunkten gleicht sich die Steigung des Splines der Steigung der Verbindungslinie zwischen dem äußersten und dem nächsten Kontrollpunkt an [30].

In dieser Arbeit wird mit zweidimensionalen Splines gearbeitet. Bei zweidimensionalen Splines wird zusätzlich zu  $\xi$  ein weiterer Splineparameter  $\eta$  eingeführt [3]. Dadurch wird eine Fläche aufgespannt, wobei der Grad in der zweiten Parameterrichtung beliebig und auch unterschiedlich zur ersten Parameterrichtung sein kann.

Das bedeutet, dass zum Aufbau eines 2D-Bézier-Splines nur die Koordinaten der Kontrollpunkte definiert werden müssen. Dabei ist es von der benötigten Komplexität der Formen abhängig, welchen Grad die einzelnen Splinedimensionen haben müssen. Aus dieser Dimension ergibt sich auch die Anzahl der Kontrollpunkte pro Raumrichtung des Bézier-Splines.

Die Basisfunktionen in  $\xi$ -Richtung werden auch im zweidimensionalen Fall wie in Gleichung 2.1 definiert. In  $\eta$ -Richtung sind die Basisfunktionen durch

$$b_{j,e}(\eta) = \frac{e!}{j!(e-j)!} \eta^j (1-\eta)^{e-j} \quad (2.3)$$

mit Bernsteinpolynomen definiert. Dabei ist  $e$  der Grad des Splines in  $\eta$ -Richtung und  $j$  ist die Kennzahl des zugehörigen Kontrollpunktes. Die von den Parametern  $\xi$  und  $\eta$  abhängige Position des Splines wird mit

$$\mathbf{S}(\xi, \eta) = \sum_{i=0}^d \sum_{j=0}^e \mathbf{P}_{i,j} b_{i,d}(\xi) b_{j,e}(\eta) \quad (2.4)$$

bestimmt [3].

Wie bereits erwähnt, spannen die beiden Splineparameter  $\xi$  und  $\eta$  die Fläche des zweidimensionalen Splines auf. Dabei wird zwischen zwei Räumen unterschieden, in denen diese Splineparameter definiert sind. Im Parameterraum wird ein Definitionsbereich der Parameter als Quadrat dargestellt [30]. In diesem Parameterraum sind die beiden Splineparameter orthogonal zueinander, und die Basisfunktionen werden in diesem Parameterraum definiert. In Abbildung 2.2 (a) ist dieser Parameterraum für einen zweidimensionalen Bézier-Spline mit  $d = 2$  und  $e = 1$  dargestellt. Dadurch ergeben sich in  $\xi$ -Richtung drei und in  $\eta$ -Richtung zwei Kontrollpunkte. Außerdem sind in dieser Abbildung die Basisfunktionen der beiden Splineparameter über die Fläche angedeutet.

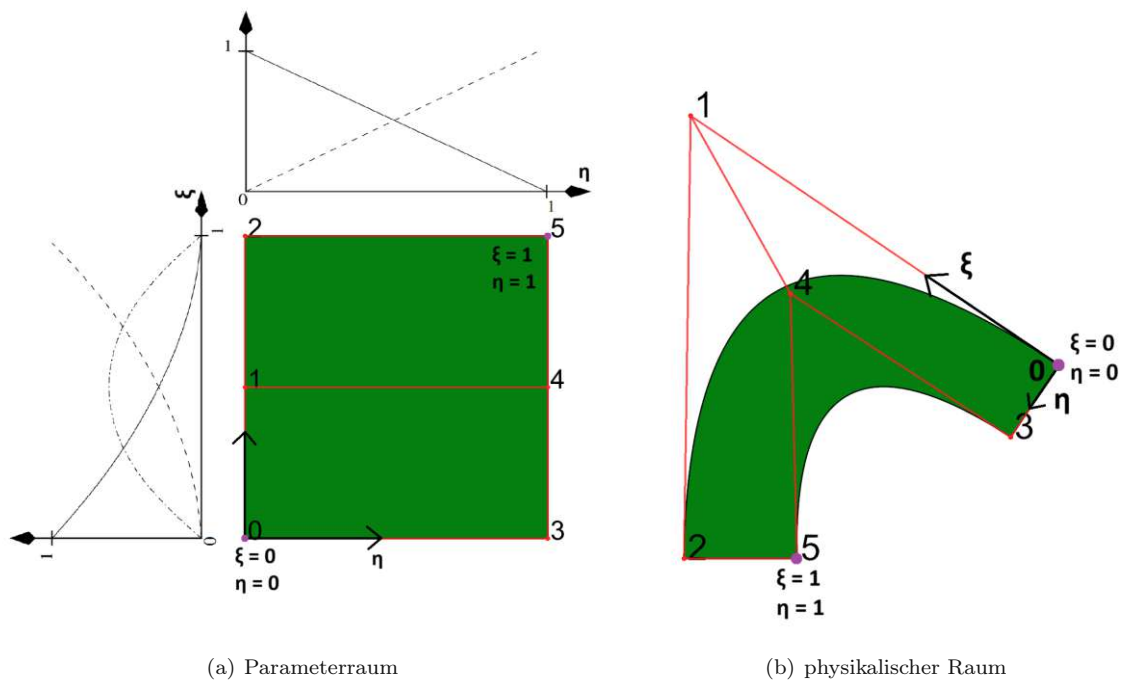


Abbildung 2.2: Abbildung eines zweidimensionalen Bézier-Splines vom Parameterraum in den physikalischen Raum [3]

In der Abbildung 2.2 (b) ist die durch die Basisfunktionen und die Kontrollpunkte definierte Bézier-Fläche dargestellt. Sie entsteht durch die Abbildung des Parameterraums in den physikalischen Raum mittels der Kontrollpunktpositionen. Diese Fläche befindet sich im physikalischen Raum. Die Form der Fläche des zweidimensionalen Bézier-Splines im physikalischen Raum ist dabei nur von der Position der sechs Kontrollpunkte abhängig.

Zur Erstellung komplexerer Geometrien werden mehrere Splines zusammengefügt. Dieser Zusammenschluss wird auch Multipatch oder im Deutschen Flächenverbund genannt [8].

Flächenverbunde kommen in Darstellungen in CAD-, FEM- und IGA-Anwendungen zum Einsatz [8]. Vor allem im CAD-Bereich ist die Nutzung von Flächenverbunden Standard. Durch das Zusammensetzen von Splines können komplexe Geometrien erstellt werden. Änderungen der Geometrie im Flächenverbund sind lokal, weil sich bei der Formänderung eines Splines im Verbund keiner der anderen Splines ändert. Ein weiterer Vorteil ist, dass die Parametrisierung sehr einfach ist. Jeder der Splines hat im zweidimensionalen Fall einen rechteckigen Parameterraum.

Die Nutzung des Flächenverbundes bringt auch Nachteile mit sich. Einer davon ist die stark eingeschränkte Stetigkeit an den Übergängen der Splines. Schwierigkeiten gibt es auch bei der Kopplung der Freiheitsgrade an den Übergängen zwischen den Splines. Dieser Nachteil ist jedoch für die Anwendung in dieser Arbeit nicht relevant, da der Flächenverbund in ein Finite-Elemente-Netz umgewandelt wird. Dort werden die verschiedenen Splines in ein durchgehendes Netz zusammengeführt [28].

Der Flächenverbund wird in dieser Arbeit eingesetzt, um mit mehreren Bézier-Splines die gewünschten Einheitszellen in benötigter Komplexität zu erstellen. In Abbildung 2.3 ist ein Beispiel einer dieser Einheitszellen abgebildet. Um die Grenzen zwischen den einzelnen Splines erkennbar zu machen, wurden die Splines unterschiedlich eingefärbt. Auch zu sehen sind in Rot die Kontrollpunkte und die Kontrollpolygone der einzelnen Splines.

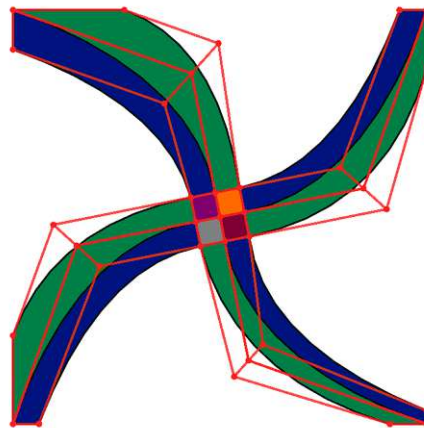
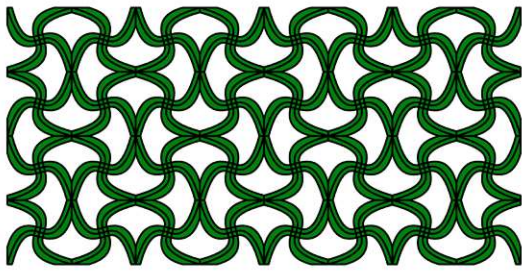
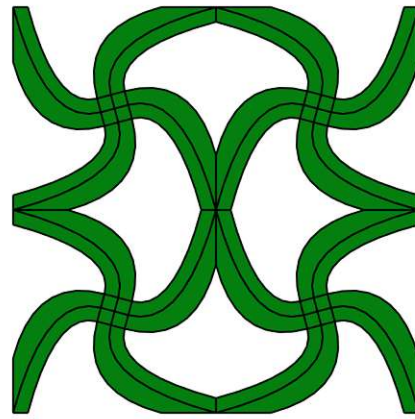


Abbildung 2.3: Einheitszelle als Flächenverbund aus Bézier-Splines

Mithilfe des Flächenverbundes werden Einheitszellen erstellt. Diese Einheitszellen werden zum Aufbau einer gesamten Gitterstruktur eingesetzt. Dazu werden die Einheitszellen nebeneinander angeordnet. So kann eine beliebig große Gitterstruktur aufgebaut werden. In Abbildung 2.4 (b) ist eine solche Gitterstruktur zu sehen, welche aus  $4 \times 2$  Einheitszellen aufgebaut wurde. In Abbildung 2.4 (a) ist die Einheitszelle zu sehen, die für den Aufbau der Gitterstruktur genutzt wurde. Diese Einheitszelle wurde im Laufe dieser Arbeit entworfen und wird später noch genauer erläutert.



(a) Gitterstruktur



(b) Einheitszelle

Abbildung 2.4: Eine aus Einheitszellen aufgebaute Gitterstruktur

Beim Aufbau einer Gitterstruktur, wie in Abbildung 2.4 (a), werden die Einheitszellen nacheinander erstellt und im Raum, in dem die Gitterstruktur aufgebaut wird, abgebildet. In Abbildung 2.4 (b) ist die Einheitszelle gezeigt, bevor sie in der Gitterstruktur abgebildet wird. Diese Einheitszelle heißt Referenzeinheitszelle. Diese Referenzeinheitszelle hat für jede Art der Einheitszellen gleiche Eigenschaften. Ihre Ausdehnung ist quadratisch und spannt sich zwischen den Koordinaten  $(0, 0)$  und  $(1, 1)$  auf. Dabei darf kein Teil des Flächenverbundes diese quadratischen räumlichen Grenze überschreiten. Diese Seitenlänge des Referenzelements wird im weiteren als Normlänge benannt.

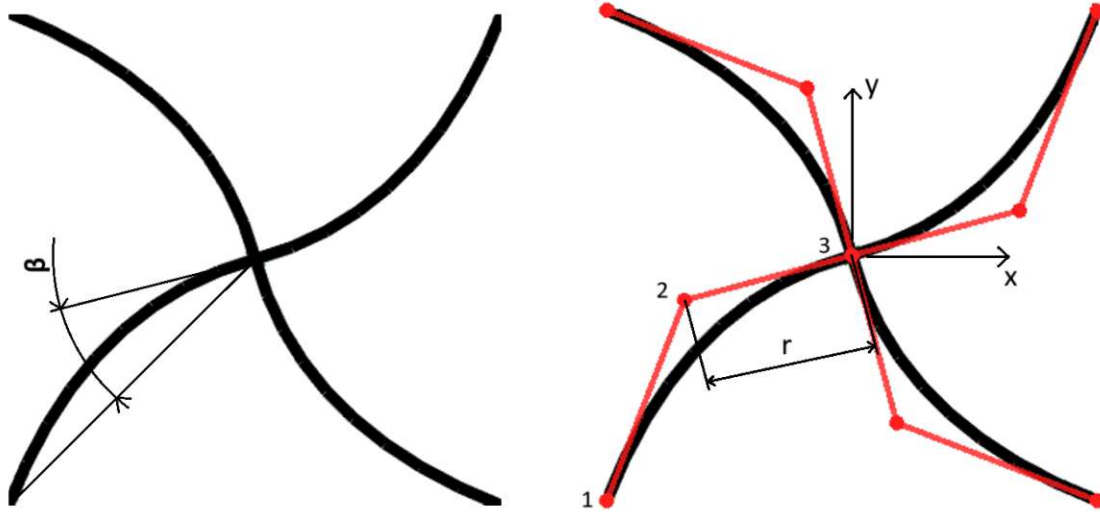
### 2.1.2 Parametrisierung der Einheitszelle

Der Begriff Parametrisierung bedeutet in diesem Kontext die Einflussnahme auf die Form der Einheitszelle durch die Änderung von definierten Parametern. Ziel ist es, den Flächenverbund der Einheitszelle in seiner Form so vielseitig wie möglich zu gestalten.

Dadurch wird eine große Bandbreite an mechanischen Eigenschaften erreicht. Das bedeutet, dass zum Beispiel die relative Dichte (nicht die Materialeigenschaft, sondern der Materialanteil in der Einheitszelle), die Steifigkeit oder die Poissonzahl beeinflussbar sind.

Die Parametrisierung geschieht parallel zum Aufbau der Gitterstruktur. Für jede Einheitszelle, die auf die Gitterstruktur abgebildet wird, wird die Form des Flächenverbundes durch die Parameter bestimmt.

Die Form der Splines wird dabei durch die Position der Kontrollpunkte definiert. Ein Parameter gibt eine Geometrie für den gesamten Flächenverbund vor. Diese Variation der Geometrie hat dabei Einfluss auf mehrere Splines im Flächenverbund. Um einen Parameter einzuführen, muss jeder Kontrollpunkt, der dadurch beeinflusst wird, eine entsprechend angepasste Position erhalten. Oft sind Kontrollpunktpositionen von mehreren Parametern abhängig. Dafür wurden Gleichungen zur Positionsbestimmung der Kontrollpunkte erarbeitet, welche alle Einflüsse der verschiedenen Parameter auf die Kontrollpunktposition berücksichtigen.



(a) Einheitszelle mit bemaßtem Parameter

(b) Einheitszelle mit bemaßtem Kontrollpolygon

Abbildung 2.5: Eindimensionale Einheitszelle

In der Abbildung 2.5 ist ein Beispiel zur Parametrisierung einer Einheitszelle dargestellt. In der dargestellten eindimensionalen Einheitszelle wird, wie in Abbildung 2.5 (a) bemaßt, der Winkel  $\beta$  als Parameter festgelegt. Um diesen Winkel zu variieren, muss der Kontrollpunkt 2 in Abbildung 2.5 (b) für jeden der vier Arme verschoben werden. Für diese Einheitszelle wurde eine kreisförmige Bewegung des Kontrollpunkts um den Mittelpunkt 3 mit Radius  $r$  definiert. Daraus ergeben sich die Positionsgleichungen

$$P_{2x} = -r \cdot \cos\left(\beta + \frac{\pi}{4}\right)$$

$$P_{2y} = -r \cdot \sin\left(\beta + \frac{\pi}{4}\right)$$

für den Punkt 2. In dieser Art werden bei Einheitszellen mit mehreren Parametern die Einflüsse dieser auf die Kontrollpunktpositionen, Parameter für Parameter, eingearbeitet. So können auch komplexe Flächenverbunde mit mehreren Parametern erstellt werden.

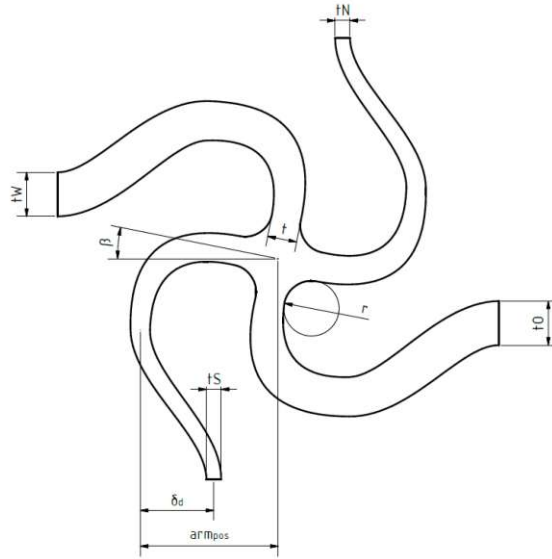


Abbildung 2.6: Flächenverbund einer Einheitszelle mit bemaßten Parametern

Abbildung 2.6 zeigt eine komplexere Einheitszelle, bei der alle Parameter bemaßt sind. Das bedeutet, dass jeder dieser 9 Parameter beliebig gewählt werden kann, solange sich die gesamte Geometrie des Flächenverbundes innerhalb der Grenze der Einheitszelle befindet.

Bei der Erstellung einer Einheitszelle, wie in Abbildung 2.6 dargestellt, sind viele der Kontrollpunkte von mehreren Parametern abhängig. Durch mehr Abhängigkeiten von Parametern werden die Gleichungen, die die Position dieser Kontrollpunkte bestimmen, immer komplexer. Die Einheitszelle ist dabei dieselbe wie in Abbildung 2.3. Dort sind die Kontrollpunkte für die entsprechenden Splines eingezeichnet.

Die Einheitszellen werden mit Parametern erstellt und nacheinander in die Gitterstruktur abgebildet. Dadurch ist es möglich, jeder Einheitszelle in der Gitterstruktur unterschiedliche Parameter zuzuweisen. So können Gitterstrukturen erstellt werden, welche lokal unterschiedliche mechanische Eigenschaften aufweisen. In Abbildung 2.7 ist eine Gitterstruktur mit über die Fläche variierenden Parametern dargestellt. Auf der Fläche der Gitterstruktur sind die Werte der Parameter festgelegt. Wenn eine Einheitszelle eine Position in der Gitterstruktur einnehmen soll, werden die Parameter von der vorbestimmten Platzierung in der Gitterstruktur eingelesen.

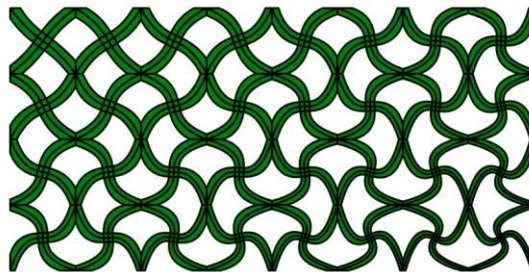


Abbildung 2.7: Gitterstruktur mit variablen Parametern

Der Aufbau der Gitterstruktur wird mithilfe der Open-Source-Python-Bibliothek Splinepy durchgeführt. Splinepy ist eine Software zur Erstellung von Splines. Der für die Erstellung der Einheitszelle benötigte

Flächenverbund kann in Splinepy nur mit Bézier-Splines erstellt werden. Darum werden keine komplexeren Splinearten zur Geometrieerstellung eingesetzt. Die Einheitszellen werden definiert und mit beliebigen Parametern zu einer Gitterstruktur zusammengesetzt. Auch das Abbilden der Einheitszellen in die Gitterstruktur wird mithilfe von Splinepy durchgeführt.

### 2.1.3 Erstellen des Finite-Elemente-Netzes im Multipatch

In dieser Arbeit wird die Simulation mit der Finite-Elemente-Methode durchgeführt. Die Gitterstruktur liegt jedoch in Form einzelner Splines vor, die die gewünschte Geometrie darstellen. Um die mechanischen Eigenschaften der Gitterstruktur zu berechnen, ist es nötig die Splinegeometrie der Gitterstruktur in ein Finite-Elemente-Netz zu übersetzen.

Beim Erstellen eines Finite-Elemente-Netzes aus Splines kann deren parametrische Struktur genutzt werden. Es wird entsprechend einer wählbaren Netzfeinheit ein Netz aus Knotenpunkten erstellt. Diese Knotenpunkte haben in Bezug auf die Splineparameter immer gleiche Abstände zueinander. Dieser Vorgang wird für jeden Parameter jedes Bézier-Splines mit immer gleicher Netzfeinheit durchgeführt.

Wenn eine Netzfeinheit von 3 Knotenpunkten pro Splineparameter gefordert ist, wird über den Splineparameter  $\xi$  ein Knotenpunkt des Finite-Elemente-Netzes bei  $\xi = 0$ ,  $\xi = 0.5$  und  $\xi = 1$  gesetzt. Selbiges gilt auch für die zweite Raumrichtung mit Splineparameter  $\eta$ . Dadurch entstehen 9 Knotenpunkte im Spline. Es werden aus diesen Punkten vier viereckige Finite-Elemente erstellt (siehe Abbildung 2.8).

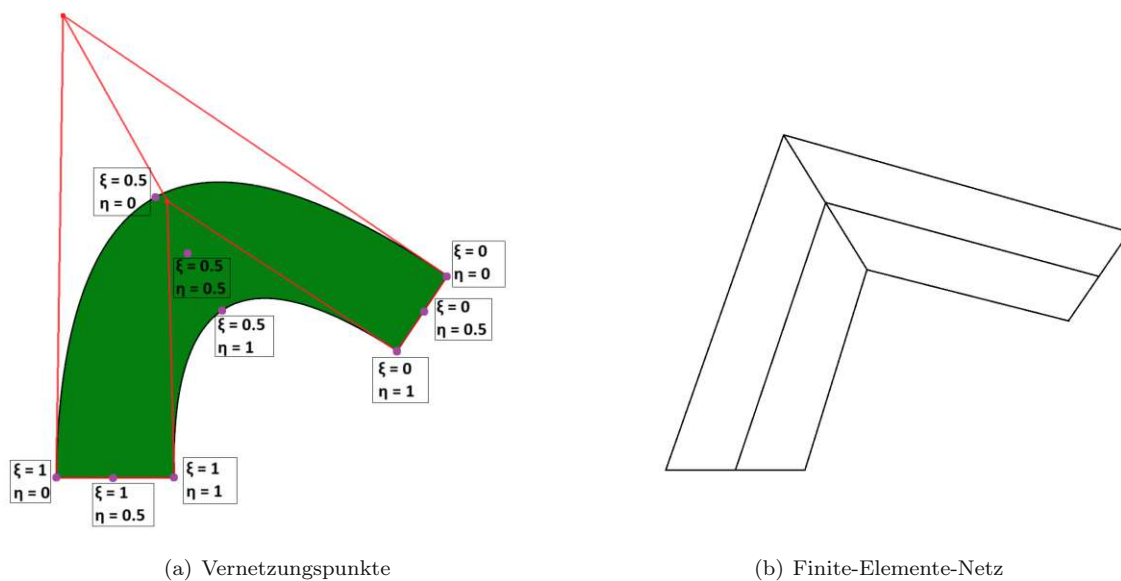


Abbildung 2.8: Zweidimensionaler Spline mit Vernetzungspunkten aus den Splineparametern und Finite-Elemente-Netz

In der Abbildung 2.8 (a) ist ein zweidimensionaler Bézier-Spline zu sehen, mit eingezeichneten Knotenpunkten, welche mithilfe der Splineparameter gesetzt wurden. Diese Knotenpunkte werden zum Vernetzen verwendet. In der Abbildung 2.8 (b) ist das Netz abgebildet, welches durch das Verbinden der erstellten Knotenpunkte entsteht.

Dieser Vorgang ist in Splinepy durch die Funktion `splinepy.helpme.extract.faces` implementiert [18]. Dieses Werkzeug, das Splinepy uns zur Verfügung stellt, ist sehr einfach und bietet große Vorteile, hat jedoch auch einige Einschränkungen. Auch wenn jeder Spline einzeln vernetzt wird, gibt es keine Möglichkeit,

die Feinheit der Unterteilung individuell einzustellen. Das bedeutet, dass jeder Splineparameter im Multipatch in gleich viele Teile geteilt wird. Dadurch weist jeder Spline die gleiche Anzahl an Finite-Elementen auf. Das bedeutet, dass kleinere Splines im Multipatch deutlich feiner vernetzt werden als größere.

Auch Splines mit sehr ungleichen Seitenverhältnissen sind mit länglichen Elementen vernetzt. Ein Vorteil dieses Vorgehens ist, dass, wenn die Splines an ihren Übergängen immer vollständig zueinander abschließen (die Eckknotenpunkte der Splines an der aneinanderliegenden Außenkante übereinstimmen), auch das Netz an diesen Übergängen exakt zusammenpasst. Weiters erhält man immer ein strukturiertes Netz. Diese Arbeit beschränkt sich auf die Nutzung dieses Splinepy-Befehls. Eine potenzielle Erweiterung wäre, in Splinepy die Möglichkeit zu schaffen, diese Unterteilung der Splineparameter individuell festzulegen.

In Abbildung 2.1.3 ist eine vernetzte Einheitszelle dargestellt. Dort ist gut zu sehen, dass in den kleineren Splines bei den Rundungen an den Übergängen zwischen Armen und Mittelteilen das Netz viel feiner ist. Jedoch ist die Konnektivität überall gegeben.

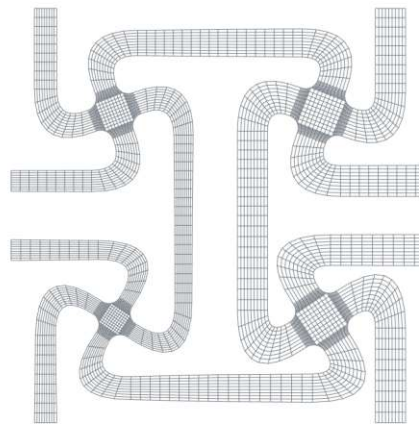


Abbildung 2.9: Finite-Elemente-Netz der Einheitszelle

Im Zuge der Erstellung des Finite-Elemente-Netzes werden auch Knotenpunkte erstellt. Diese befinden sich überall dort, wo Netzlinien aufeinandertreffen oder, in anderen Worten, immer an den Ecken der Viereckelemente.

An diesen Knotenpunkten werden im weiteren Verlauf die Randbedingungen definiert. Dieses Definieren der Randbedingungen soll, unabhängig von der Form der Einheitszelle, vollautomatisch ablaufen. Dazu werden nach der Netzerstellung die Knoten, die innerhalb einer numerischen Abweichung am Rand liegen, nach ihrer Ausrichtung (Norden, Osten, Süden, Westen) gruppiert und als Randknotenpunkte im Netz zusammengefasst. Bei der Simulation werden diese Knotenpunktgruppen dann aufgerufen und mit der gewünschten Randbedingung verknüpft.

Wenn das Netz erstellt ist und die betroffenen Knotenpunkte zu Randknotenpunktgruppen zusammengefügt wurden, folgt der Export dieser Informationen. Dabei gibt es in Splinepy die Möglichkeit, direkt eine Netzdatei auszugeben, die in MFEM eingelesen wird. Die Schnittstelle zwischen Splinepy und MFEM ist die Übergabe der Netzdatei. Diese Schnittstelle ist sehr effizient und ermöglicht somit eine vollautomatisierte Gesamtlösung von der Erstellung der Einheitszelle oder auch einer Gitterstruktur durch die Eingabe weniger Parameter mit direkter Berechnung der mechanischen Eigenschaften der erstellten Einheitszelle [3, 11, 12].

## 2.2 Berechnung der mechanischen Eigenschaften

Im folgenden Teil wird auf die Simulation eingegangen. Diese wird mit der Open-Source-Python-Bibliothek MFEM durchgeführt. Die Netzdatei wurde bereits erstellt und wird als Grundlage der Simulation genutzt.

Es wird beschrieben, wie die Finite-Elemente-Simulation aufgebaut ist und wie die Randbedingungen aufgebracht werden. Nachdem in der Simulation die Verschiebungen der Knotenpunkte berechnet wurden, werden im Postprocessing die Daten so weiterverarbeitet, dass die gewünschten Kennzahlen der mechanischen Eigenschaften ausgegeben werden. Die gesuchten Eigenschaften sind Poissonzahl, maximale Spannung und benötigte Kraft für die Deformation.

### 2.2.1 Finite-Elemente-Simulation

Zur Berechnung der mechanischen Eigenschaften wird eine Finite-Elemente-Simulation (FE-Simulation) durchgeführt [7]. Im Kapitel 2.1 wurde bereits die Erstellung der Geometrie und die Vernetzung dieser beschrieben. Dieses Rechenetz wird in der FE-Simulation für die Berechnung der mechanischen Eigenschaften der Einheitszelle herangezogen. In diesem Kapitel wird auf die Schritte einer FE-Simulation eingegangen.

#### 2.2.1.1 Gleichungen der Kontinuumsmechanik im linear elastischen Fall

Ausgangspunkt der Herleitung ist das Prinzip des statischen Kräftegleichgewichts

$$\int_{\partial\Omega} \boldsymbol{\sigma} \cdot \mathbf{n} dA + \int_{\Omega} \mathbf{f} dV = \mathbf{0} \quad (2.5)$$

im infinitesimalen Volumenelement. Dabei ist  $\boldsymbol{\sigma}$  der Spannungstensor zweiter Stufe und  $\mathbf{f}$  der Vektor der Volumskräfte.  $\Omega$  beschreibt das Rechengebiet und  $\partial\Omega$  die Oberfläche dieses Rechengebiets. Dabei ist  $\mathbf{n}$  der Normalvektor zur Oberfläche des Rechengebiets. Das Flächenintegral wird mithilfe der Gauß-Quadratur in das Volumenintegral

$$\int_{\Omega} \nabla \cdot \boldsymbol{\sigma} dV + \int_{\Omega} \mathbf{f} dV = \mathbf{0} \Rightarrow \int_{\Omega} (\nabla \cdot \boldsymbol{\sigma} + \mathbf{f}) dV = \mathbf{0} \quad (2.6)$$

umgewandelt. Dieser Zusammenhang gilt für jedes Volumen. Punktweise gilt

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{f} = \mathbf{0} \quad \text{in } \Omega \quad (2.7)$$

was in der Regel als

$$-\nabla \cdot \boldsymbol{\sigma} = \mathbf{f} \quad \text{in } \Omega \quad (2.8)$$

dargestellt wird. Diese Form der Gleichgewichtsgleichung ist die starke Form der partiellen Differentialgleichung, die mit der FE-Methode in weiterer Folge gelöst wird [7]. Weiters wird ein Konstitutivgesetz benötigt. Für linear elastisches Materialverhalten gilt die allgemeine Form des Hooke'schen-Gesetzes

$$\boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{u}) \quad (2.9)$$

wobei  $\mathbb{C}$  der Steifigkeitstensor vierter Stufe und  $\boldsymbol{\varepsilon}$  der Verzerrungstensor zweiter Stufe sind [7].  $\mathbf{u}$  ist das kontinuierliche Verschiebungsfeld. Es kann unter der hier getroffenen Annahme, dass nur kleine Verschiebungen auftreten, der lineare Zusammenhang zwischen Verzerrungstensor und der Verschiebung als

$$\boldsymbol{\varepsilon}(\mathbf{u}) = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^\top) \quad (2.10)$$

definiert werden [7].

### 2.2.1.2 Bildung der schwachen Form der Differentialgleichung

Die starke Form der partiellen Differentialgleichung ist in Gleichung 2.8 hergeleitet worden. In dieser Form wird  $\boldsymbol{\sigma}$  einmal abgeleitet. Aus den Gleichungen 2.9 und 2.10 zeigt sich, dass die Verschiebung  $\mathbf{u}$  zweifach differenzierbar sein muss [6].

Die Einführung der schwachen Form der Differentialgleichung hat verschiedene Vorteile. Es wird sich zeigen, dass die Anforderungen an die Differenzierbarkeit von  $\mathbf{u}$  verringert werden, wodurch auch realistische (nicht-glatte Funktionen) betrachtet werden können. Durch die Formulierung der schwachen Lösung ist es möglich, die Differentialgleichung auf ein Variationsproblem herunterzubrechen. Eine Lösung des Variationsproblems entspricht einer Minimierung eines Energie-Funktional, was in der Finite-Elemente-Methode genutzt wird [6].

Um die schwache Form zu bilden, wird die Gleichung 2.9 mit einer Testfunktion  $\mathbf{v}$  multipliziert und über das gesamte Volumen  $\Omega$  integriert. Die Testfunktion ist ein Vektorfeld und soll im Raum der unendlich oft stetig differenzierbaren Funktionen liegen. Zusätzlich ist  $\mathbf{v}$  am Rand mit Dirichlet-Randbedingungen  $\Gamma_D$  unseres Volumens  $\Omega$  gleich null. Dadurch folgt die Gleichung

$$\int_{\Omega} (-\nabla \cdot \boldsymbol{\sigma}) \cdot \mathbf{v} dV = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} dV \quad (2.11)$$

in der  $\mathbf{v}$  ein Vektorfeld und  $\boldsymbol{\sigma}$  ein Tensorfeld ist [6, 21]. Die partielle Integration der linken Seite führt zu

$$\int_{\Omega} \boldsymbol{\sigma} : \nabla \mathbf{v} dV - \int_{\Omega} \nabla \cdot (\boldsymbol{\sigma} \mathbf{v}) dV = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} dV \quad (2.12)$$

was mit der Anwendung der Gauß-Quadratur auf  $\int_{\Omega} \nabla \cdot (\boldsymbol{\sigma} \mathbf{v}) dV$  die Gleichung

$$\int_{\Omega} \boldsymbol{\sigma} : \nabla \mathbf{v} dV - \int_{\partial\Omega} (\boldsymbol{\sigma} \cdot \mathbf{n}) \cdot \mathbf{v} ds = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} dV \quad (2.13)$$

ergibt. Im nächsten Schritt werden die Randbedingungen definiert und in der Gleichung angewendet. Wie bereits beschrieben, ist eine der Eigenschaften der Testfunktion  $\mathbf{v}$ , dass sie am Dirichlet-Rand  $\Gamma_D$  verschwindet. Es soll am Neumann-Rand  $\Gamma_N$  der Zusammenhang  $\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{t}_N$  gelten, wobei  $\mathbf{t}_N$  der Neumann-Randwert ist. Wenn diese Definitionen der Randbedingungen angewendet werden, ergibt sich

$$\int_{\Omega} \boldsymbol{\sigma} : \nabla \mathbf{v} dV = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} dV + \int_{\Gamma_N} \mathbf{t}_N \cdot \mathbf{v} ds \quad (2.14)$$

als Gleichung. In diese Gleichung wird die Gleichung 2.9 eingesetzt [6,21]. Mithilfe der Symmetrie von  $\mathbb{C}$  wird der Ausdruck  $\nabla \mathbf{v}$  durch  $\boldsymbol{\varepsilon}(\mathbf{v})$  ersetzt, und man erhält

$$\int_{\Omega} \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) dV = \int_{\Omega} \mathbf{f} \cdot \mathbf{v} dV + \int_{\Gamma_N} \mathbf{t}_N \cdot \mathbf{v} ds \quad (2.15)$$

als schwache Form der Differentialgleichung. Durch das Einführen der Testfunktion wird  $\mathbf{u}$  einmal weniger abgeleitet als in der starken Form. Das bedeutet, dass durch die Umformung auf die schwache Form die benötigte Ableitbarkeit von  $\mathbf{u}$  auf einfach differenzierbar verringert wurde. Dadurch konnte der Raum der möglichen Lösungen um die Funktionen, die einen Knick aufweisen, erweitert werden [6].

### 2.2.1.3 Finite-Elemente-Diskretisierung

Die Herleitung der nachfolgenden Gleichungen sowie die verwendeten Definitionen orientieren sich an [7]. Da wir nun die schwache Form der Differentialgleichung haben, muss diese noch in einer Form diskretisiert werden, sodass sie mithilfe computergestützter Rechenmethoden lösbar oder approximierbar wird. Computergestützte Rechenmethoden erfordern eine Diskretisierung der schwachen Form der Differentialgleichung. Mithilfe dieser diskretisierten Form wird eine approximierte Lösung der Differentialgleichung berechnet. Die Differentialgleichung wird dafür durch ein Gleichungssystem aus algebraischen Gleichungen dargestellt. Der erste Schritt hierfür ist die Diskretisierung des Rechengebiets. Diese Diskretisierung geschieht, wie in Kapitel 2.1.3 bereits beschrieben, durch die Vernetzung der Splines, die die Einheitszelle aufbauen.

Ein Hauptaspekt der Diskretisierung des Rechengebiets ist in unserem Fall die Unterteilung der Fläche in rechteckige Elemente. Diese Elemente haben als Ursprung alle dasselbe Referenzelement, welches verzerrt, verdreht und verschoben wird. Dazu wird das Referenzelement in globale Koordinaten abgebildet. Dieses Referenzelement ist für ein rechteckiges lineares Element in Abbildung 2.10 dargestellt.

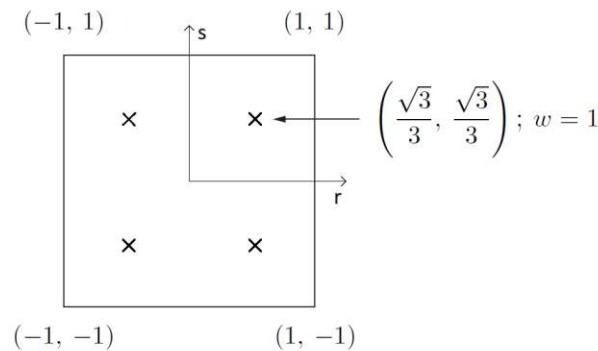


Abbildung 2.10: Referenzelement für rechteckige lineare Elemente [7]

Jedes Finite-Elemente-Netz besteht aus Knotenpunkten, und jedem dieser Knotenpunkte wird eine Ansatzfunktion  $N$  zugeteilt. Die Form dieser Ansatzfunktionen ist abhängig vom Grad der gewünschten Elemente im Finite-Elemente-Netz. Im einfachen Beispiel eines linearen rechteckigen Elements sehen die Ansatzfunktionen, wie in Abbildung 2.11 dargestellt, aus und haben folgende Eigenschaften:

- Haben den Wert 1 an der Stelle des dazugehörigen Knotenpunkts
- Haben den Wert 0 in allen anderen Knotenpunkten

- Ansatzfunktionen sind vom Grad, den das Element vorgibt (hier linear)
- Sie sind bei linearen Basisfunktionen nur in den angrenzenden Elementen ungleich null

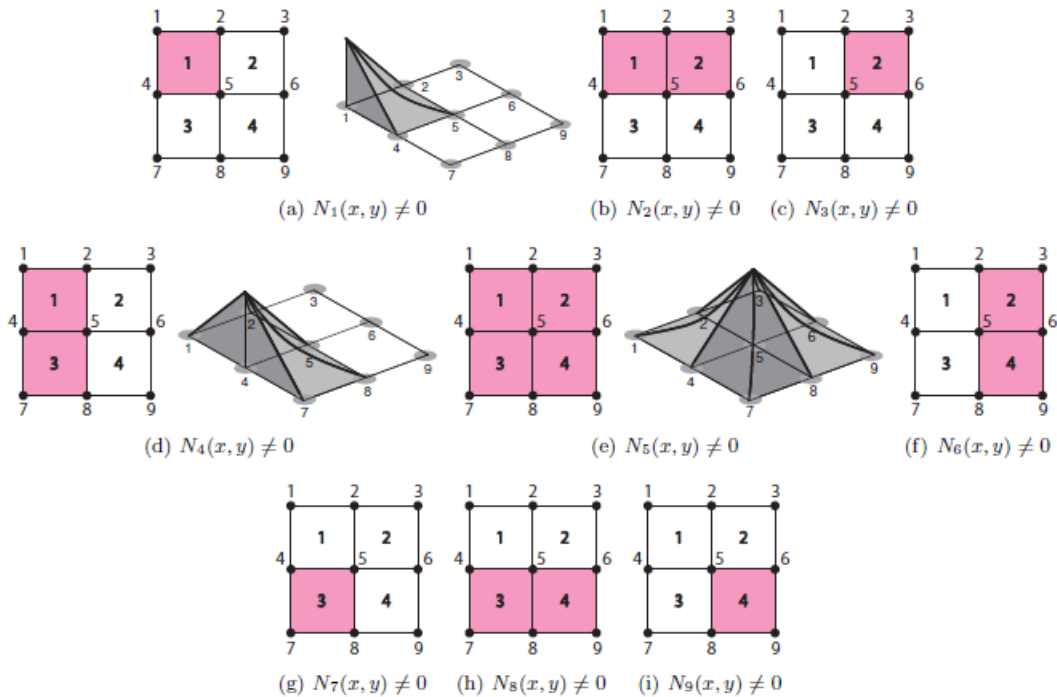


Abbildung 2.11: Ansatzfunktionen für vier lineare rechteckige Elemente [7]

Mithilfe der Ansatzfunktionen ist an jeder Stelle im Element die Verschiebung  $\mathbf{u}$  berechenbar. Diese Berechnung wird mit der Gleichung

$$\mathbf{u}(x, y) = \sum_{i=1}^n \mathbf{N}_i(x, y) \cdot \mathbf{u}_{k,i} \quad (2.16)$$

durchgeführt.  $\mathbf{N}_i(x, y)$  sind die Ansatzfunktionen des Knotenpunktes  $i$ ,  $\mathbf{u}_{k,i}$  die Verschiebung am Knotenpunkt  $i$  und  $n$  die Anzahl der Knotenpunkte [7]. Der Unterschied zwischen  $\mathbf{u}$  und  $\mathbf{u}_{k,i}$  liegt darin, dass  $\mathbf{u}_{k,i}$  sich ausschließlich auf die Verschiebung des Knotenpunktes  $i$  bezieht.  $\mathbf{u}$  dagegen kann an jeder Stelle der Geometrie ausgewertet werden. Streng genommen ist  $\mathbf{u}$  in diesem Kontext nur eine Näherungslösung und ist dadurch nicht exakt gleich dem  $\mathbf{u}$ , das in den Kapiteln 2.2.1.1 und 2.2.1.2 verwendet wird. Im weiteren Verlauf dieser Arbeit wird die diskretisierte Form von  $\mathbf{u}$  der Einfachheit halber ebenfalls mit  $\mathbf{u}$  bezeichnet. Der nächste Schritt ist die Diskretisierung der schwachen Form der Differentialgleichung 2.15. Dafür ist in

$$\mathbf{K} \cdot \mathbf{u}_k = \mathbf{f} \quad (2.17)$$

das lineare Gleichungssystem gegeben, welches die Näherungslösung der Differentialgleichung darstellt [7]. Der Vektor  $\mathbf{u}_k$  ist hierbei der Lösungsvektor und enthält alle Verschiebungen der Knotenpunkte.  $\mathbf{K}$  ist die Steifigkeitsmatrix oder Systemmatrix und wird im Folgenden zusammengesetzt.  $\mathbf{f}$  ist der Kräftevektor und wird abhängig von den Randbedingungen befüllt.

Der Zusammenhang zwischen der schwachen Form der Differentialgleichung 2.15 und dem Gleichungssystem 2.17 wird nun schrittweise untersucht.

$\mathbf{K}$  hat ihren Ursprung im Integral  $\int_{\Omega} \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{u}) : \boldsymbol{\varepsilon}(\mathbf{v}) dV$ . Dieses Integral wird umgestellt, sodass es die Form  $\int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{u}) : \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{v}) dV$  erhält.

Wenn nun in die Gleichung 2.10 die diskretisierte Form der Verschiebung aus Gleichung 2.16 eingesetzt und abgeleitet wird, wird  $\boldsymbol{\varepsilon}(\mathbf{u})$  in der diskretisierten Form

$$\boldsymbol{\varepsilon} = \mathbf{B} \cdot \mathbf{u}_k \quad (2.18)$$

definiert. Dabei ist  $\mathbf{B}$  die Dehnungs-Verschiebungs-Matrix und stellt den Zusammenhang zwischen den nodalen Verschiebungen und den Verzerrungen her.  $\mathbf{B}$  ist hierbei an jeder Stelle des Elements auswertbar. Nach dem Einsetzen dieser diskretisierten Definition von  $\boldsymbol{\varepsilon}$  in das Integral  $\int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{u}) : \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{v}) dV$  folgt

$$\int_{\Omega} \boldsymbol{\varepsilon}(\mathbf{u}) : \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{v}) dV = \int_{\Omega} (\mathbf{B}^T \mathbb{C} \mathbf{B}) dV \cdot \mathbf{u}_k = \mathbf{K} \cdot \mathbf{u}_k \Rightarrow \mathbf{K} = \int_{\Omega} (\mathbf{B}^T \mathbb{C} \mathbf{B}) dV \quad (2.19)$$

mit der Elastizitätsmatrix  $\mathbb{C}$ , die aus den Materialkennwerten berechnet wird und ein Tensor vierter Stufe ist [7]. Durch die Symmetrie der Elastizitätsmatrix lässt diese sich für die Erstellung des Gleichungssystems stark vereinfachen. In dieser Arbeit werden zusätzlich nur zweidimensionale Simulationen durchgeführt. Dadurch wird diese Matrix nochmals vereinfacht, was im Kapitel 2.2.1.4 nochmals genauer erläutert wird. Das auftretende Integral in Gleichung 2.19 muss numerisch gelöst werden. Eine Möglichkeit ist die Gauß-Quadratur. Das Integral wird hierbei durch die Summe

$$\mathbf{K}^e = \sum_{i=1}^{IP} \mathbf{B}_i^T \mathbb{C} \mathbf{B}_i \cdot \det(\mathbf{J}_i) \cdot w_i \quad (2.20)$$

approximiert.  $\mathbf{K}^e$  ist die Steifigkeitsmatrix, welche sich nur auf das Element  $e$  bezieht.  $\mathbf{J}_i$  und  $w_i$  sind die Jacobi-Matrix und das Gewicht bezogen auf den Integrationspunkt  $i$  [7]. Bei der Gauß-Quadratur werden die Ansatzfunktionen abgeleitet und an den Integrationspunkten ausgewertet. Die Integrationspunkte (und die zugehörigen Gewichte  $w$ ) sind im Referenzelement, wie in Abbildung 2.10 dargestellt, verteilt. Die Dehnungs-Verschiebungs-Matrix  $\mathbf{B}$  wird an den Stellen der Integrationspunkte ausgewertet und in Gleichung 2.20 eingesetzt. Der Elastizitätstensor  $\mathbb{C}$  ist in unserem Fall konstant, da ein linear elastisches, homogenes und isotropes Materialverhalten angenommen wird. Im Ausdruck  $\det(\mathbf{J}_i)$  ist  $\mathbf{J}_i$  die Jacobi-Matrix der Abbildung vom Referenzelement in das globale Element am Integrationspunkt.

$\mathbf{K}^e$  ist die Steifigkeitsmatrix, welche sich nur auf ein Element bezieht. Dieses  $\mathbf{K}^e$  wird für jedes Element berechnet und zur gesamten Steifigkeitsmatrix  $\mathbf{K}$  des Systems zusammengesetzt. Die Steifigkeitsmatrix ist dabei in der Regel nur dünn besetzt und ist dadurch sehr effizient lösbar.

Der Lastvektor  $\mathbf{f}^e$  wird aus den beiden Integralen auf der rechten Seite der Gleichung 2.15 gebildet. Dadurch ergibt sich  $\mathbf{f}^e$  als Lastvektor für das Element  $e$  zu

$$\mathbf{f}^e = \int_{\Omega} \mathbf{N}^T \mathbf{f} d\Omega + \int_{\Gamma_N} \mathbf{N}^T \mathbf{t} d\Gamma \quad (2.21)$$

worin das Volumenintegral den Volumskräften und das Flächenintegral den Flächenkräften entspricht [7]. Da in dieser Arbeit keine Volumskräfte auftreten, werden hier nur die Flächenkräfte diskretisiert. Im bei uns vorliegenden zweidimensionalen Fall sind die Flächenkräfte nur Kräfte, die auf einen Rand wirken. Wenn im Referenzelement (siehe Abbildung 2.10) zum Beispiel der untere Rand ( $s = -1$ ,  $r \in [-1, 1]$ ) mit einer Kraft belastet wird, ergibt sich das Integral

$$\mathbf{f}^e = \int_{-1}^1 \mathbf{N}^T(r) \cdot \bar{\mathbf{t}} \cdot |J_{1D}(r)| dr \quad (2.22)$$

welches wieder mithilfe der Gauß-Quadratur auf jeden belasteten Rand angewandt wird. Im Unterschied zur Berechnung von  $\mathbf{K}^e$  wird ein Linienintegral berechnet. Im Fall der Belastung der unteren Grenze ist  $s = -1$  fix, und im Bereich  $r \in [-1, 1]$  liegen zwei Integrationspunkte auf  $r = -\frac{1}{\sqrt{3}}$  und  $r = \frac{1}{\sqrt{3}}$ . Durch Anwendung des Integrationsssatzes folgt die diskretisierte Form

$$\mathbf{f}^e = \sum_{i=1}^2 \mathbf{N}^T(r_i, s = -1) \cdot \mathbf{t} \cdot |J_{1D}(r_i)| \cdot w_i \quad (2.23)$$

die für jeden belasteten Rand berechnet wird. Anschließend werden die einzelnen  $\mathbf{f}^e$  zum  $\mathbf{f}$  des Gesamtsystems zusammengesetzt.

Falls die Krafteinwirkung nicht entlang einer Kante, sondern direkt an einem Knotenpunkt angreift, vereinfacht sich das Bilden von  $\mathbf{f}^e$ . Wie der Lösungsvektor  $\mathbf{u}_k$  ist der Kraftvektor  $\mathbf{f}$  nach Knotenpunkten und Raumrichtungen aufgeteilt. Wirkt am Knotenpunkt  $i$  eine Kraft in  $x$ -Richtung, wird der Betrag dieser Kraft einfach an die entsprechende Stelle im Kraftvektor eingetragen.

Durch das Miteinbeziehen der Kräfte in das Gleichungssystem wurden hiermit bereits die Neumann-Randbedingungen berücksichtigt. Als letzten Schritt müssen die Dirichlet-Randbedingungen berücksichtigt werden.

Bei den Dirichlet Randbedingungen handelt es sich um gesperrte Freiheitsgrade der Knotenpunkte oder um vorgegebene Verschiebungen dieser. Während die Neumann-Randbedingungen im Vektor  $\mathbf{f}$  Platz finden, muss die Steifigkeitsmatrix  $\mathbf{K}$  für die Dirichlet-Randbedingung bearbeitet werden. Dort werden an den Knotenpunkten, die eine Randbedingung erhalten, die Einträge auf Null gesetzt und mit den gewünschten Bedingungen befüllt. Genauer dazu noch in Kapitel 2.2.1.4 [7].

#### 2.2.1.4 Beispiel zum Aufbau des Gleichungssystems

Zur Darstellung der Diskretisierung und der Erstellung des Gleichungssystems wird ein einfaches Beispiel durchgeführt. Die Einheiten zu den Werten in den Matrizen und Vektoren sind in der Sammlung der Formelzeichen auf Seite  $i$  angegeben.

Das Finite-Elemente-Netz besteht nur aus einem zweidimensionalen rechteckigen linearen Element, welches in den Ausmaßen dem Referenzelement, wie in Abbildung 2.10 dargestellt, entspricht. Dieses Element ist am linken unteren Knotenpunkt ( $x = -1$ ,  $y = -1$ ) in  $x$ - und  $y$ -Richtung gesperrt. Dieser Knotenpunkt ist auch der Knotenpunkt mit Index 1. Am linken oberen Knotenpunkt (Knotenpunkt mit Index 3,  $x = -1$ ,  $y = 1$ ) wird in  $x$ -Richtung gesperrt. Am rechten Rand wird an beiden Knotenpunkten jeweils eine Kraft von 100N aufgebracht. In Abbildung 2.12 ist dieses Element mit Randbedingungen und Knotenpunktnumerierung dargestellt.

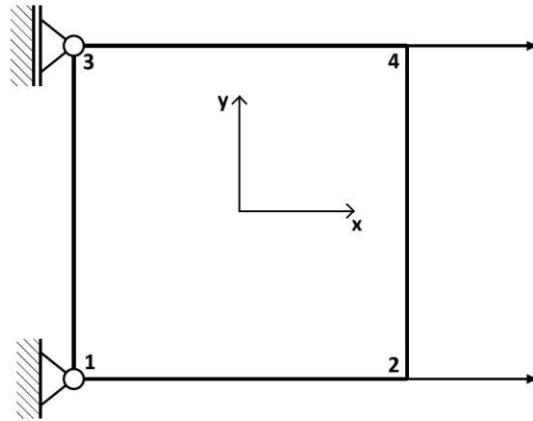


Abbildung 2.12: Beispielement mit Randbedingungen

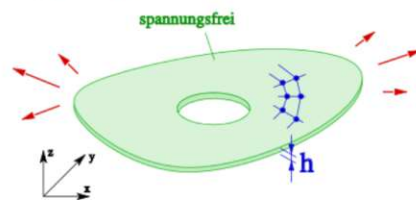
Die Materialkennwerte sind mit

$$\begin{aligned} E &= 210000 \text{ MPa} \\ \nu &= 0.3 \end{aligned} \tag{2.24}$$

gegeben. Zur Berechnung der Steifigkeitsmatrix  $\mathbf{K}$  für dieses Beispiel wird die Gleichung 2.20 verwendet. Dafür wird der Elastizitätstensor für das gegebene Beispiel assembliert. Im dreidimensionalen Fall ist der Elastizitätstensor  $\mathbb{C}$  ein Tensor 4. Stufen mit je 3 Einträgen. Das ergibt 81 Einträge. Dieser Tensor wird im dreidimensionalen Fall durch Einbeziehen der Symmetrien des Tensors auf eine  $6 \times 6$  Matrix reduziert. Im zweidimensionalen Fall wird diese Matrix nochmals auf eine  $3 \times 3$  Matrix vereinfacht. Dafür werden Annahmen getroffen, welche auf zwei Anwendungsfälle zurückzuführen sind. Diese Fälle heißen ebener Verzerrungszustand und ebener Spannungszustand [7].

Sie sind Ansätze, wie bei einer Simulation einer Fläche mit der normal dazu stehenden Achse umgegangen wird. Ein Ansatz ist, die Verformung in Normal-Richtung zuzulassen. Dadurch werden die Spannungen in dieser Richtung auf Null gesetzt. Ein Anwendungsfall ist ein dünnes Blech, das in der Blechebene verformt wird. Das Blech erfährt eine Änderung in der Dicke, und der Spannungszustand bleibt zweidimensional, wie auch in der Abbildung 2.13 dargestellt.

### Ebener Spannungszustand



$$\begin{aligned} \sigma_{zz} &= 0 & \varepsilon_{zz} &\neq 0 \\ \sigma_{xz} &= 0 \\ \sigma_{yz} &= 0 \end{aligned}$$

Abbildung 2.13: zweidimensionaler Spannungszustand mit spannungsfreier z-Achse [7]

Dieser Ansatz wird ebener Spannungszustand genannt. Durch Einbeziehen der Vereinfachungen, die dieser Spannungszustand mit sich bringt, folgt

$$\mathbf{C}_{\text{stress}} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \quad (2.25)$$

als Elastizitätstensor [7]. Wird in der Dicke keine Verformung zugelassen, entstehen Spannungen in Normal-Richtung. Ein Beispiel ist die Simulation des Querschnitts eines Balkens, der über die Länge gleichmäßig belastet wird (siehe Abbildung 2.14).

### Ebener Verzerrungszustand

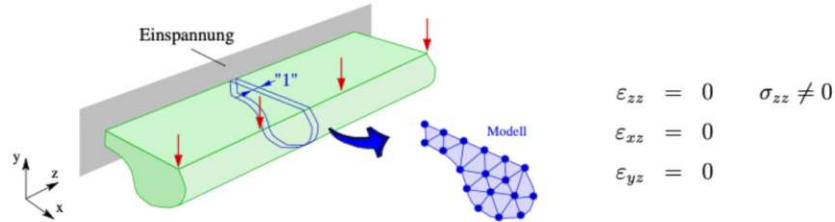


Abbildung 2.14: zweidimensionaler Spannungszustand mit verzerrungsfreier z-Achse [7]

Dieser Ansatz wird ebener Verzerrungszustand genannt. Durch diese Annahmen ändern sich die Einträge zu

$$\mathbf{C}_{\text{strain}} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & 0.5-\nu \end{bmatrix} \quad (2.26)$$

für den Elastizitätstensor [7]. In unserem Fall wird ein ebener Verzerrungszustand angenommen. Wenn die Materialkennwerte in Gleichung 2.26 eingesetzt werden, folgt

$$\mathbf{C} = \begin{bmatrix} 282692.31 & 121153.85 & 0.00 \\ 121153.85 & 282692.31 & 0.00 \\ 0.00 & 0.00 & 80769.23 \end{bmatrix} \quad (2.27)$$

als Elastizitätstensor zweiter Stufe beziehungsweise Elastizitätsmatrix. Diese Matrix ist für alle Integrationspunkte gültig. Die Ansatzfunktionen

$$\begin{aligned} N_1 &= \frac{1}{4}(1-x)(1-y), \\ N_2 &= \frac{1}{4}(1+x)(1-y), \\ N_3 &= \frac{1}{4}(1+x)(1+y), \\ N_4 &= \frac{1}{4}(1-x)(1+y) \end{aligned} \quad (2.28)$$

gelten für dieses Element. Die Dehnungs-Verschiebungs-Matrix  $\mathbf{B}$  ergibt sich aus dem Zusammenhang in Gleichung  $\boldsymbol{\varepsilon} = \mathbf{B} \cdot \mathbf{u}_k$ . Im Verschiebungsvektor

$$\mathbf{u}_k = \left[ u_{k,x1} \quad u_{k,x2} \quad u_{k,x3} \quad u_{k,x4} \quad u_{k,y1} \quad u_{k,y2} \quad u_{k,y3} \quad u_{k,y4} \right]^T \quad (2.29)$$

sind nur die sich ergebenden Verschiebungen der vier Elementknotenpunkte enthalten, nicht aber die

Positionen der Knotenpunkte selbst. Die Länge des Vektors ist die Anzahl der Knotenpunkte multipliziert mit der Anzahl der Dimensionen der simulierten Elemente. Diese Länge entspricht der Anzahl der Freiheitsgrade im System. Der Verzerrungstensor  $\boldsymbol{\varepsilon}$  wird durch Vereinfachung vom Allgemeinen auf den ebenen Verzerrungszustand zu

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \frac{\partial u_x}{\partial x} \\ \frac{\partial u_y}{\partial y} \\ \frac{\partial u_x}{\partial y} + \frac{\partial u_y}{\partial x} \end{bmatrix} \quad (2.30)$$

vereinfacht. Dieser Vektor beschreibt die Verzerrung, die an einem beliebigen Punkt (im Folgenden für die Integration an den Integrationspunkten gewählt) über  $\mathbf{B}$  aus  $\mathbf{u}_k$  berechnet wird. Dass die Verzerrung  $\boldsymbol{\varepsilon}$  als Vektor dargestellt wird, wird Voigt-Notation genannt. Üblicherweise wird die Verzerrung im zweidimensionalen Fall in einer  $2 \times 2$  Matrix dargestellt. Die Voigt-Notation nutzt die Symmetrie dieser Matrix aus und fasst sie dadurch zu einem Vektor mit drei Einträgen zusammen. Auch die Formen der Elastizitätsmatrizen  $\mathbf{C}$  aus Gleichung 2.25 und 2.26 unterliegen der Voigt-Notation. Diese Notation bietet Vorteile wie eine einfachere Implementierung und kompaktere Darstellung [4].  $\mathbf{B}$  ergibt sich durch die Ableitungen der Ansatzfunktionen in der Form, wie sie im Vektor in Gleichung 2.30 gefordert werden.  $\mathbf{B}$  nimmt für dieses Beispiel die Form

$$\mathbf{B} = \begin{bmatrix} \frac{\partial N_1}{\partial r} & 0 & \frac{\partial N_2}{\partial r} & 0 & \frac{\partial N_3}{\partial r} & 0 & \frac{\partial N_4}{\partial r} & 0 \\ 0 & \frac{\partial N_1}{\partial s} & 0 & \frac{\partial N_2}{\partial s} & 0 & \frac{\partial N_3}{\partial s} & 0 & \frac{\partial N_4}{\partial s} \\ \frac{\partial N_1}{\partial s} & \frac{\partial N_1}{\partial r} & \frac{\partial N_2}{\partial s} & \frac{\partial N_2}{\partial r} & \frac{\partial N_3}{\partial s} & \frac{\partial N_3}{\partial r} & \frac{\partial N_4}{\partial s} & \frac{\partial N_4}{\partial r} \end{bmatrix} \quad (2.31)$$

an. Auch der Aufbau dieser Matrix folgt der Voigt-Notation. Die benötigten Ableitungen der Ansatzfunktionen

$$\begin{aligned} \frac{\partial N_1}{\partial r} &= -\frac{1}{4}(1-s), & \frac{\partial N_1}{\partial s} &= -\frac{1}{4}(1-r) \\ \frac{\partial N_2}{\partial r} &= \frac{1}{4}(1-s), & \frac{\partial N_2}{\partial s} &= -\frac{1}{4}(1+r) \\ \frac{\partial N_3}{\partial r} &= \frac{1}{4}(1+s), & \frac{\partial N_3}{\partial s} &= \frac{1}{4}(1+r) \\ \frac{\partial N_4}{\partial r} &= -\frac{1}{4}(1+s), & \frac{\partial N_4}{\partial s} &= \frac{1}{4}(1-r) \end{aligned} \quad (2.32)$$

werden in die Dehnungs-Verschiebungs-Matrix  $\mathbf{B}$  eingesetzt. Diese Matrix wird für jeden Integrationspunkt mit den Koordinaten dieses Integrationspunktes im Referenzelement berechnet. Für die Zusammenstellung von  $\mathbf{K}$  wird die Summe in Gleichung 2.20 an jedem der vier Integrationspunkte des Elements berechnet. Hierzu fehlt noch die Berechnung der Determinante der Jacobi-Matrix für unser Element an den einzelnen Integrationspunkten. Die Jacobi-Matrix ist als

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial s} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial s} \end{bmatrix} = \sum_{i=1}^4 \begin{bmatrix} \frac{\partial N_i}{\partial r} \cdot x_i & \frac{\partial N_i}{\partial s} \cdot x_i \\ \frac{\partial N_i}{\partial r} \cdot y_i & \frac{\partial N_i}{\partial s} \cdot y_i \end{bmatrix} \quad (2.33)$$

definiert und beschreibt die Abbildung vom Referenzelement zum realen Element, das im Finite-Elemente-Netz liegt. In unserem sehr einfach gehaltenen Fall, in dem das Element dem Referenzelement gleicht, nimmt die Jacobi-Matrix an jeder Stelle des Elements die Werte

$$\mathbf{J} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow \det(\mathbf{J}) = 1 \quad (2.34)$$

an. Es bleibt die Matrix  $\mathbf{B}$  zu berechnen. Diese ist in jedem Integrationspunkt unterschiedlich. Im rechten

oberen Integrationspunkt folgt

$$\mathbf{B} = \begin{bmatrix} -0.394 & 0.394 & -0.106 & 0.106 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.394 & -0.106 & 0.394 & 0.106 \\ -0.394 & -0.106 & 0.394 & 0.106 & -0.394 & 0.394 & -0.106 & 0.106 \end{bmatrix} \quad (2.35)$$

als Dehnungs-Verschiebungs-Matrix  $\mathbf{B}$ . Diese Berechnung wird für alle Integrationspunkte wiederholt. Danach sind alle Werte, die zur Berechnung von  $\mathbf{K}$  nötig sind, gegeben. Mit der Gleichung 2.20 wird

$$\mathbf{K} = \begin{bmatrix} 121154 & -80769 & 20192 & -60577 & 50481 & 10096 & -10096 & -50481 \\ -80769 & 121154 & -60577 & 20192 & -10096 & -50481 & 50481 & 10096 \\ 20192 & -60577 & 121154 & -80769 & 10096 & 50481 & -50481 & -10096 \\ -60577 & 20192 & -80769 & 121154 & -50481 & -10096 & 10096 & 50481 \\ 50481 & -10096 & 10096 & -50481 & 121154 & 20192 & -80769 & -60577 \\ 10096 & -50481 & 50481 & -10096 & 20192 & 121154 & -60577 & -80769 \\ -10096 & 50481 & -50481 & 10096 & -80769 & -60577 & 121154 & 20192 \\ -50481 & 10096 & -10096 & 50481 & -60577 & -80769 & 20192 & 121154 \end{bmatrix} \quad (2.36)$$

berechnet. Der Vektor  $\mathbf{f}$  ist leicht zu bestimmen, denn die Kräfteinträge sind direkt an den Knotenpunkten. Durch das Einsetzen der Kraftwerte an den entsprechenden Stellen ergibt sich

$$\mathbf{f} = \begin{bmatrix} 0 & 100 & 0 & 100 & 0 & 0 & 0 & 0 \end{bmatrix}^T \quad (2.37)$$

als Kraftvektor. Damit wurden die Neumann-Randbedingungen berücksichtigt. Anschließend müssen die Dirichlet-Randbedingungen integriert werden. Dafür muss die Steifigkeitsmatrix  $\mathbf{K}$  direkt geändert werden.

In diesem Beispiel werden drei Freiheitsgrade gesperrt. Die Sperrung von Freiheitsgraden erfolgt, indem an den entsprechenden Stellen in  $\mathbf{K}$  die gesamte Reihe und Spalte mit Ausnahme der Diagonaleinträge samt dem Eintrag in  $\mathbf{f}$  auf Null gesetzt werden. Die Diagonaleinträge bleiben, um eine Division durch Null zu verhindern. Diese Vorgehensweise entspricht der Eliminationsmethode zur Umsetzung von Dirichlet-Randbedingungen [22]. Das gesamte Gleichungssystem ergibt sich daraus zu:

$$\underbrace{\begin{bmatrix} 121154 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 121154 & 0 & 20192 & 0 & -50481 & 50481 & 10096 \\ 0 & 0 & 121154 & 0 & 0 & 0 & 0 & 0 \\ 0 & 20192 & 0 & 121154 & 0 & -10096 & 10096 & 50481 \\ 0 & 0 & 0 & 0 & 121154 & 0 & 0 & 0 \\ 0 & -50481 & 0 & -10096 & 0 & 121154 & -60577 & -80769 \\ 0 & 50481 & 0 & 10096 & 0 & -60577 & 121154 & 20192 \\ 0 & 10096 & 0 & 50481 & 0 & -80769 & 20192 & 121154 \end{bmatrix}}_{\mathbf{K}} \cdot \underbrace{\begin{bmatrix} u_{k,x1} \\ u_{k,x2} \\ u_{k,x3} \\ u_{k,x4} \\ u_{k,y1} \\ u_{k,y2} \\ u_{k,y3} \\ u_{k,y4} \end{bmatrix}}_{\mathbf{u}_k} = \underbrace{\begin{bmatrix} 0 \\ 100 \\ 0 \\ 100 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{f}} \quad (2.38)$$

Der hier beschriebene Vorgang zur Erstellung dieses Gleichungssystems wird durch die Python-Bibliothek MFEM realisiert. In MFEM müssen nach dem Einlesen des Finite-Elemente-Netzes nur die Materialkennwerte und die Randbedingungen definiert werden. Mithilfe von MFEM werden die Steifigkeitsmatrix und der Kraftvektor erstellt. Das Gleichungssystem wird mit einem Lösungsalgorithmus gelöst. Der Lösungsvektor dieses Gleichungssystems entspricht den Verschiebungen der Knotenpunkte.

Im Anhang 4.3 befindet sich der Text der MFEM kompatiblen Netzdatei, die ein Finite-Elemente-Netz beschreibt, welches dem Aufbau in Abbildung 2.12 entspricht. Im Anhang 4.3 befindet sich das Python-Programm, in dem die Netzdatei eingelesen und mit Randbedingungen sowie Materialkennwerten versehen wird. Nach der Erstellung der Steifigkeitsmatrix und des Kraftvektors werden diese im Programm dargestellt. Da die Materialkennwerte und Randbedingungen gleich gewählt wurden wie im Beispiel in diesem Kapitel, sind die Ergebnisse identisch.

In MFEM wird im Unterschied zu unserem Beispiel die Elastizitätsmatrix nicht mit E-Modul und Poisson-Zahl, sondern mit den Lamé-Konstanten berechnet. Um die gleichen Werte der Steifigkeitsmatrix in unserem Beispiel und in der Simulation zu erreichen, werden die Lamé-Konstanten aus den hier verwendeten Materialkennwerten berechnet. Die Gleichungen

$$\lambda = \frac{E \nu}{(1 + \nu)(1 - 2\nu)} \quad (2.39)$$

$$\mu = \frac{E}{2(1 + \nu)} \quad (2.40)$$

ermöglichen diese Umrechnung. Dabei ist  $E$  der E-Modul,  $\nu$  die Poissonzahl und  $\lambda$  sowie  $\mu$  die Lamé-Konstanten [1].

### 2.2.1.5 Periodische Randbedingung

Die Einheitszellen werden aneinander anliegend angeordnet und bilden so eine Gitterstruktur. Diese Gitterstruktur wird durch äußere Krafteinwirkung deformiert. Für die Einheitszelle bedeutet das, dass die freie Deformation durch die angrenzenden Einheitszellen eingeschränkt wird.

Um bei der Deformation einer einzelnen Einheitszelle diese eingeschränkte Deformation zu modellieren, wird eine periodische Randbedingung genutzt. Diese periodische Randbedingung stellt sicher, dass keine Deformationen der Einheitszelle stattfinden, die in einer Gitterstruktur zu Klaffen oder Überlappungen führen würden [19].

Eine vollständige periodische Randbedingung, wie in Abbildung 2.15, gibt der Einheitszelle die Möglichkeit, die volle Deformation einer Einheitszelle in einer Gitterstruktur zu vollziehen.

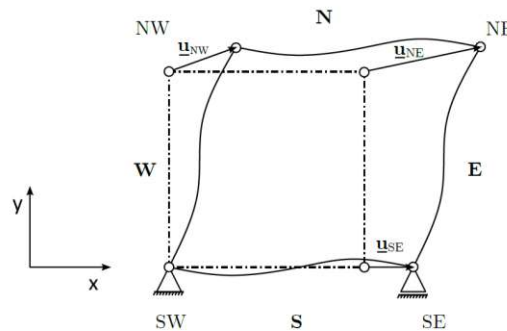


Abbildung 2.15: vollständig periodische Randbedingung [19]

Die Simulation in dieser Arbeit erfolgt mit MFEM. In MFEM gibt es keine integrierte Möglichkeit, periodische Randbedingungen wie in Abbildung 2.15 zu nutzen. Darum wird eine Möglichkeit genutzt, das Gleichungssystem  $\mathbf{K}\mathbf{u}_k = \mathbf{f}$  so zu erweitern, dass die periodische Randbedingung eingehalten wird,

jedoch die Steifigkeitsmatrix in ihren Einträgen nicht verändert wird. Diese Einträge dürfen nicht geändert werden, da sich sonst die Steifigkeitszusammenhänge zwischen den Knotenpunkten verändern. Dadurch würde sich auch die Steifigkeit der gesamten Einheitszelle ändern, wodurch diese nicht korrekt simuliert wird.

Eine Realisierung einer Randbedingung, wie in Abbildung 2.15, ist jedoch nur mit sehr hohem Aufwand in MFEM umsetzbar. Um den Umfang dieser Diplomarbeit nicht zu überschreiten, wird für die Simulation eine vereinfachte Variante der periodischen Randbedingung implementiert. In dieser vereinfachten Variante, wie in Abbildung 2.16 dargestellt, wird die Deformation der Einheitszelle zusätzlich zur vollständig periodischen Randbedingung eingeschränkt.

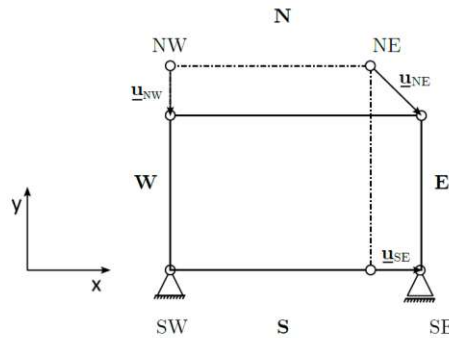


Abbildung 2.16: vereinfachte periodische Randbedingung [19]

Die Deformationen bleiben an den vier Grenzen auf einer horizontalen bzw. vertikalen Linie ausgerichtet. Weiter wird am westlichen Rand die Verschiebung in x-Richtung und am südlichen Rand in y-Richtung gesperrt. Diese Randbedingungen können durch einfache Dirichlet-Randbedingungen realisiert werden. Am östlichen Rand wird eine Deformation in x-Richtung erzwungen. Die Implementierung dieser Randbedingung ist komplex, wird jedoch von MFEM durch einen Befehl korrekt berücksichtigt.

Am nördlichen Rand muss jeder Knoten, bis auf den Eckknoten am westlichen Rand, in x- und y-Richtung verschiebbar bleiben. Dabei müssen alle Knoten am nördlichen Rand die gleiche Verschiebung in y-Richtung erfahren. Zur Realisierung dieser Randbedingung werden Zwangsbedingungen eingeführt [19]. Die Zwangsbedingungen werden mithilfe der Methode der Lagrange-Multiplikatoren manuell implementiert, da MFEM keine integrierte Unterstützung für die Definition solcher Bedingungen bereitstellt. Bei der Lagrange-Multiplikatoren-Methode wird das Gleichungssystem erweitert, und die Abhängigkeiten der betroffenen Randknotenpunkte zueinander werden definiert. Die folgenden Herleitungen und Definitionen basieren im Wesentlichen auf [26].

Das Gleichungssystem  $\mathbf{K}\mathbf{u}_k = \mathbf{f}$  hat  $n$  Gleichungen, wobei  $n$  die Anzahl der Freiheitsgrade im System ist. Werden  $m$  Abhängigkeiten von Knotenpunkten zueinander gefordert, werden diese in der Gleichung 2.41 definiert. Die periodische Randbedingung wird durch Zwangsbedingungen am nördlichen Rand realisiert. Diese Randbedingungen werden im Gleichungssystem

$$\mathbf{C} \cdot \mathbf{u}_k - \mathbf{q} = 0 \quad (2.41)$$

definiert [26]. Um den Aufbau dieses Gleichungssystems zu beschreiben, wird als Beispiel das Finite-Elemente-Netz in Abbildung 2.17 herangezogen.

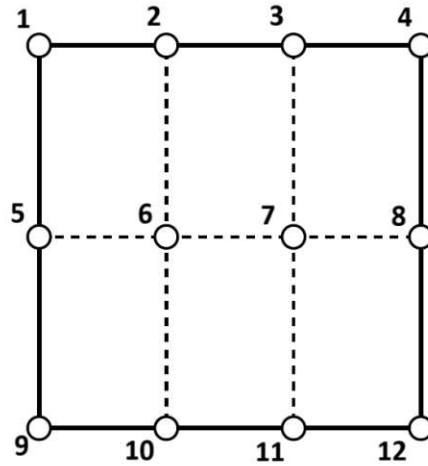


Abbildung 2.17: Finite-Elemente-Netz mit Zwangsbedingungen am nördlichen Rand

Eine quadratische Geometrie wird durch neun Finite-Elemente diskretisiert. Dadurch ergeben sich zwölf Knotenpunkte, die durchnummeriert sind. Am nördlichen Rand soll eine periodische Randbedingung modelliert werden. Dazu müssen durch Zwangsbedingungen die Verschiebungen der Knotenpunkte 2 bis 4 in  $y$ -Richtung mit der Verschiebung des ersten Knotenpunktes in  $y$ -Richtung gekoppelt werden. Daraus ergeben sich

$$\begin{aligned}
 u_{k,y1} &= u_{k,y2} \\
 u_{k,y1} &= u_{k,y3} \\
 u_{k,y1} &= u_{k,y4}
 \end{aligned} \tag{2.42}$$

als die drei notwendigen Zwangsbedingungen für eine periodische Randbedingung. Die Anzahl der Zwangsbedingungen wird mit  $m$  bezeichnet. Diese Zwangsbedingungen werden wie in Gleichung 2.41 dargestellt.  $\mathbf{u}_k$  ist der Vektor der Verschiebungen und hat in diesem Beispiel 24 Einträge. Die Länge von  $\mathbf{u}_k$  der Anzahl der Freiheitsgrade im System und wird im folgenden mit  $n$  beschrieben.

In der Matrix  $\mathbf{C}$  werden die Zwangsbedingungen der Knotenpunkte zueinander definiert.  $\mathbf{C}$  hat eine Größe von  $m \times n$ . Im Beispiel aus Abbildung 2.17 sind das die Zwangsbedingungen 2.42.  $\mathbf{q}$  ist ein Vektor, der die Abstände der Positionen der Knotenpunkte zueinander definiert. Da im gegebenen Beispiel die Knotenpunkte bei Deformierung horizontal ausgerichtet bleiben, muss die Verschiebung in  $y$ -Richtung die gleiche sein. Dadurch ist in diesem Beispiel  $\mathbf{q}$  ein Nullvektor. Das Gleichungssystem 2.50 folgt zu

$$\begin{bmatrix} \dots & 1 & -1 & 0 & 0 & \dots \\ \dots & 1 & 0 & -1 & 0 & \dots \\ \dots & 1 & 0 & 0 & -1 & \dots \end{bmatrix} \cdot \begin{bmatrix} \vdots \\ u_{k,y1} \\ u_{k,y2} \\ u_{k,y3} \\ u_{k,y4} \\ \vdots \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.43}$$

mit dem Wert Null in allen nicht aufgeführten Einträgen in  $\mathbf{C}$  [26]. Im Folgenden wird beschrieben, wie das Gleichungssystem  $\mathbf{K}\mathbf{u}_k = \mathbf{f}$  verändert werden muss, um den Zwangsbedingungen aus Gleichung 2.41 zu unterliegen. Dafür wird die Lagrange-Multiplikatoren-Methode herangezogen. Bei der Lagrange-

Multiplikator-Methode werden  $m$  Lagrange-Multiplikatoren als Vektor

$$\boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_m \end{bmatrix} \quad (2.44)$$

eingeführt [26]. Dieser Vektor der Lagrange-Multiplikatoren wird mit  $\mathbf{C} \cdot \mathbf{u}_k - \mathbf{q} = 0$  multipliziert und folgt zu:

$$\boldsymbol{\lambda}^T \cdot (\mathbf{C} \cdot \mathbf{u}_k - \mathbf{q}) = 0 \quad (2.45)$$

Zur Beschreibung des Gleichgewichtszustands des Systems wird anschließend das potenzielle Energie-Funktional hergeleitet. Durch das Integrieren der Gleichung  $\mathbf{K}\mathbf{u}_k = \mathbf{f}$  nach dem Verschiebungsvektor  $\mathbf{u}_k$  ergibt sich das potenzielle Energie-Funktional

$$\Pi_p = \frac{1}{2} \mathbf{u}_k^T \mathbf{K} \mathbf{u}_k - \mathbf{u}_k^T \mathbf{f} \quad (2.46)$$

welches das potenzielle Energie-Funktional des Systems darstellt [26]. Es wird mit diesem Ausdruck das Potenzial eines linear-elastischen Systems dargestellt. Dabei beschreibt der erste Term die in Folge der Deformation gespeicherte elastische Energie, während der zweite Term die Arbeit der äußeren Kräfte repräsentiert. Zu diesem Funktional wird die linke Seite der Gleichung 2.45 addiert, und es ergibt sich die Gleichung

$$\Pi_p = \frac{1}{2} \mathbf{u}_k^T \mathbf{K} \mathbf{u}_k - \mathbf{u}_k^T \mathbf{f} + \boldsymbol{\lambda}^T \{ \mathbf{C} \mathbf{u}_k - \mathbf{q} \} \quad (2.47)$$

als modifiziertes Funktional [26]. Dieses modifizierte Funktional wird einmal nach  $\mathbf{u}_k$  und einmal nach  $\boldsymbol{\lambda}$  abgeleitet. Diese beiden sich daraus ergebenden Gleichungssysteme werden in eines zusammengeführt und ergeben dadurch das erweiterte Gleichungssystem

$$\begin{bmatrix} \mathbf{K} & \mathbf{C}^T \\ \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}_k \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{q} \end{bmatrix} \quad (2.48)$$

in dem das ursprüngliche Gleichungssystem unverändert enthalten ist [26]. Es wurden  $m$  Gleichungen hinzugefügt, die die periodischen Randbedingungen repräsentieren. Das Gleichungssystem wird gelöst und die Verschiebungen mit den Einflüssen der Zwangsbedingungen berechnet.

Ein Nachteil dieser Methode ist, dass die Anzahl an Unbekannten erhöht wird. Zusätzlich werden nicht positive Diagonalterme in der Steifigkeitsmatrix eingeführt. Aus diesen Gründen wird das Lösen des Gleichungssystems länger dauern. Der Vorteil dieser Methode ist, dass die Zwangsgleichungen exakt erfüllt werden und in MFEM nachträglich einfach hinzugefügt werden können [26].

Um zu zeigen, wie das Gleichungssystem einer Finite-Elemente-Simulation bei der Implementierung einer Zwangsbedingung erweitert wird, wird das Beispiel aus 2.2.1.4 aufgegriffen. Gleichzeitig wird mithilfe dieses Beispiels gezeigt, dass die Lagrange-Multiplikator-Methode die Zwangsbedingung exakt einhält. Dafür erfolgt eine kleine Änderung im Aufbau des Beispiels. Die Kräfte, die in 2.12 jeweils auf die Knoten des östlichen Randes wirken, werden aufaddiert und nur am oberen Knotenpunkt aufgebracht (siehe 2.18).

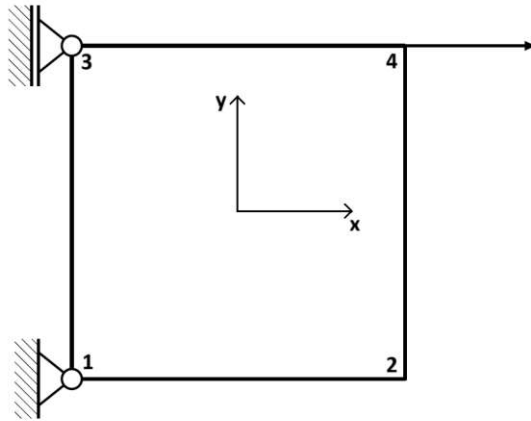


Abbildung 2.18: Beispiелеlement mit Randbedingungen und Zwangsbedingungen

Die Zwangsbedingung soll die Verschiebungen der Knoten zwei und vier in x-Richtung koppeln. Dadurch entstehen drei Szenarien aus den gegebenen Beispielen. Die Kraft ist auf beide Knotenpunkte aufgeteilt (Abbildung 2.12) ohne Zwangsbedingungen. Die Kräfte werden am oberen Knotenpunkt aufsummiert aufgebracht (Abbildung 2.18) und einmal ohne sowie einmal mit Zwangsbedingung simuliert.

Die beiden Fälle ohne Zwangsbedingung unterscheiden sich im Bezug auf die Verformung der Geometrie durch die unterschiedliche Krafteinwirkung. Wird im Fall der Zwangsbedingung diese genau eingehalten, muss die Verformung der Geometrie exakt mit der im Fall der aufgeteilten Kräfte übereinstimmen. Die Zwangsbedingung muss dazu führen, dass die Verschiebungen in x-Richtung gleich groß sind, was einer Aufteilung der Kräfte wie im Ursprungsbeispiel entspricht. Wenn das der Fall ist, wurde gezeigt, dass die Zwangsbedingungen exakt eingehalten werden. Der Aufbau des Gleichungssystems mit nur einer Kraft und einer Zwangsbedingung wird im Folgenden gezeigt. Die Zwangsbedingung

$$u_{k,x2} = u_{k,x4} \quad (2.49)$$

wird in Form des Gleichungssystems 2.41 dargestellt:

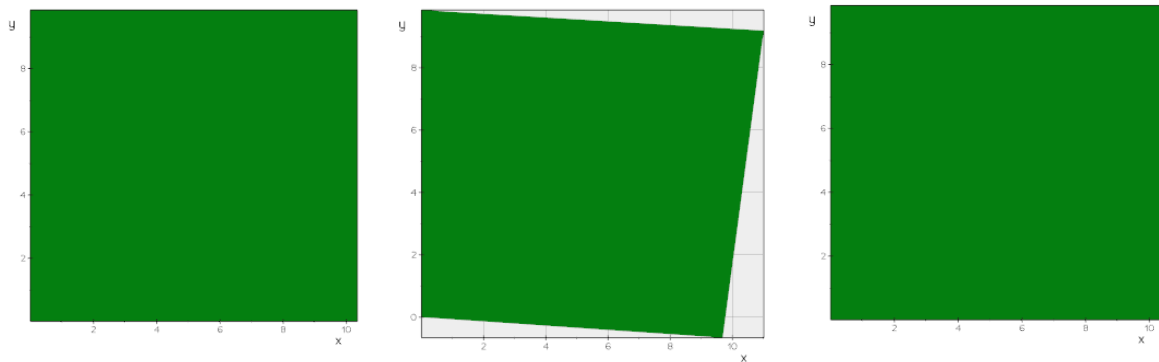
$$\mathbf{C} \cdot \mathbf{u}_k = \mathbf{q} \Rightarrow \begin{bmatrix} 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} u_{k,x1} \\ u_{k,x2} \\ u_{k,x3} \\ u_{k,x4} \\ u_{k,y1} \\ u_{k,y2} \\ u_{k,y3} \\ u_{k,y4} \end{bmatrix} = 0 \quad (2.50)$$

Werden  $\mathbf{C}$  und  $\mathbf{q}$  in die Gleichung 2.48 eingesetzt und berücksichtigt, sodass nur noch die Kraft an Punkt vier mit doppelter Kraft wirkt, folgt

$$\underbrace{\begin{bmatrix} 121154 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 121154 & 0 & 20192 & 0 & -50481 & 50481 & 10096 & 1 \\ 0 & 0 & 121154 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 20192 & 0 & 121154 & 0 & -10096 & 10096 & 50481 & -1 \\ 0 & 0 & 0 & 0 & 121154 & 0 & 0 & 0 & 0 \\ 0 & -50481 & 0 & -10096 & 0 & 121154 & -60577 & -80769 & 0 \\ 0 & 50481 & 0 & 10096 & 0 & -60577 & 121154 & 20192 & 0 \\ 0 & 10096 & 0 & 50481 & 0 & -80769 & 20192 & 121154 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\mathbf{K}} \cdot \underbrace{\begin{bmatrix} u_{k,x1} \\ u_{k,x2} \\ u_{k,x3} \\ u_{k,x4} \\ u_{k,y1} \\ u_{k,y2} \\ u_{k,y3} \\ u_{k,y4} \\ \lambda_1 \end{bmatrix}}_{\mathbf{u}_k} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 200 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{f}} \quad (2.51)$$

als Gleichungssystem für das gegebene Beispiel. Nach dem Lösen des Gleichungssystems 2.51 sind die Lösungen der Verschiebungen das Ergebnis dieser Simulation, während  $\lambda_1$  keine wichtige Bedeutung hat [19, 26].

Für dieses Beispiel werden diese drei Simulationen durchgeführt und anschließend verglichen. Damit wird untersucht, ob die Zwangsbedingungen exakt umgesetzt werden. Die Verformung der Geometrie wird in Abbildung 2.19 dargestellt.



(a) Zwei Kräfte ohne Zwangsbedingung (b) Eine Kraft ohne Zwangsbedingung (c) Eine Kraft mit Zwangsbedingung

Abbildung 2.19: Vergleich der Verformung der drei Fälle der Randbedingungen

Der optische Vergleich von 2.19 (a) und 2.19 (c) bestätigt unsere Annahme, dass die Zwangsbedingung mit der verwendeten Methode umgesetzt wird. Um zu bestimmen, ob die Methode auch exakt ist, werden die Verschiebungen vom rechten oberen Knoten im Fall (a) und (c) verglichen. Die Verschiebungen für den Fall (a) mit zwei Kräften und ohne Zwangsbedingung ergeben gerundet  $u_{k,x4} = 0.08666685$  und  $u_{k,y4} = -0.03714302$ . Im Fall (c) mit einer Kraft und mit Zwangsbedingung sind die Verschiebungen gerundet  $u_{k,x4} = 0.08666685$  und  $u_{k,y4} = -0.03714302$ . Da diese Verschiebungen übereinstimmen, ist die Methode, die Zwangsbedingungen umzusetzen, für dieses Beispiel exakt.

## 2.2.2 Berechnung der Spannungen in simulierter 2D Geometrie

Nach dem Lösen des Gleichungssystems sind die resultierenden Verschiebungen der Knotenpunkte im Verschiebungsvektor  $\mathbf{u}_k$  gespeichert. Um die Spannungen in der gesamten Einheitszelle zu berechnen, wird die Formel des Hooke'schen-Gesetzes  $\boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{u})$  angewandt. Diese Formel ist allgemein gültig und wird für den zweidimensionalen Fall, der in dieser Arbeit vorliegt, in vereinfachter Form genutzt.

Wie in Kapitel 2.2.1.4 bereits erörtert, vereinfacht sich aufgrund der Zweidimensionalität unserer Geometrie und der Annahme des ebenen Verzerrungszustands der Elastizitätstensor zu 2.26. Um die Spannung dieses Spannungszustands zu berechnen, wird der Verzerrungstensor  $\boldsymbol{\varepsilon}$  benötigt. Dieser Verzerrungstensor ist mit

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{xy} \\ \varepsilon_{yx} & \varepsilon_{yy} \end{bmatrix} \quad (2.52)$$

definiert [7]. Zur Berechnung dieses Tensors werden die Verschiebungen benötigt, die bereits durch das Lösen des Finite-Elemente-Problems im Vektor  $\mathbf{u}$  bekannt sind. Um aus diesem Verschiebungsvektor den Verzerrungstensor zu berechnen, muss die Gleichung  $\boldsymbol{\varepsilon} = \frac{1}{2} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)$  (siehe Kapitel 2.2.1.1) herangezogen werden. Diese Gleichung beschreibt den Zusammenhang zwischen Verschiebungsvektor und Verzerrung im linearen Fall. Es wird der Verzerrungstensor  $\boldsymbol{\varepsilon}$  aus dem Verschiebungsgradiententensor  $\nabla \mathbf{u}$  der Verschiebungen berechnet. Dieser Verschiebungsgradiententensor ist definiert als

$$\nabla \mathbf{u} = \begin{bmatrix} \frac{\partial u_x}{\partial x} & \frac{\partial u_x}{\partial y} \\ \frac{\partial u_y}{\partial x} & \frac{\partial u_y}{\partial y} \end{bmatrix}. \quad (2.53)$$

Es ist notwendig, diese Ableitung von  $\mathbf{u}$  für die Finite-Elemente-Simulation zu diskretisieren. Dafür wird die Diskretisierung von  $\mathbf{u}$  aus Gleichung 2.16 verwendet. Die Ableitung dieser diskretisierten Form in Gleichung 2.16 führt zu

$$\nabla \mathbf{u} = \sum_{i=1}^n \nabla \mathbf{N}_i(x, y) \mathbf{u}_{k,i} = \sum_{i=1}^n \begin{bmatrix} \frac{\partial N_x}{\partial x} & \frac{\partial N_x}{\partial y} \\ \frac{\partial N_y}{\partial x} & \frac{\partial N_y}{\partial y} \end{bmatrix} \mathbf{u}_{k,i}. \quad (2.54)$$

Nach Berechnen der Matrix  $\boldsymbol{\varepsilon}$  wird diese nach Hook ( $\boldsymbol{\sigma} = \mathbb{C} : \boldsymbol{\varepsilon}(\mathbf{u})$ ) mit dem Elastizitätstensor multipliziert [7]. Wie in Kapitel  $\boldsymbol{\varepsilon}$  bereits gezeigt, ist dieser Elastizitätstensor in der Voigt-Notation formuliert. Darum muss auch Epsilon unter Ausnutzung der Symmetrie der Matrix in die Form  $\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{xx} & \varepsilon_{yy} & \gamma_{xy} \end{bmatrix}^T$  gebracht und anschließend multipliziert werden. Auch die Spannungen, die dadurch berechnet werden, liegen in der Voigt-Notation vor [4].

Da bei der numerischen Integration in Kapitel 2.2.1.3 bereits die Werte aller Ableitungen der Ansatzfunktionen an den Integrationspunkten berechnet wurden, werden diese Ergebnisse hier wiederverwendet. Dadurch ist die Berechnung im Postprocessing sehr effizient. Das ist nur möglich, wenn die Spannungen direkt am Integrationspunkt berechnet werden. Aus diesem Grund wird bei dieser Simulation und auch bei kommerzieller Software die Spannungsberechnung an den Integrationspunkten durchgeführt.

Die vier Integrationspunkte und deren Gewichte eines solchen rechteckigen Referenzelements sind definiert und in der Abbildung 2.10 dargestellt. Es wird im ebenen Verzerrungszustand der Spannungstensor an allen Integrationspunkten berechnet. Der Spannungstensor wird von der Voigt-Notation zurück in eine Matrixform überführt. Dadurch ergibt sich die Matrix

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{yx} & \sigma_{yy} \end{bmatrix} \quad (2.55)$$

für jeden Integrationspunkt [7]. Für eine Visualisierung der Spannungen im Finite-Elemente-Netz nach

einer Verformung ist es nützlicher, nur einen charakteristischen Wert darzustellen, anstatt der vier Spannungsrichtungen, die im Spannungstensor vorkommen.

Um die Spannungen visualisierbar zu machen, wird die Form der Von-Mises-Vergleichsspannungen genutzt. Diese Spannungsdarstellung vereint alle Werte aus dem Spannungstensor und stellt sie in einem positiven Spannungswert dar. Die Von-Mises-Vergleichsspannungen werden wie in Gleichung 2.56 berechnet.

$$\sigma_{\text{vm}} = \sqrt{\sigma_{xx}^2 - \sigma_{xx} \cdot \sigma_{yy} + \sigma_{yy}^2 + 3\sigma_{xy}^2} \quad (2.56)$$

Diese Von-Mises-Vergleichsspannungen reduziert die Spannungskomponenten auf einen Spannungswert pro Integrationspunkt [4]. Es werden alle Spannungswerte der Integrationspunkte pro Element gemittelt, und dieser Wert wird nun bei der Visualisierung als konstante Spannung über das Element angezeigt.

### 2.2.3 Berechnung der mechanischen Eigenschaften der Einheitszelle

Aus diesen Ergebnissen der Verschiebungen und Spannungen werden die mechanischen Eigenschaften als Kennwerte berechnet. Die Kennwerte, die berechnet werden sollen, sind die Poissonzahl, die maximale Spannung und die Kraft, die benötigt wird, um die aufgezogene Verschiebung zu erreichen.

Die maximal vorkommende Spannung wird aus den in Kapitel 2.2.2 berechneten Spannungen bestimmt. Der höchste vorkommende Wert der dort berechneten Von-Mises-Spannungen wird als maximale Spannung in der Einheitszelle ausgegeben.

Die Poissonzahl wird definiert als  $\nu = -\frac{\Delta y}{\Delta x}$ . Dabei ist  $\Delta x$  die erzwungene Verschiebung des östlichen Randes in x-Richtung und  $\Delta y$  die Verschiebung am nördlichen Rand in y-Richtung. Diese Verschiebungen bleiben durch die periodische Randbedingung über den nördlichen Rand gleich [24].

Die Berechnung der benötigten Kraft, die für die aufgezogene Deformation benötigt wird, ist komplexer. In Kapitel 2.2.1.4 wurde nur gezeigt, wie Dirichlet-Randbedingungen aufgebracht werden, welche einen Freiheitsgrad komplett sperren. Wenn nun eine Deformation aufgezogen wird, wird das Gleichungssystem auch geändert, jedoch fällt dieser Freiheitsgrad im Vergleich zum vollkommenen Sperren des Freiheitsgrades nicht weg. Das bedeutet, dass im Gleichungssystem der Eintrag für diese Verschiebung nicht wegfällt, sondern bereits bekannt ist.

Die Einträge des Verschiebungsvektors, bei denen keine Verschiebung erzwungen wird, sind weiterhin unbekannt. Um diese Randbedingung zu implementieren, werden in der Steifigkeitsmatrix  $\mathbf{K}$  Änderungen vorgenommen. Die Eingriffe in die Steifigkeitsmatrix sind wesentlich komplexer als die Implementierung von Dirichlet-Randbedingungen, bei denen der Freiheitsgrad komplett gesperrt wird. Im Fall der vorgegebenen Verschiebung können die Einträge in  $\mathbf{K}$  nicht einfach gleich Null gesetzt werden. Der Einfluss der vorgegebenen Verschiebung muss auf das Gleichungssystem übertragen werden. Es wird nicht näher auf diese Vorgehensweise eingegangen.

Durch diese Änderungen der Einträge im Gleichungssystem ist es nicht mehr korrekt, die Kräfte, die auf die Randknotenpunkte wirken, dort mit der aufgezogenen Deformation mittels der Gleichung  $\mathbf{K} \cdot \mathbf{u}_k = \mathbf{f}$  zu berechnen. Um diese Kräfte berechnen zu können, werden die resultierenden Kräfte aus den Spannungen der Elemente, die am östlichen Rand angrenzen, berechnet. Dazu wird die Gleichung 2.57 herangezogen [4].

$$F = \sum_{i=1}^{n_R} \boldsymbol{\sigma}_{mi} \cdot \mathbf{n} \cdot l_i \quad (2.57)$$

Hierbei ist  $\boldsymbol{\sigma}_{mi}$  der gemittelte Spannungstensor über alle Integrationspunkte im Randelement  $i$ .  $\mathbf{n}$  ist der Normalvektor auf dem Rand und ergibt sich in diesem Fall als  $\mathbf{n} = \begin{bmatrix} 1 & 0 \end{bmatrix}^T$ .  $l_i$  ist die Länge des Elements  $i$  an der Seite des Elements, die am Rand anschließt. Diese Rechnung wird für alle Randelemente wiederholt, und die Summe aller Einzelkräfte ist die Kraft, die zur Deformierung der Einheitszelle benötigt wird [4].

# Kapitel 3

## Ergebnisse

In diesem Kapitel werden die Ergebnisse dieser Arbeit präsentiert. Diese Ergebnisse sind im ersten Teil die entworfenen Einheitszellen. Dabei wird jede Einheitszelle vorgestellt und das Potenzial ihrer Anpassbarkeit durch die Parameter wird genauer betrachtet.

Im zweiten Teil wird das mechanische Verhalten der Einheitszelle untersucht. Dabei wird zuerst eine Netzfeinheitsstudie durchgeführt. Mit der optimalen Netzfeinheit wird anschließend die Parameterstudie durchgeführt. Dabei werden die Parameter systematisch verändert und die mechanischen Eigenschaften dokumentiert.

### 3.1 Entwurfsprozess der Einheitszellen

In Kapitel 2.1 wurde beschrieben, wie die Einheitszellen aus einzelnen Bézier-Splines zusammengefügt werden und so eine Gitterstruktur bilden. Aufbauend darauf werden in diesem Kapitel verschiedene Einheitszellen vorgestellt, die folgende Anforderungen erfüllen sollen:

- Maximierung der Bandbreite an mechanischen Eigenschaften, die durch Parameteränderungen erzielt werden kann
- Konnektivität zwischen aneinandergereihten Einheitszellen in der Gitterstruktur
- Optimierung der Geometrie für die Durchführung einer Finite-Elemente-Simulation

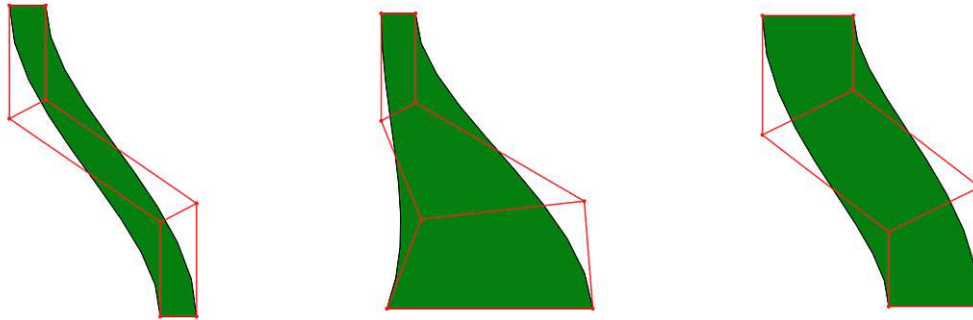
#### 3.1.1 Entwurf parametrisierbarer Einheitszellen mit variablen mechanischen Eigenschaften

Die zugrundeliegende Idee zur Erstellung der Einheitszellen war die Möglichkeit, die mechanischen Eigenschaften dieser Einheitszelle möglichst variabel zu halten. Der Grund dafür ist ihr Einsatz in Gitterstrukturen, die je nach Optimierung lokal unterschiedliche Eigenschaften benötigen.

Die Bandbreite an mechanischen Eigenschaften wird erreicht, indem Parameter definiert werden, welche die Spline-Kontrollpunkte verschieben. Dadurch wird die Form der Einheitszelle, die aus diesen Splines aufgebaut ist, verändert und somit auch die mechanischen Eigenschaften.

Die Kontrollpunkte der Splines werden verschoben, um die Geometrie anzupassen. Wie Parameter in die Gleichungen der Kontrollpunktpositionen eingebunden werden, wurde bereits in Kapitel 2.1.2 erläutert.

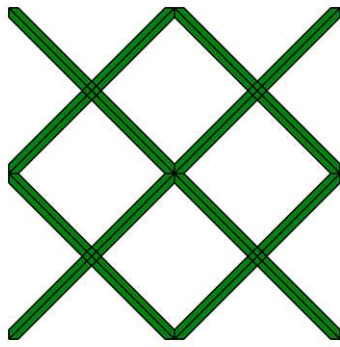
In Abbildung 3.1 wird ein Spline dargestellt, dessen Dicke durch einen Parameter geregelt werden soll. Werden ausgehend vom Spline 3.1(a) jedoch nur einzelne Kontrollpunkte im Spline verschoben, ohne diese über einen Parameter zu organisieren, geht die ursprüngliche Form der Geometrie verloren (siehe Abbildung 3.1 (b)). Durch die Einführung eines Parameters, der die Dicke dieses Splines regelt, wird die Dicke des Splines verändert, ohne dessen Grundform zu ändern (siehe Abbildung 3.1 (c)).



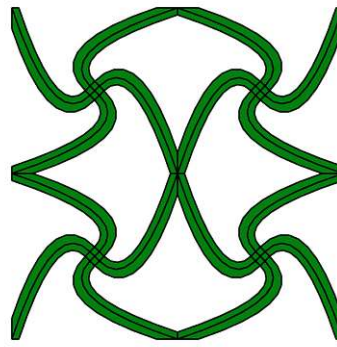
(a) Ursprüngliche Splinegeometrie (b) Verschieben einzelner Kontrollpunkte (c) Kontrolliertes Verschieben der Kontrollpunkte durch einen Parameter

Abbildung 3.1: Änderung der Geometrie eines Splines mithilfe eines Parameters

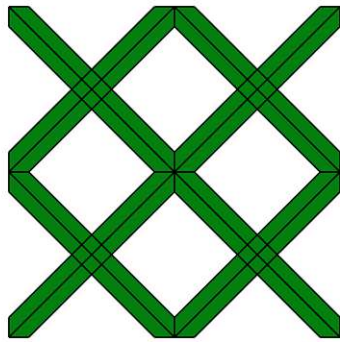
Im Folgenden wird die Einheitszelle in Abbildung 3.2 vorgestellt und anhand dieser die Variation der mechanischen Eigenschaften erläutert.



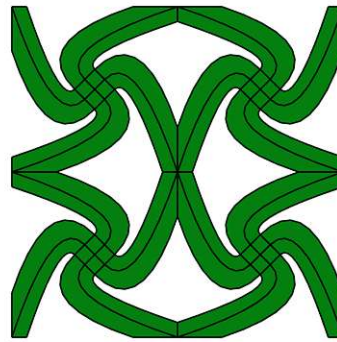
(a) geringes Gewicht mit hoher Steifigkeit



(b) geringes Gewicht mit geringer Steifigkeit



(c) hohe Festigkeit mit hoher Steifigkeit



(d) hohe Festigkeit mit geringer Steifigkeit

Abbildung 3.2: Einfluss von zwei Parametern auf die mechanischen Eigenschaften einer Einheitszelle

In der Abbildung 3.2 ist eine im Zuge dieser Arbeit entworfene Einheitszelle zu sehen. In dieser Einheitszelle kann mit zwei Parametern der aus Splines aufgebauten Flächenverbund, welcher die Einheitszelle aufbaut, beeinflusst werden. In dieser Einheitszelle wird, wie auch in allen weiteren, ein Element viermal gespiegelt zueinander angeordnet. Die Begründung dazu befindet sich in Kapitel 3.1.2.

Der erste Parameter ist die Dicke der Einheitszelle. Mit diesem Parameter kann die Festigkeit erhöht werden. Zusätzlich erhöht sich das Gewicht, da ein größerer Anteil der Einheitszelle mit Materie gefüllt wird. Dieser Parameter wurde definiert, um eine Gewichtsoptimierung für eine gewünschte Festigkeit zu ermöglichen.

Der zweite Parameter ist der Winkel der Verdrehung im Mittelpunkt der vier Kreuze. Wenn dieser Verdrehwinkel null ist (Abbildung 3.2 a und c), ist die Steifigkeit der Einheitszelle besonders hoch. Wenn dieser Verdrehwinkel erhöht wird, sinkt auch die Steifigkeit der Einheitszelle. Auch die Poissonzahl wird von diesem Parameter stark beeinflusst. Wie genau die mechanischen Eigenschaften durch die Geometrieänderungen beeinflusst werden, wird mithilfe der Finite-Elemente-Simulation untersucht.

Durch die Wahl dieses Parameters ist es möglich, auch auf die Steifigkeit und Poissonzahl der indivi-

duellen Einheitszellen in einer Gitterstruktur Einfluss zu nehmen. Dadurch wird eine Optimierung der Gitterstruktur, im Hinblick auf eine erwünschte Deformierung dieser, bei gegebener Belastung ermöglicht.

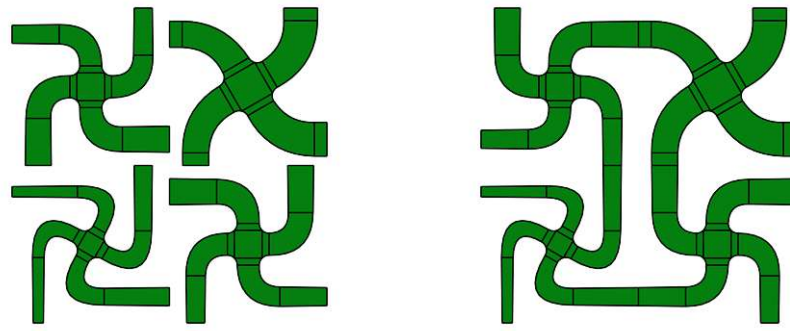
Diese Überlegungen fließen in den Designprozess der Einheitszellen ein, um eine möglichst effektive Wahl der Parameter zu erreichen. Effektiv im Hinblick auf die Bandbreite der mechanischen Eigenschaften, die durch die Variation dieser Parameter erreichbar ist.

### 3.1.2 Sicherstellen der Konnektivität in der Gitterstruktur durch die Definition variabler Parameter innerhalb der Einheitszelle

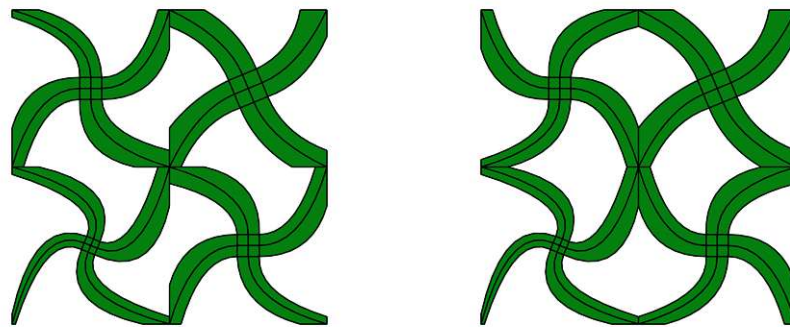
Die Einhaltung der Konnektivität birgt zwei Herausforderungen beim Aufbau einer Gitterstruktur aus sich wiederholenden Einheitszellen. Ein Problem ist die Konnektivität zwischen den Einheitszellen bei Verformung der Gitterstruktur. Dieses Problem wurde bereits in Kapitel 2.2.1.5 mithilfe der periodischen Randbedingung gelöst. Die zweite Herausforderung ist die Konnektivität zwischen den aneinanderliegenden Einheitszellen beim Aufbau der Gitterstruktur. Dort muss sichergestellt werden, dass die Splines der aneinander positionierten Einheitszellen an den Rändern übereinstimmen. Dabei ist die Schwierigkeit, dass die Form des Flächenverbundes an diesen Rändern von den Parametern abhängt.

Um die Konnektivität der Einheitszelle selbst zu garantieren, müssen alle Splines, die an den Rändern der Einheitszelle liegen, entsprechend definiert werden. Das bedeutet, dass im ersten Schritt bei gleichbleibenden Parametern die gegenüberliegenden Ränder spiegelgleich mit Splines besetzt sind. In anderen Worten muss, wenn die Parameter über die gesamte Einheitszelle gleich bleiben, der linke Randbereich dem rechten gleichen. Dasselbe gilt für den oberen und unteren Randbereich, weil die angrenzenden Einheitszellen jeweils mit der im Vergleich gegenüberliegenden Seite an die benachbarte Einheitszelle anstoßen.

Die in Abbildung 3.2 dargestellte Einheitszelle besteht jeweils aus vier einzelnen, sich wiederholenden Kreuzen. Diese Kreuze sind jeweils um die Achse, an der sie aneinander anliegen, gespiegelt. Der Grund dafür liegt in der Konnektivität und der Kontinuität der Splines in der Einheitszelle. In Abbildung 3.3 wird für zwei Einheitszellen untersucht, welche Probleme auftreten, wenn nur ein Kreuz pro Einheitszelle genutzt wird. Diese Einheitszellen sind in Abbildung 3.3 (a) und (c) jeweils viermal in eine quadratische Gitterstruktur eingebettet.



(a) Gitterstruktur aus vier Einheitszellen mit einem Kreuz (b) Einheitszelle aus vier gespiegelten Kreuzen



(c) Gitterstruktur aus vier Einheitszellen mit einem Kreuz (d) Einheitszelle aus vier gespiegelten Kreuzen

Abbildung 3.3: Einheitszellen bestehend aus nur einem Kreuz vier Mal wiederholt

In Abbildung 3.3 (a) ist klar zu sehen, dass bei Verwendung der Einheitszellen mit nur einem Kreuz diese in der Gitterstruktur keine Verbindungen aufweisen. Dadurch ist die Konnektivität nicht gegeben. In Abbildung 3.3 (c) bestehen zwar Übergänge zwischen den Kreuzen, jedoch ist durch die Stufen, die im Berührungspunkt der vier Kreuze entstehen, die Kontinuität nicht zufriedenstellend.

Durch die Nutzung von nur einem Kreuz pro Einheitszelle wäre die Gestaltungsmöglichkeit stark eingeschränkt, um Kontinuität und Konnektivität zu gewährleisten. Aus diesem Grund wird eine Einheitszelle immer aus vier gespiegelten Kreuzen erstellt. In den Abbildungen 3.3 (b) und (d) sind Einheitszellen dargestellt, in denen die Kreuze vier Mal gespiegelt angeordnet sind.

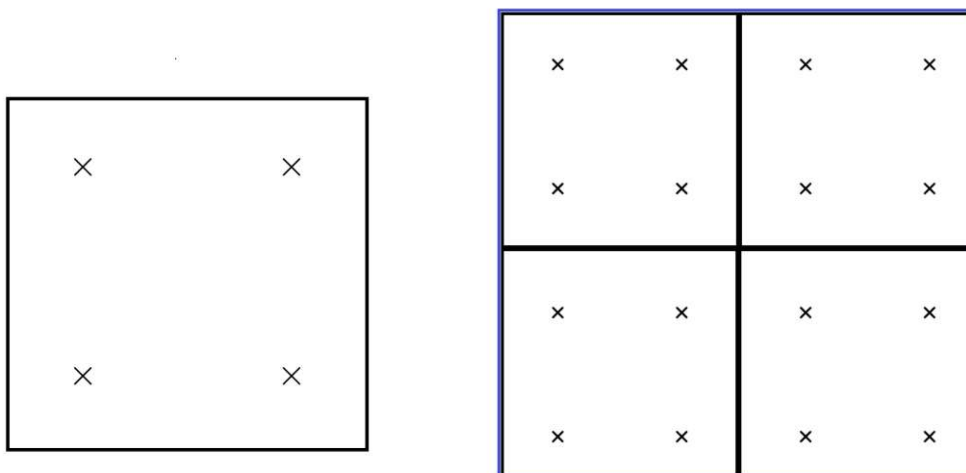
Eine essenzielle Eigenschaft der Gitterstruktur ist die Möglichkeit, die Parameter von Einheitszelle zu Einheitszelle unterschiedlich zu gestalten. Das bedeutet zum Beispiel, dass zwei aneinander angrenzende Einheitszellen in der Gitterstruktur verschiedene Splinedicken aufweisen. Würde in einer Gitterstruktur beispielsweise die Einheitszelle 3.2 (a) an die Einheitszelle 3.2 (c) angrenzen, entsteht eine Stufe in der Geometrie am Übergang zwischen den Einheitszellen. Dadurch wäre die Konnektivität innerhalb der Gitterstruktur nicht mehr gegeben. Um dieses Problem zu lösen, werden die Parameter, wie die Dicke

der Splines anpassen, innerhalb der Einheitszelle variabel gestaltet. So kann sich die Dicke über die Einheitszelle hinweg verändern, ohne die Konnektivität in der Gitterstruktur zu verlieren.

Im Vergleich zu den gezeigten Einheitszellen in Abbildung 3.2 muss also auch die Dicke der Splines variabel gehalten werden. Da die Einheitszellen aus vier einzelnen Kreuzen bestehen, wird zusätzlich der Verdrehwinkel für jedes Kreuz einzeln eingelesen. Es wird im Folgenden geklärt wie die Parameter über die Gitterstruktur hinweg definiert werden. Anschließend wird beschrieben wie und an welcher Stelle die Parameterwerte an der Einheitszelle eingelesen werden.

Die Parameter werden im Zuge der Erstellung der Form der Gitterstruktur über diese Fläche dieser Gitterstruktur definiert. Das bedeutet, dass an jedem Punkt der Gitterstruktur alle Parameter definiert sind. Die Einheitszelle wird mithilfe der Referenzeinheitszelle definiert. Diese Referenzeinheitszelle wird im vorgegebenen Bereich der Gitterstruktur abgebildet. Die Parameter, die nur auf der Gitterstruktur definiert wurden, müssen für die Erstellung der Einheitszelle eingelesen werden. Dafür werden Punkte auf der Referenzeinheitszelle definiert, an denen die Parameter eingelesen werden. Die Ausdehnung der Referenzeinheitszelle (immer ein Quadrat mit Normlänge) wird samt den Punkten, an denen die Parameter eingelesen werden, auf die Fläche der Gitterstruktur abgebildet. An den Stellen, an denen diese Punkte auf der Fläche der Gitterstruktur abgebildet werden, werden alle Parameter, die an diesem Punkt definiert sind, eingelesen und zur Erstellung der Einheitszelle verwendet. Bei diesem gesamten Ablauf wurde noch kein Spline erstellt. Es wurde nur die Ausdehnung der Referenzeinheitszelle auf die Fläche der zu erstellenden Gitterstruktur abgebildet.

In der Abbildung 3.4 (a) ist die Ausdehnung einer Referenzeinheitszelle mit vier Punkten, die durch ein X markiert sind, dargestellt. An diesen Punkten sollen die Parameter für die Erstellung des Flächenverbundes der Einheitszelle eingelesen werden. In Abbildung 3.4 (b) ist in Blau eine quadratische Gitterstruktur eingezeichnet, die aus vier Einheitszellen aufgebaut ist. In dieser Gitterstruktur wurden die Parameter definiert. Es werden die Einheitszellen, die durch die schwarzen Umrandungen dargestellt sind, samt den X-Markierungen auf die Gitterstruktur abgebildet. An den Stellen, an denen die Abgebildeten Markierungen liegen, werden die Parameter von der Gitterstruktur eingelesen und für die Erstellung der Splines der Einheitszelle verwendet.



(a) Ausdehnung einer Referenzeinheitszelle mit Markierungen zum Einlesen von Parametern      (b) Gitterstruktur mit vier abgebildeten Einheitszellen

Abbildung 3.4: Abbildung der Ausdehnung der Einheitszelle auf die Gitterstruktur zum Ablesen der Parameter

Die Wahl der Punkte, an denen die Parameter eingelesen werden, erfolgt so, dass eine maximale Anzahl sinnvoller Parameter innerhalb der Einheitszelle definiert werden kann. Auch bei variierenden Parametern über die Gitterstruktur muss die Konnektivität bestehen bleiben. Das bedeutet, dass an den Teilen der Ränder, an denen Splines anliegen, dieselbe Splinedicke vorliegt, wie in der anschließenden Einheitszelle. Die Splinedicke am Einheitszellenrand wird, wie bereits beschrieben, am vordefinierten Punkt in der Referenzeinheitszelle aus der Fläche der Gitterstruktur abgelesen. Um sicherzustellen, dass derselbe Splinedickenparameter an aneinander anliegenden Splines vorliegt, muss dieser für beide Einheitszellen an derselben Stelle in der Gitterstruktur abgelesen werden. Das bedeutet, dass nicht nur die Splinepositionen bei gleichbleibenden Parametern auf die gegenüberliegende Seite spiegelgleich sein müssen, sondern auch die Position des Punktes, an dem der Splinedickenparameter eingelesen wird. Dadurch wird sichergestellt, dass die Splines an den Einheitszellenrändern mit den Splines der angrenzenden Einheitszelle übereinstimmt.

Wenn, wie in Abbildung 3.3 (b) und (d), die Dicke der Splines über ihre Länge abnimmt, muss jede dieser Dickenänderungen auch durch die Parameter in den Gleichungen der Position der Kontrollpunkte definiert sein. Dadurch sind die Gleichungen zur Positionsbestimmung der Kontrollpunkte von verschiedenen Parametern abhängig. Durch die nicht triviale Form des Flächenverbundes sind diese Positionsgleichungen sehr komplex und schwierig zu definieren. Die Positionsgleichungen der Kontrollpunkte werden manuell erstellt. Aus diesem Grund sind in dieser Arbeit nicht alle möglichen Parameter implementiert worden, da sonst die Positionsgleichung zu komplex wäre.

Wenn all diese Punkte beachtet werden, sind Gitterstrukturen unabhängig von den definierten Parametern immer mit Einheitszellen befüllt, die sowohl eine Konnektivität als auch eine ausreichende Kontinuität aufweisen. In Abbildung 3.5 ist ein Beispiel abgebildet, welches eine quadratische Gitterstruktur zeigt, die von  $3 \times 3$  Einheitszellen aufgebaut wird.

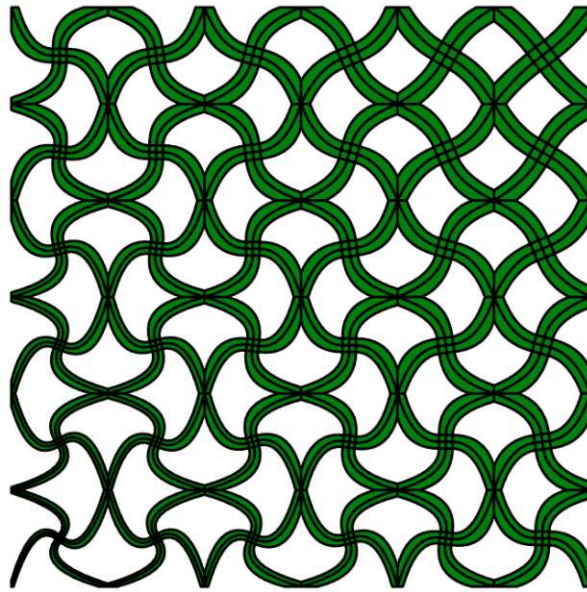


Abbildung 3.5: Quadratische Gitterstruktur mit neun Einheitszellen

### 3.1.3 Optimierung der Geometrie für eine Finite-Elemente-Simulation

Die Finite-Elemente-Simulation wird zur Berechnung der mechanischen Eigenschaften genutzt. Es soll damit auch eine Netzfeinheitsstudie und eine Parameterstudie durchgeführt werden. In der Netzfeinheitsstudie wird die Simulation mit immer größerer Netzfeinheit durchgeführt. Durch das Auswerten von verschiedenen Parametern wird herausgearbeitet, welche Netzfeinheit ein ausreichend genaues Ergebnis bei angemessener Simulationszeit ergibt. In der Parameterstudie werden die Parameter bei gleichbleibender Netzfeinheit variiert und der Einfluss der Parameteränderungen auf die mechanischen Eigenschaften untersucht.

In all diesen Simulationsszenarien ist die Berechnung der Spannungen in der verformten Einheitszelle von Bedeutung. In der Netzfeinheitsstudie und der Parameterstudie wird die maximale Spannung als Ergebnis verwendet. Weiters ist die maximal vorkommende Spannung auch für die Gitterstruktur von Bedeutung. Diese maximale Spannung wird als Indikator verwendet, ob die zulässige Spannung für die in der Fertigung verwendeten Materialien in Kombination mit der Fertigungsweise zulässigen Spannungen überschreitet.

Das Problem ist, dass die Spannungsberechnung in der Finite-Elemente-Simulation kein akkurates Ergebnis liefert, wenn die vernetzte Geometrie, in unserem Fall die Einheitszelle, die die Gitterstruktur aufbaut, Ecken aufweist. Diese Ecken führen dazu, dass die simulierten Spannungen an diesen Ecken sehr hohe Werte aufweisen. Diese Werte sind unrealistisch hoch. Der Grund dafür ist, dass an solchen Ecken in realen Verformungsszenarien eine plastische Verformung einsetzen würde. Dadurch wird die Höhe der

vorkommenden Spannungen beschränkt. Diese plastische Verformung kann jedoch in unserer rein linear elastischen Simulation nicht abgebildet werden.

Ein weiteres auftretendes Phänomen ist, dass die Spannung an diesen Ecken höher wird, je höher die Netzfeinheit gewählt ist [9, 14, 25].

Aus diesen Gründen ist es nicht möglich, die Netzfeinheitsstudie im beschriebenen Umfang durchzuführen. Die Verwendung der maximalen Spannung in der Einheitszelle wird keine Ergebnisse erzielen, die sich bei einer Netzfeinheit stabilisieren. Um dieses Problem zu beseitigen, wurde in einer der entworfenen Einheitszellen an den Stellen, an denen sich solche Ecken befinden, um zusätzliche Splines ergänzt, welche dort eine Rundung erstellen. In Abbildung 3.6 wird eine solche Änderung in der Einheitszelle vorgenommen.

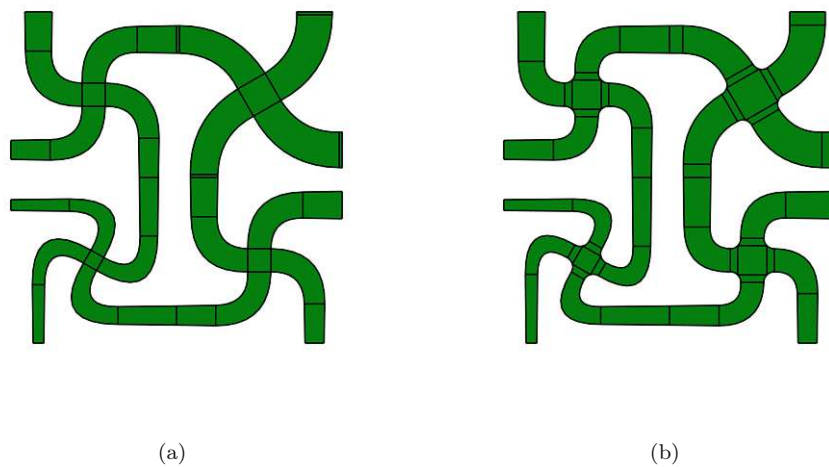


Abbildung 3.6: Erstellen von Rundungen an Ecken in der Einheitszelle

In der Abbildung 3.6 (a) ist eine Einheitszelle mit Ecken dargestellt, an denen keine Spannungen simuliert werden können. In der Abbildung 3.6 (b) ist eine Einheitszelle mit Rundungen zu sehen, an denen eine Spannungssimulation sinnvoll ist.

## 3.2 Einheitszellendesign

In diesem Kapitel werden die im Zuge dieser Arbeit entworfenen Einheitszellen vorgestellt. Dabei werden für jede Einheitszelle die Punkte aus Kapitel 3.1 aufgegriffen und die Eigenschaften der Einheitszellen in Bezug auf diese Punkte beschrieben.

Alle hier vorgestellten Einheitszellen haben die Gemeinsamkeit, dass sie aus vier Kreuzen bestehen, welche sich nur gespiegelt zueinander wiederholen. Auch die Kreuze selbst sind immer mit vier Armen und einem Mittelteil aufgebaut. Die Positionen der Kontrollpunkte, die die Splines aufspannen, werden hierbei mit von den Parametern abhängigen Gleichungen definiert. Diese Gleichungen wiederholen sich in einem Kreuz, jedoch in jedem Arm erneut. Auch die vier gespiegelten Kreuze haben dieselben Gleichungen. Die Splines werden im Anschluss alle gespiegelt. Dadurch sind alle Gleichungen, die die Knotenpunkte der Einheitszellen definieren, gegeben, wenn ein einzelner Arm untersucht wird. Dabei wird immer der linke untere Arm des linken unteren Kreuzes betrachtet. Das wurde für alle folgenden Einheitszellen, die im

Zuge dieser Arbeit gestaltet wurden, gemacht. Die Gleichungen der Positionen der Kontrollpunkte sind im Anhang 4.3 aufgelistet.

Zu diesen gesammelten Gleichungen ist noch zu erwähnen, dass diese Positionen noch mit einer Translation an die richtige Stelle verschoben werden müssen. Die Referenzeinheit zelle wird in einem Quadrat von  $(0,0)$  bis  $(1,1)$  aufgespannt (siehe Kapitel 2.1.1). Die Gleichungen, die im Anhang angeführt sind, spannen einen Arm eines Kreuzes auf (entsprechende Abbildung dazu befindet sich im Anhang). Wenn dieses Kreuz fertiggestellt ist, nimmt es ein Quadrat von  $(-L, -L)$  bis  $(L, L)$  ein. Dadurch bleiben die Gleichungen so simpel wie möglich und können auch für alle vier Kreuze der Einheitszelle genutzt werden. Erst nach dem Aufbau des Flächenverbundes des Kreuzes werden diese, wenn nötig, gespiegelt und mittels Translation in die richtige Position gerückt. Dazu muss  $L = 0.25$  gelten, um eine Referenzeinheit zelle zu erreichen.  $L$  entspricht also einer Länge von einem Viertel der Normlänge der Referenzeinheit zellen. Die Vorgänge des Spiegels und der Translation sind in den Gleichungen im Anhang nicht mitberücksichtigt und müssen danach noch durchgeführt werden.

Wie in Kapitel 2.1.3 beschrieben, wird das Finite-Elemente-Netz direkt aus dem Multipatch, der den Flächenverbund bildet, erstellt. Alle zweidimensionalen Splines der erstellten Einheitszellen werden nach der Erstellung des Flächenverbundes vollautomatisch vernetzt und die Randknotenpunkte werden zu Gruppen zusammengefasst, die die Simulation zur Zuweisung der Randbedingungen nutzt.

### 3.2.1 Chi-Einheitszelle

Diese folgende Einheitszelle wird Chi genannt. Der Name dieser Einheitszelle wurde aufgrund der Ähnlichkeit der Form der Kreuze dieser Einheitszelle mit dem griechischen Buchstaben  $\chi$  gewählt. Diese Einheitszelle Chi besteht aus einem zweidimensionalen Flächenverbund. In der Abbildung 3.7 ist die Grundidee dieser Einheitszelle mit ihren Parametern dargestellt. Die Parameter sind dabei immer in positiver Richtung eingezeichnet. Mit dieser Form einer Einheitszelle soll es ermöglicht werden, mithilfe der Parameter die Steifigkeit dieser zu verändern. Wird der Parameter  $\beta$  auf Null gesetzt, bildet dieses Chi ein X. Dadurch soll eine hohe Steifigkeit erreicht werden. Wenn der Winkel  $\beta$  in positiver oder negativer Richtung erhöht wird, nimmt die Steifigkeit ab. Die Dicke des Chis ist an allen vier Ecken und in der Mitte jeweils als Parameter variabel.

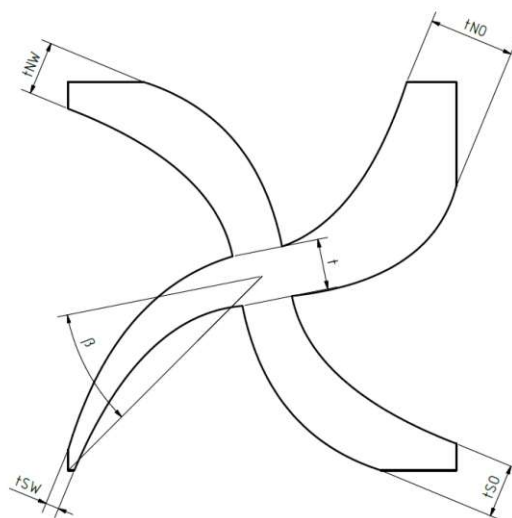
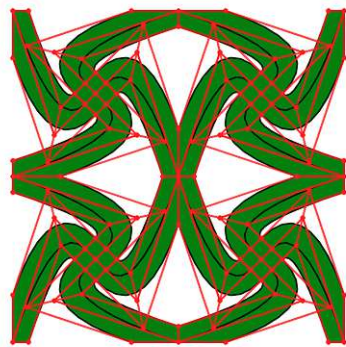
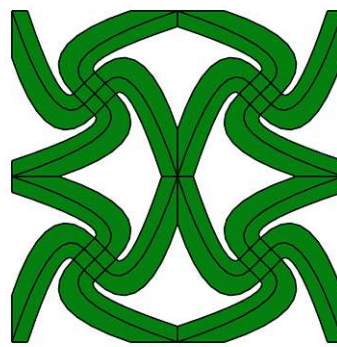


Abbildung 3.7: Positive Richtung der Parameter der Einheitszelle Chi

Die Einheitszelle wird aus vier gespiegelten Chis aufgebaut, um die Konnektivität dieser zu erreichen. Diese zweidimensionale Einheitszelle wird in der Abbildung 3.8 mit konstanten Parametern über die Zelle dargestellt. Die Gleichungen, die die Positionen der Kontrollpunkte in dieser Einheitszelle definieren, sind im Anhang 4.3 aufgelistet. Jedes dieser vier Chis hat vier Arme, die mit zweidimensionalen Splines aufgebaut sind. Dabei werden zwei Splines pro Arm verwendet, um auch die Ecken der Einheitszelle vollständig ausfüllen zu können. Diese beiden Splines sind Bézier-Splines. In Dickenrichtung sind sie linear, und längs entlang des Arms sind sie quadratisch. Das bedeutet, dass die Form der Arme der Form einer quadratischen Funktion folgt. Um das automatisch erstellte Finite-Elemente-Netz zwischen den Splines kompatibel zu halten, müssen alle Eckpunkte der Splines mit den Eckpunkten der angrenzenden Splines übereinstimmen. Darum wird das Quadrat in der Mitte der Chis in vier Quadrate unterteilt.



(a) Einheitszelle mit Kontrollpunkten



(b) Flächenverbund der Einheitszelle

Abbildung 3.8: Einheitszelle Chi mit gleichbleibenden Parametern über die Zelle

Auch in dieser Einheitszelle wurde die Möglichkeit implementiert, die Parameter innerhalb der Einheitszelle zu variieren. Die Parameter hierbei sind die Dicke der Arme und die Verdrehwinkel der Chis. Die Punkte, die in der Einheitszelle zum Ablesen der Parameter definiert wurden, sind auch in Abbildung 3.9 dargestellt.

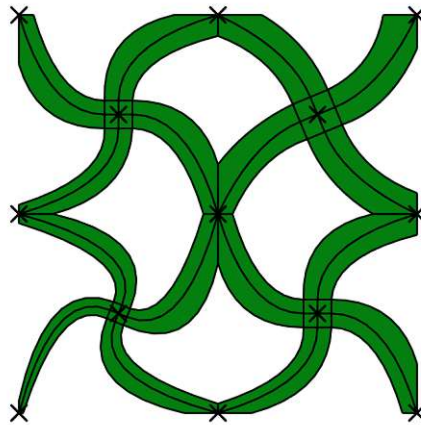


Abbildung 3.9: Markierte Punkte an denen in der Einheitszelle die Parameter abgelesen werden

An diesen insgesamt 13 Punkten wird jeweils die gewünschte Dicke der Splines eingelesen. Die Gleichungen der Kontrollpunkte sind so gestaltet, dass diese eingelesenen Dicken an diesen Stellen der Dicke der Arme der Chis entsprechen. Gibt es eine Dickendifferenz zwischen Endpunkt des Armes und der Chi-Mitte, sind die Kontrollpunkte dazwischen so gewählt, dass die Dicke der Arme über deren Länge kontinuierlich zu- beziehungsweise abnimmt.

Die Verdrehwinkel der Chis werden jeweils an den Mittelpunkten dieser eingelesen. Dadurch wird jedem Chi in der Einheitszelle ein individueller Verdrehwinkel zugewiesen. Auch dieser Winkel beeinflusst die Positionsgleichungen der Kontrollpunkte und muss korrekt in diese integriert werden.

Für diese Einheitszelle muss der Winkel an allen 13 Punkten eingelesen werden. In der Abbildung 3.10 (c) ist im linken oberen Eck gut zu sehen, was der Grund dafür ist. Die Dicke des Splines an diesem Eck ist durch Einlesen des Parameters der Dicke  $t_{NW}$  an dieser Stelle bereits bekannt. Jedoch ist es vom Winkel  $\beta_{NW}$  abhängig, wie diese Dicke aufgeteilt wird. Mit der Aufteilung ist gemeint, mit welcher Länge der Spline am nördlichen- und westlichen Rand angrenzt, um die Splinedicke bei dem Winkel an dieser Stelle korrekt abzubilden. Dafür müssen die Parameter der Dicke und des Winkels direkt am Eck eingelesen werden. Es ist wichtig, die Parameter dort abzulesen, weil die drei potenziell anliegenden Einheitszellen in einer Gitterstrukturanordnung auch dieselben Parameter einlesen müssen, um eine kontinuierliche Einheitszelle zu erreichen. Das ist der Grund, warum auch die Winkel an den neun Punkten, die nicht im Chi-Mittelpunkt liegen, gebraucht werden. In Abbildung 3.10 sind Extremfälle dargestellt, in denen die Parameter über die Einheitszelle hinweg stark variieren.

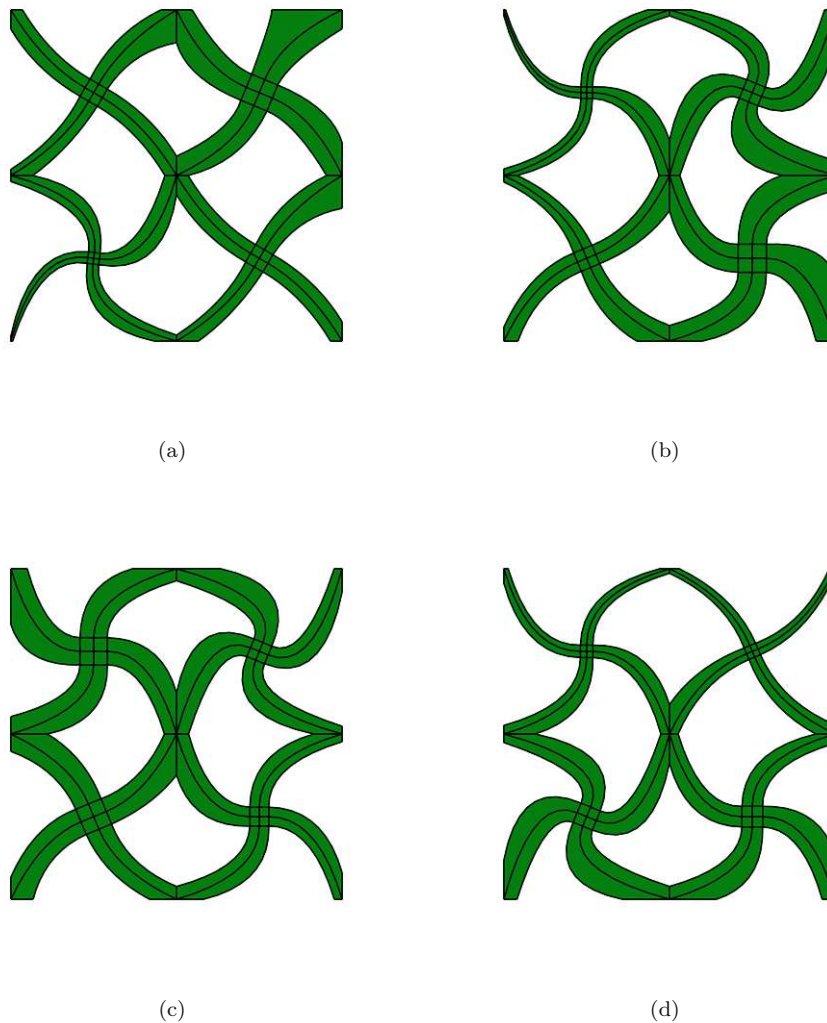


Abbildung 3.10: Einheitszelle Chi mit verschiedenen Parameterverteilungen

Wie in Kapitel 3.1.3 beschrieben ist die Berechnung der Spannungen im Postprocessing nur zielführend, wenn keine Kanten in der Geometrie vorhanden sind. Darum ist bei dieser Einheitszelle zu erwähnen, dass der Flächenverbund immer noch solche Kanten aufweist. Diese Rundungen könnten in einer ähnlichen Weise wie in der Einheitszelle 3.2.2 oder 3.2.3 hinzugefügt werden. Diese Schritte wurden nicht durchgeführt, da in dieser Einheitszelle ein Vielfaches solcher Kanten vorkommt. Die Erstellung all dieser Rundungen ist sehr zeitintensiv und wurde deshalb nicht umgesetzt. Ein weiterer Grund ist die Menge an zusätzlichen Finite-Elementen, die eine solche Implementierung mit sich bringt. Wie in Kapitel 2.1.3 bereits beschrieben, wird jeder Splineparameter jedes Splines in der Einheitszelle mit gleich vielen Elementen vernetzt. In der Form der Einheitszelle, wie sie jetzt vorliegt, enthält die Einheitszelle 48 Bézier-Splines. Wenn nun alle Kanten mit Rundungen versehen werden, erhöht sich die Anzahl der Bézier-Splines auf 122 pro Einheitszelle. Durch Unterteilung aller Splines in gleich viele Elemente, wird sich deren Anzahl bei gleichbleibender Netzfeinheit mehr als verdoppeln. Das führt auch zu einer drastischen Erhöhung der Simulationszeit und der benötigten Rechenkapazitäten.

Die Spannungen sind als Ergebnisse unbrauchbar, wenn die Kanten keine Rundungen haben. Die restlichen mechanischen Eigenschaften sind trotzdem repräsentativ für die generierte Einheitszelle. Aufgrund des großen Mehraufwands in Arbeits- und Rechenzeit wurde also entschieden, diese Einheitszelle trotzdem

einer Netzfeinheits- und Parameterstudie zu unterziehen, ohne die Kanten abzurunden. Die Ergebnisse der Spannungen wurden ausgeklammert, und der Fokus wird auf Ergebnissen wie der Poissonzahl gelegt.

### 3.2.2 Chi-Bar-Einheitszelle

Diese Einheitszelle erhält den Namen Chi-Bar, weil auch diese Kreuze der Einheitszelle an ein griechisches  $\chi$  erinnern. Außerdem sehen die Zwischenräume bei einer aus diesen Elementen zusammengesetzten Gitterstruktur (siehe Abbildung 1.1) wie eine Langhantel (auf Englisch Barbell und hier verkürzt Bar genannt) aus. Diese Einheitszelle Chi besteht aus einem zweidimensionalen Flächenverbund.

Die Grundidee dieser Einheitszelle ist in Abbildung 3.11 dargestellt. Des weiteren sind die Parameter in positiver Richtung eingezeichnet. Durch den Winkel  $\beta$  soll wieder die Steifigkeit variierbar sein. Es ist zu erwarten, dass bei Dehnung dieser Geometrie in der Breite der Winkel  $\beta$  des Kreuzes sich ausdreht und so die Höhe zunimmt. Das ergibt eine negative Poissonzahl, welche mit dem Winkel  $\beta$  beeinflusst werden kann.

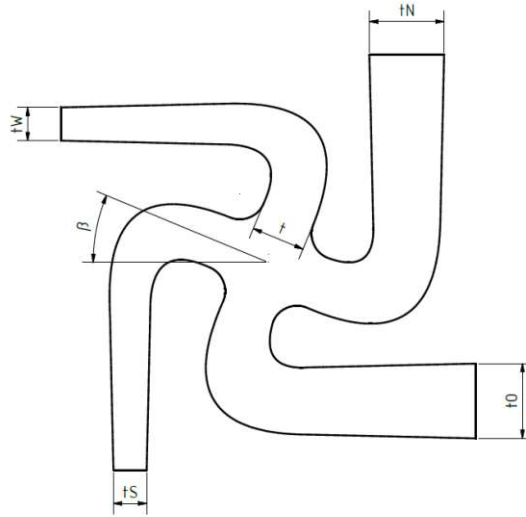
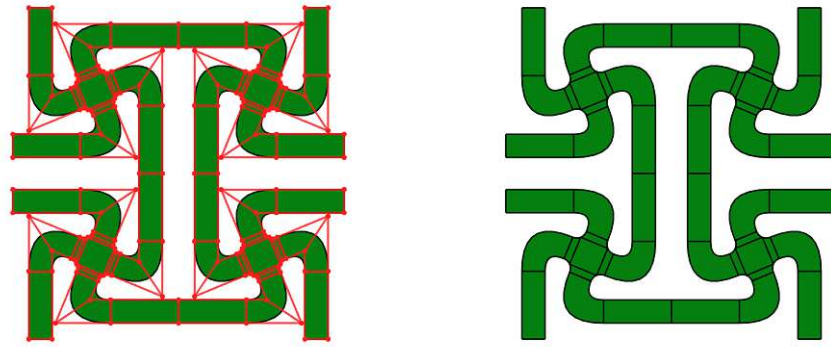


Abbildung 3.11: Positive Richtung der Parameter der Einheitszelle Chi-Bar

Diese Einheitszelle wird aus zweidimensionalen Bézier-Splines zusammengesetzt. In Abbildung 3.12 ist diese Einheitszelle mit über die Einheitszelle konstanten Parametern dargestellt. Die Positionsgleichungen der Kontrollpunkte des Kontrollpolygons dieser Einheitszelle sind im Anhang 4.3 aufgelistet.



(a) Einheitszelle mit Kontrollpunkten

(b) Flächenverbund der Einheitszelle

Abbildung 3.12: Einheitszelle Chi-Bar mit gleichbleibenden Parametern über die Zelle

Auch hier sind wieder vier Chis zueinander gespiegelt angeordnet. Die vier Arme pro Chi enden hierbei aber im Vergleich zur Chi-Einheitszelle am Rand der Zelle und nicht in der Ecke. Dadurch kann die Dicke des Armes in einem Spline dargestellt werden, weil dieser an beiden Enden des Armes geradlinig abschließen kann. Der Arm besteht aus einem geraden und einem gebogenen Bereich. Diese beiden Bereiche werden jeweils mit einem Bézier-Spline dargestellt. Während die Splines des Armes in Dickenrichtung beide linear sind, sind die Grade in Längsrichtung entsprechend einmal linear und einmal quadratisch.

Die Parameter werden über die Einheitszelle hinweg variabel gestaltet. Die Parameter sind die Dicke der Arme, die an jedem Endpunkt dieser und im Mittelpunkt der Chis abgelesen und definiert werden. Auch der Verdrehwinkel der vier Chis wird im Mittelpunkt dieser abgelesen und definiert. Daraus ergeben sich 16 Punkte, an denen die Parameterdaten abgelesen werden. Diese Punkte wurden in der Abbildung 3.13 eingezeichnet.

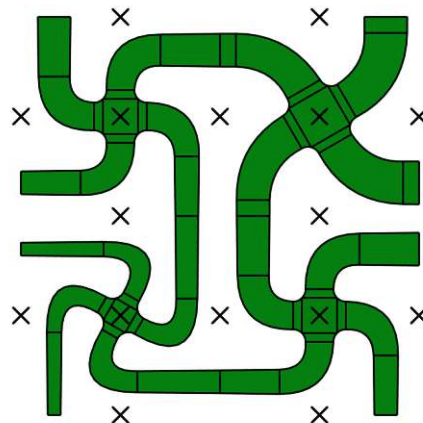


Abbildung 3.13: Markierte Punkte an denen in der Einheitszelle die Parameter abgelesen werden

Die Abbildung 3.14 zeigt Möglichkeiten, wie die Parameter über die Einheitszelle hinweg variiert werden können. Dadurch soll gezeigt werden, welches Potenzial ausgeschöpft werden kann, wenn aus dieser Einheitszelle eine Gitterstruktur aufgebaut wird.

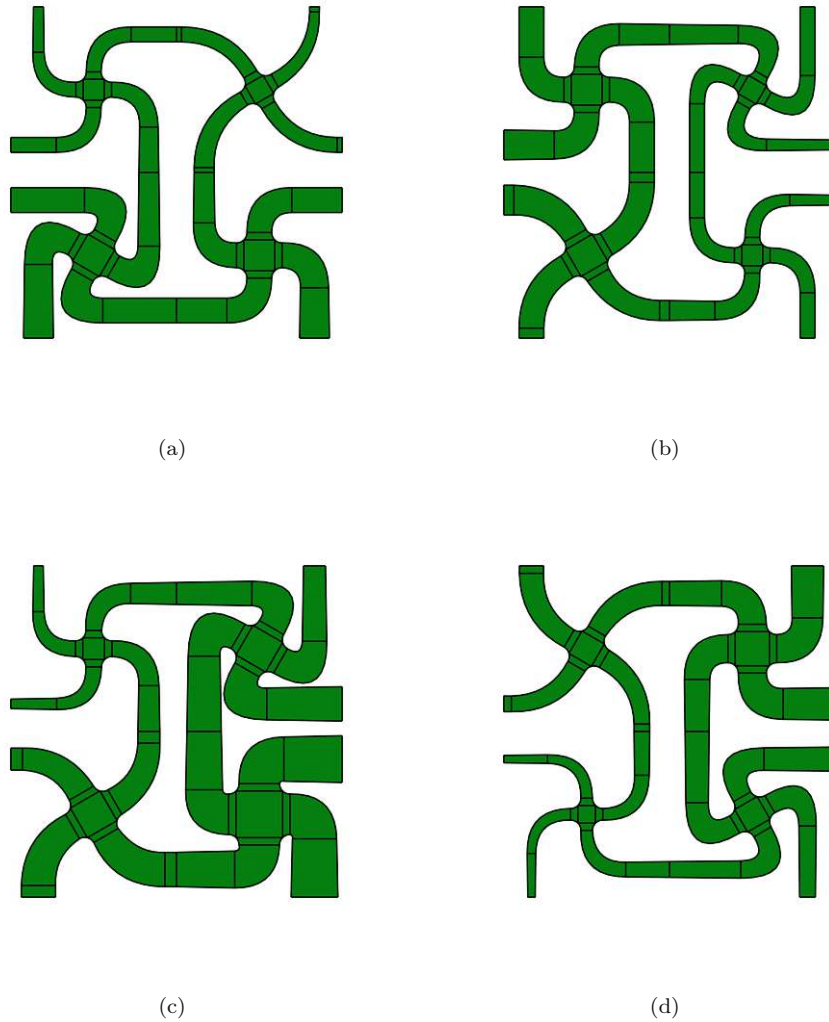


Abbildung 3.14: Einheitszelle Chi-Bar mit verschiedenen Parameterverteilungen

Im Kapitel 3.1.3 wurde bereits erörtert, warum das Abrunden von Kanten in der Einheitszelle essenziell für die Spannungsberechnung ist. Aus diesem Grund wurden in dieser Geometrie zusätzliche Splines eingeführt, die, wie in Abbildung 3.6 dargestellt, die Kanten abrunden. In der Abbildung 3.15 sind die Splines, die die Rundung bilden, zu sehen.

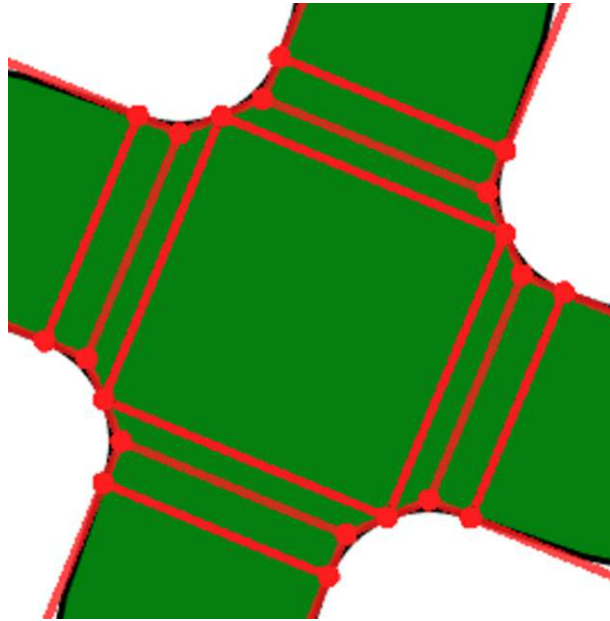


Abbildung 3.15: Rundung in der Einheitszelle Chi-Bar

Diese Abrundung basiert erneut auf der Tatsache, dass Bézier-Splines an ihren Enden die Steigung der Verbindungslinie der Kontrollpunkte annehmen. An der Stelle, an der sich die Splines mit dem benachbarten Spline, der eine Rundung formt, und dem quadratischen Mittelspline treffen, muss also nur sichergestellt werden, dass die Steigung der Verbindungslinien der Kontrollpunkte gleich bleibt. Dadurch ist auch der Übergang zwischen den Teilen der Rundungen kontinuierlich und frei von Knicken.

Der Radius dieser Rundungen ist in der Definition der Kontrollpunktpositionen variabel gehalten und kann deshalb verändert werden. Jedoch ist dieser Rundungsradius kein zusätzlicher Parameter. Diese Anpassung ließe sich mit geringem Aufwand umsetzen, war jedoch im Rahmen dieser Arbeit nicht relevant und wurde daher über die gesamte Gitterstruktur hinweg konstant gehalten.

### 3.2.3 Chi-Bar-Flex-Einheitszelle

Die Chi-Bar-Flex-Einheitszelle ist eine Erweiterung der Chi-Bar-Einheitszelle. Der Unterschied liegt darin, dass es möglich ist, die Arme der Chis am äußeren Ende zu verschieben. Dadurch entsteht eine weitere Biegung in den Übergängen zwischen den Chi-Armen, welche die Einheitszelle noch flexibler macht. Darum wird diese Einheitszelle Chi-Bar-Flex genannt. Auch diese Einheitszelle ist aus einem zweidimensionalen Flächenverbund aufgebaut.

Die Grundidee dieser Einheitszelle gleicht der der Chi-Bar-Einheitszelle und ist in Abbildung 3.16 dargestellt. Die Parameter sind dort in positiver Richtung eingezeichnet. Die zusätzliche Biegung des äußeren Teils der Arme soll die Steifigkeit verringern. Diese Maße, die diese Biegung definieren, sind in diesem Fall keine Parameter.

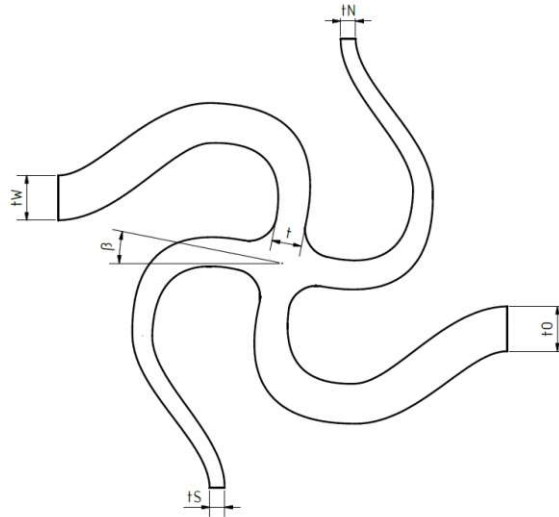
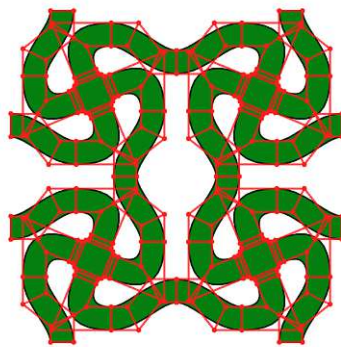
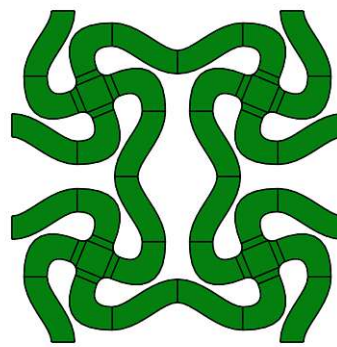


Abbildung 3.16: Positive Richtung der Parameter der Einheitszelle Chi-Bar-Flex

Diese Einheitszelle wird aus zweidimensionalen Bézier-Splines aufgebaut und ist in Abbildung 3.17 dargestellt. Die Positionsgleichungen der Kontrollpunkte des Kontrollpolygons dieser Einheitszelle werden im Anhang 4.3 aufgelistet.



(a) Einheitszelle mit Kontrollpunkten



(b) Flächenverbund der Einheitszelle

Abbildung 3.17: Einheitszelle Chi mit gleichbleibenden Parametern über die Zelle

Die zusätzliche Biegung wurde realisiert, indem der rechteckige Spline außen am Arm in der Chi-Bar-Einheitszelle durch einen Spline dritten Grades ersetzt wurde. Durch diesen Spline, welcher in Abbildung 3.18 nochmals isoliert dargestellt ist, ist eine zusätzliche Krümmung im Übergang zwischen den Chis möglich.

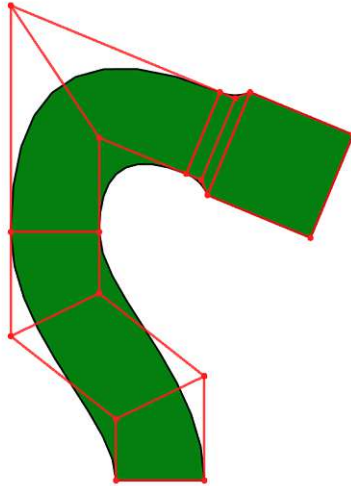


Abbildung 3.18: Detailabbildung eines Armes des Chis das in der Chi-Bar-Flex-Einheitszelle verwendet wird

Gleich wie in der Einheitszelle Chi-Bar sind als Parameter die Dicke und der Verdrehwinkel gewählt. Dabei wird die Dicke der Splines an 16 Punkten eingelesen und der Verdrehwinkel an 4 Punkten im Zentrum der Chis. In Abbildung 3.19 sind die Punkte dargestellt, an denen die Parameter eingelesen werden.

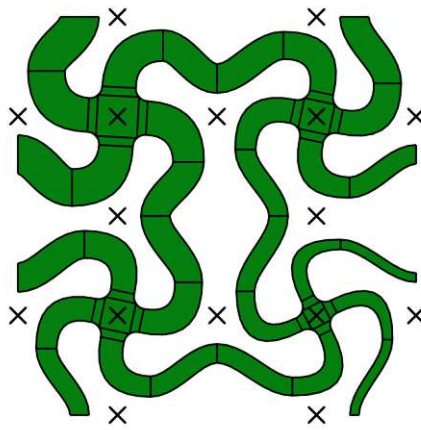


Abbildung 3.19: Punkte, an denen in der Einheitszelle Chi-Bar-Flex die Parameter abgelesen werden

In der Abbildung 3.20 werden mögliche Variationen der Parameter über die Einheitszelle gezeigt. Das soll das Potenzial der Wandelbarkeit dieser Einheitszelle zeigen.

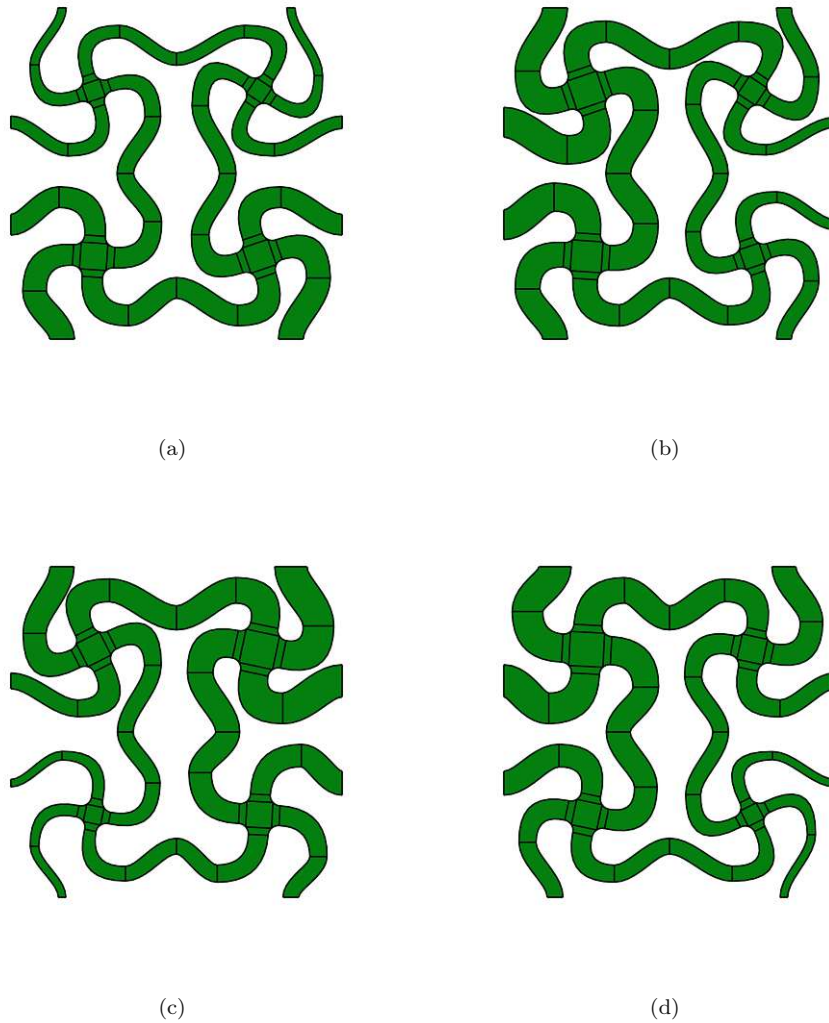


Abbildung 3.20: Einheitszelle Chi-Bar-Flex mit verschiedenen Parameterverteilungen

In diesem Aufbau der Einheitszelle ist die Position, an der die Arme der Chis an den Rand der Einheitszelle angrenzen, ebenfalls variabel. Diese Position setzt sich aus der Position des Arms nach dem inneren Spline ( $arm_{pos}$ ) und der Verschiebung von dort bis zum Rand ( $\delta_d$ ) zusammen. Jedoch werden diese beiden Größen ( $arm_{pos}$  und  $\delta_d$ ) nicht als Parameter genutzt. Der Grund dafür liegt in Problemen mit der Konnektivität zwischen den Einheitszellen der Gitterstruktur (siehe Kapitel 3.1.2).

Es gäbe jedoch die Möglichkeit, diese Größen auch noch als Parameter einzubetten und so eine noch größere Variabilität der Einheitszelle zu erreichen. Dieser Schritt ist in Teilen bereits durch die variabel gehaltenen Gleichungen der Positionen der Kontrollpunkte in Bezug auf diese beiden Größen vorbereitet. Sind die zusätzlichen Parameter für eine Anwendung zielführend, können diese hinzugefügt werden.

Der Grund, warum diese Parameter nicht gleich als solche definiert wurden, ist, dass durch so viele zusätzliche Parameter die Parameterstudie exponentiell aufwendiger wird. Die Parameter Dicke und Verdrehung wurden als die entscheidenden Größen hinsichtlich ihres Einflusses auf die mechanischen Eigenschaften identifiziert.

### 3.3 Simulationsergebnisse

In diesem Kapitel werden der Ablauf der Simulationen in MFEM erläutert und die ausgegebenen Ergebnisse betrachtet. Auf Basis dieser Ergebnisse werden anschließend die Netzfeinheits- und Parameterstudie durchgeführt. Die Simulation beschränkt sich hierbei auf die Geometrie einer einzigen Einheitszelle, die gleichzeitig die Referenzeinheitszelle ist. Das Ziel der Simulation ist die Berechnung der mechanischen Eigenschaften dieser Einheitszelle bei gegebener Geometrie. Diese sind: Poisson-Zahl, relative Dichte, maximale Spannung und Deformationskraft. Die Deformation ist eine Verschiebung der Knotenpunkte am östlichen Rand in positive x-Richtung (siehe Abbildung 3.22). Am nördlichen Rand wird die in Kapitel 2.2.1.5 vorgestellte Methode zur Aufbringung einfacher periodischer Randbedingungen verwendet. Die Verschiebung am südlichen Rand wird in y-Richtung und die Verschiebung des westlichen Randes in x-Richtung gesperrt.

Als Beispiel wird die Einheitszelle Chi-Bar-Flex simuliert. Die Randbedingungen und Materialeigenschaften, die in diesem Beispiel angewendet werden, sind im weiteren Verlauf der Arbeit bei allen Simulationen gleich. Als Material dient 3D-gedrucktes PLA. Die Materialeigenschaften von PLA sind in Lamé-Konstanten  $\lambda = 1200MPa$  und  $\mu = 1100MPa$ .

Die Randbedingungen werden vollautomatisiert aufgebracht, und die Verschiebung des östlichen Randes beträgt 10 Prozent der Ursprungslänge der Einheitszelle. Diese Verformung überschreitet die ertragbare Verformung eines gefertigten Bauteils. Es wird hier nur linear elastisches Materialverhalten simuliert, wodurch kleine Verschiebungen skalierbar sind. Eine Deformation von 10 Prozent wird in allen im Rahmen dieser Arbeit durchgeführten Simulationen beibehalten.

Zur Vorbereitung einer Simulation sind folgende Schritte zu erledigen. Die Gitterstruktur wird aus Einheitszellen zusammengesetzt. Die Anzahl der Einheitszellen, die die Gitterstruktur aufbauen, ist frei wählbar. Das Parameterfeld wird über die Gitterstruktur für jede Einheitszelle definiert. Falls keine Parameter gegeben sind, wird ein Default-Datenset an Parametern herangezogen, welches für alle Einheitszellen der Gitterstruktur dasselbe ist.

In den Simulationen dieser Arbeit wird eine einzelne Einheitszelle in einem Quadrat mit einer Seitenlänge von 10 mm simuliert. Für die Erstellung dieser Einheitszelle muss trotzdem eine Gitterstruktur samt Parameterfeld definiert werden. Die Gitterstruktur wird nur mit einer Einheitszelle über die gesamte Fläche befüllt und anschließend simuliert. Die Parameter sind über die Fläche hinweg unterschiedlich. In diesem Beispiel nimmt der Verdrehwinkel nach rechts oben ab, während die Dicke der Arme nach rechts oben zunimmt.

Im letzten Schritt wird das Finite-Elemente-Netz aufgebaut. Dafür wird nur die Eingabe der Anzahl der Elemente pro Splineparameter der Splines benötigt. In Abbildung 3.21 ist ein Finite-Elemente-Netz der Chi-Bar-Flex-Einheitszelle mit 29 Elementen pro Splineparameter dargestellt.

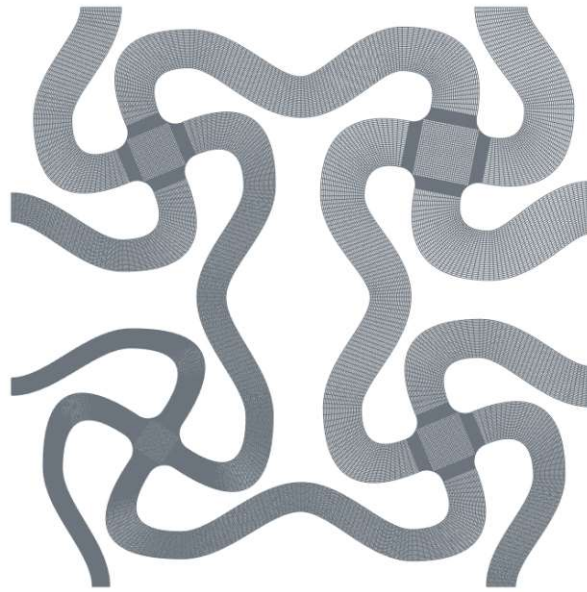
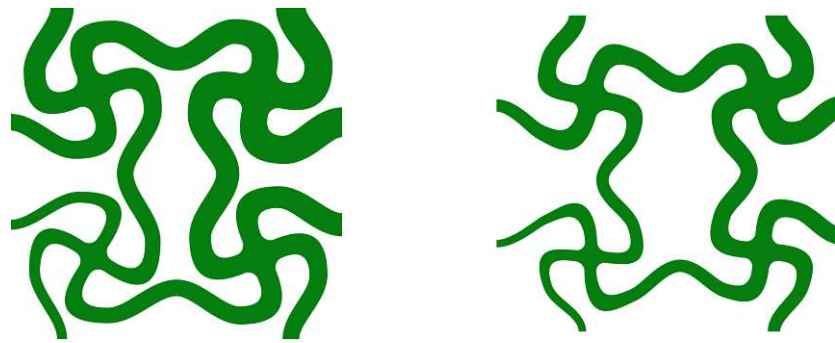


Abbildung 3.21: Finite-Elemente-Netz der Beispielsimulation

Somit sind alle Vorbereitungen für die Simulation getroffen. Die Netzdatei wird an MFEM übergeben und die Simulation startet. Hierbei gibt es in MFEM die Möglichkeit, den Grad der Elemente zu erhöhen. Die erstellte Netzdatei hat nur Knotenpunkte an den Ecken der Elemente und ist somit ein quadratisches Element erster Ordnung. MFEM kann dieses Netz automatisiert in ein Netz aus quadratischen Elementen zweiter Ordnung übersetzen. Das bedeutet, dass Knotenpunkte in der Mitte der vier Seiten der Elemente und im Mittelpunkt des Elements selbst hinzugefügt werden. Auch die Anzahl der Integrationspunkte erhöht sich von vier auf neun. Auch in der Simulation selbst wird daraufhin mit quadratischen Elementen zweiter Ordnung gearbeitet. Das Postprocessing wurde so aufgebaut, dass unabhängig davon, ob die Elemente erster oder zweiter Ordnung sind, die Ergebnisse richtig berechnet werden. In den Simulationen dieser Arbeit kommen quadratische Elemente zweiter Ordnung zum Einsatz.

Das Gleichungssystem wird erstellt, die Randbedingungen werden an den bei der Netzerstellung bereits gebildeten Knotenpunktgruppen definiert und in das Gleichungssystem eingebettet. Nach dem Lösen des Gleichungssystems sind die Verschiebungen aller Knotenpunkte im Netz, die durch die erzwungene Verschiebung verursacht werden, bekannt. Die Visualisierung dieser Verformung ist in Abbildung 3.22 zu sehen. Die Verschiebung wurde zur besseren Sichtbarkeit um das Vierfache vergrößert.



(a) Einheitszelle vor der Deformation

(b) Einheitszelle nach der Deformation

Abbildung 3.22: Simulationsergebnisse der Einheitszelle Chi-Bar-Flex nach Verformung

Mit den berechneten Verschiebungen werden im Postprocessing die Spannungen in der Einheitszelle berechnet. Aus diesen wird die durchschnittliche Von-Mises-Spannung pro Element berechnet. Diese Von-Mises-Spannungen werden automatisiert in einer Grafik auf der undeformierten Einheitszelle farblich dargestellt. Diese Grafik ist für unser Simulationsergebnis in Abbildung 3.23 dargestellt.

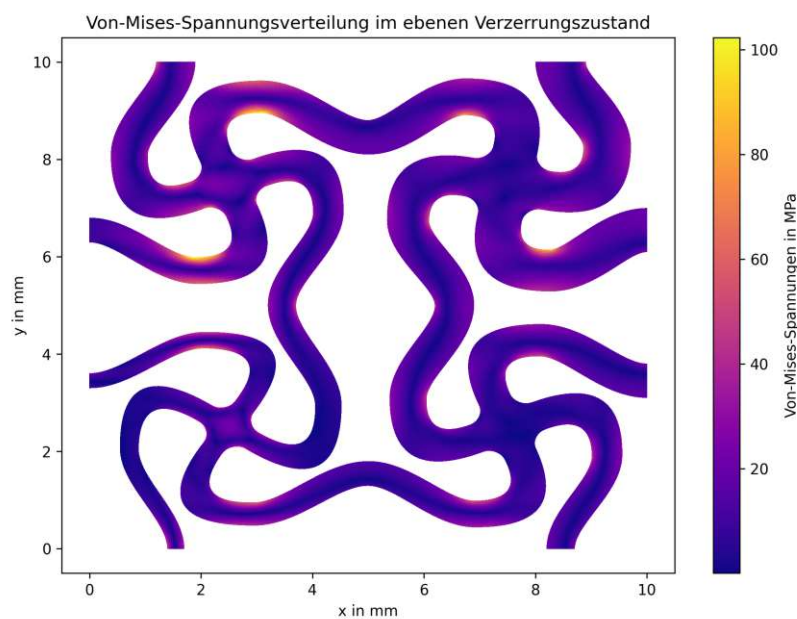


Abbildung 3.23: Verteilung der Von-Mises-Spannungen in der undeformierten Einheitszelle

Im letzten Schritt werden aus diesen Ergebnissen die mechanischen Eigenschaften berechnet. Diese Werte werden automatisch in ein Textdokument gespeichert. Die Informationen werden in Form einer Datenreihe der verschiedenen mechanischen Eigenschaften gespeichert. Diese Daten sind für verschiedene Simulationen in Grafiken abgebildet. Diese Grafiken werden für die Netzfeinheits- und Parameterstudie herangezogen. In einem zweiten Textfile wird auch noch ein anschaulicheres Format gespeichert, aus dem man die Werte händisch ablesen kann. Dort sind die Werte für unsere Simulation wie folgt abgespeichert.

Die Poissonzahl

-0.6995275991539582

Benötigte Gesamtkraft, die für die Deformation benötigt wird in N

7.031853484577633

Die maximale Von-Mises-Spannung im ebenen Verzerrungszustand in MPa

102.29663528309415

Fläche der Splines

44.643068020357454

Diese Werte sind unsere gesuchten mechanischen Eigenschaften, oder sie lassen sich einfach daraus berechnen, wie im Falle der relativen Dichte. Nach Definition der beschriebenen Werte erfolgen diese Simulationsabschnitte automatisiert. In der Netz- und Parameterstudie wird auch das Ändern der Startwerte automatisiert.

### 3.3.1 Vergleichssimulation in Abaqus

In der hier durchgeführten MFEM Simulation werden viele Aspekte der Erstellung des Gleichungssystems manuell durchgeführt, da Funktionen wie die periodische Randbedingung in MFEM nicht vorhanden sind. Um die in MFEM berechneten Simulationsergebnisse auf ihre Richtigkeit zu überprüfen, wird eine Vergleichssimulation in Abaqus erstellt.

Als Input für die Abaqus-Simulation wird die gleiche Netzdatei des Finite-Elemente-Netzes wie in MFEM verwendet. Die Dirichlet-Randbedingungen des Süd- und Westrandes werden in Abaqus gleich definiert wie in MFEM. Auch die Materialeigenschaften werden gleich gewählt. Dadurch wird eine vergleichbare Simulation garantiert. Die periodische Randbedingung am Nordrand und die erzwungene Verschiebung am Ostrand werden definiert. Nach der Durchführung beider Simulationen erfolgt ein Vergleich der Ergebnisse der Verschiebungen und der Postprocessing-Größen (Spannungen und Kennwerte der mechanischen Eigenschaften) mit den Werten der Simulation in MFEM.

In der Simulation mit Elementen erster Ordnung ergaben sich zwar für die mechanischen Eigenschaften vergleichbare Werte, jedoch unterschieden sich die Verschiebungen und die Spannungen lokal stark. Diese Unterschiede verringern sich, wenn quadratische Elemente zweiter Ordnung verwendet werden. Dort haben die Ergebnisse der Spannungen eine deutlich geringere Abweichung als in der Vergleichssimulation mit Elementen erster Ordnung.

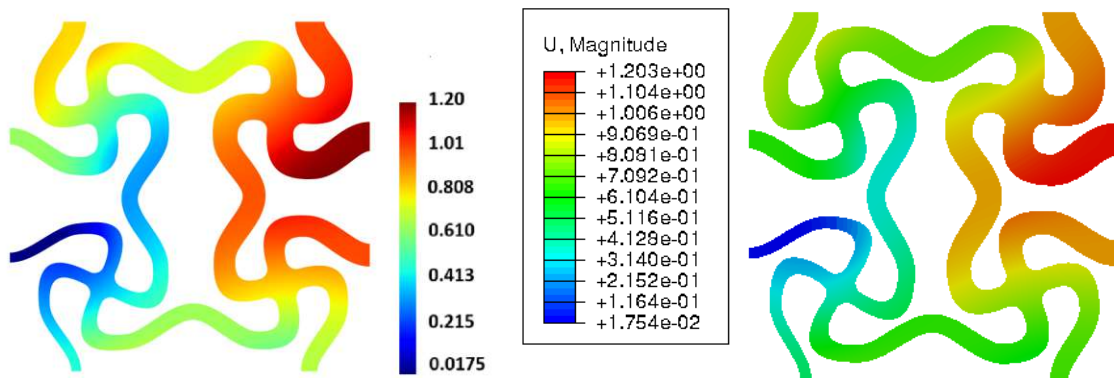
Die Ursache dieser Abweichungen liegt darin, dass das Gleichungssystem in MFEM genau so aufgebaut wird, wie in Kapitel 2.2.1.4 beschrieben. In Abaqus werden zusätzliche Änderungen an diesem Gleichungssystem durchgeführt, um die Ergebnisse akkurater und die Simulation effizienter zu gestalten. Welche Änderungen genau durchgeführt werden, ist schwer herauszufinden. In den Elementen erster Ordnung gibt es das Phänomen Shear-Locking. Shear-Locking tritt auf, wenn ein einzelnes quadratisch geformtes Element erster Ordnung auf Biegung beansprucht wird. In diesem Fall verhält sich das Element steifer als es die Materialeigenschaften vorgeben. Mithilfe der B-Bar-Methode kann der Einfluss des Shear-Lockings verringert werden. In Abaqus wird diese B-Bar-Methode im Hintergrund eingesetzt. Das ist der Grund, warum die Ergebnisse bei den Simulationen mit Elementen erster Ordnung abweichen [20, 27].

In den MFEM Simulationen, deren Ergebnisse in dieser Arbeit verwendet werden, werden nun ausschließlich quadratische Elemente zweiter Ordnung verwendet. Der Grund liegt auch hier an der besseren Übereinstimmung dieser Ergebnisse mit Abaqus. Die Annahme ist, dass eine kommerzielle Software ein

Ergebnis liefert, welches so nahe wie möglich an der Realität liegt. Das ist auch das Ziel unserer Simulation in MFEM und darum werden die Elemente zweiter Ordnung verwendet.

Es wird das Finite-Elemente-Netz aus Abbildung 3.21 für die Simulation sowohl in MFEM als auch in Abaqus verwendet. Es werden Elemente zweiter Ordnung simuliert. In MFEM wird das Netz automatisiert in ein Finite-Elemente-Netz zweiter Ordnung übersetzt. In Abaqus muss die Netzdatei manuell bearbeitet werden. Dazu wird mithilfe der Software Hyperworks die Netzdatei bearbeitet. In Hyperworks wird an den Grenzen der Elemente jeweils ein zusätzlicher Knotenpunkt hinzugefügt [2]. Das bedeutet, dass an den Elementen acht statt zuvor vier Knotenpunkte angrenzen. Diese neue Netzdatei wird im Format .inp exportiert. Netzdateien mit diesem Format können in Abaqus direkt geöffnet werden. Jetzt werden in Abaqus Elemente zweiter Ordnung simuliert.

Die Deformationen der Einheitszellen, die sich durch die Verformung ergeben, werden für beide Simulationen in Abbildung 3.24 dargestellt. Dabei wird die Gesamtverformung der jeweiligen Knoten farblich auf der Geometrie abgebildet. Die Werte der Verformung sind in mm angegeben.



(a) Deformation der Einheitszelle in mm in MFEM

(b) Deformation der Einheitszelle in mm in Abaqus

Abbildung 3.24: Vergleich der Ergebnisse der Deformationen in MFEM und Abaqus

Es ist zu beachten, dass die Farbskalen nicht vollkommen identisch sind. Wird dies berücksichtigt, sind die Verformungen über die Einheitszelle hinweg augenscheinlich gleich. Es werden zusätzlich die Verschiebungen der beiden Punkte verglichen, an denen die höchste Verschiebung zu erwarten ist.

Der östlichste Knoten am nördlichen Rand:

Verschiebung in MFEM:

In x-Richtung:

0.7166

In y-Richtung:

0.6995

Verschiebung in Abaqus:

In x-Richtung:

0.7167

In y-Richtung:

0.6995

Der nördlichste Knoten am östlichen Rand:

Verschiebung in MFEM:

In x-Richtung:

1.00000

In y-Richtung:

0.6375

Verschiebung in Abaqus:

In x-Richtung:

1.00000

In y-Richtung:

0.6375

Es zeigt sich, dass die Ergebnisse der Verschiebungen mit numerischer Genauigkeit exakt übereinstimmen.

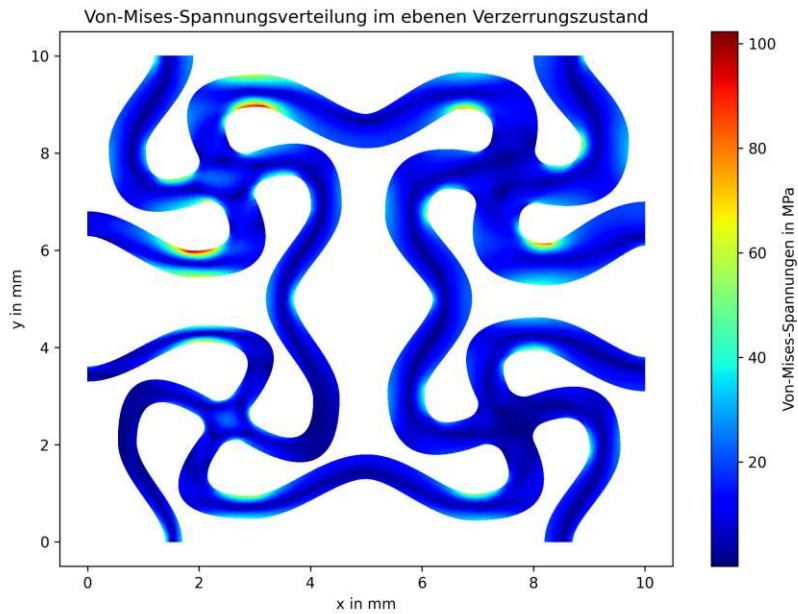
Die Simulationsergebnisse der Von-Mises-Spannungen werden in der Abbildung 3.25 für MFEM und Abaqus auf der undeformierten Einheitszelle farblich dargestellt. Das ermöglicht einen Vergleich der Ergebnisse der Von-Mises-Spannungen.

Werden auch hier die leicht unterschiedlichen Farbskalen berücksichtigt, so ist ersichtlich, dass die Spannungen innerhalb der Einheitszelle bei der Simulation in MFEM und Abaqus gleich verteilt sind. Ein auffälliger Unterschied in den abgebildeten Ergebnissen besteht darin, dass die Von-Mises-Spannung in Abaqus ein höheres Maximum erreicht als in MFEM. Dieser Unterschied ergibt sich daraus, dass in MFEM die Spannungen aller neun Integrationspunkte jedes Elements gemittelt werden. Diese gemittelte Von-Mises-Spannung wird in der Abbildung 3.25 (a) dargestellt. Abaqus extrapoliert die Werte der Steigung der Von-Mises-Spannung zum Rand hin. Das Maximum der Von-Mises-Spannung, das in Abbildung 3.25 (b) gegeben ist, ist also die maximale Spannung dieser Extrapolierten Spannungen.

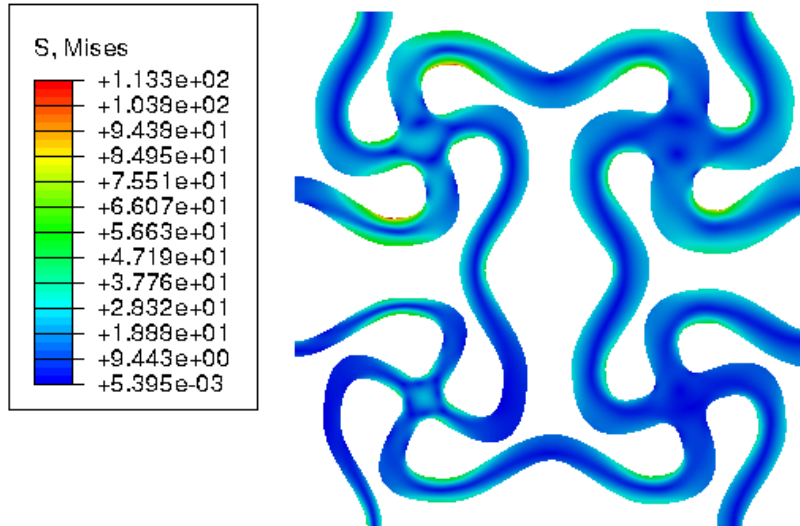
Um die Spannungen zu vergleichen, ist es erforderlich, die gemittelte Spannung der Elemente in Abaqus zu ermitteln. Der höchste Wert der durchschnittlichen Von-Mises-Spannung der Abaqus Simulation ergibt sich zu 103.72MPa. In der MFEM-Simulation ist die höchste gemittelte Von-Mises-Spannung bei 102.3MPa und ist damit sehr nahe am Ergebnis von Abaqus.

Werden die gemittelten Von-Mises-Spannungen an weiteren Elementen an verschiedenen Stellen der Einheitszelle verglichen, bewegen sich die prozentualen Abweichungen der Spannungen zwischen den Simulationen der meisten Elementen zwischen 0.4 und 6 Prozent. In manchen Elementen sind die Abweichungen jedoch auch mit bis zu 25 Prozent deutlich höher. Die höheren prozentualen Abweichungen kommen jedoch meist bei Elementen mit niedriger Spannung vor. Um zu untersuchen, wie die Abweichungen sich über die Einheitszelle hinweg verhält, wurden die gemittelten Von-Mises-Spannungen an 92 zufällig ausgewählten Elementen überprüft. Die gemittelte prozentuale Abweichung über diese Elemente liegt bei rund 1.36 Prozent. Daraus lässt sich schließen, dass einzelne Elemente Unterschiede in den berechneten Spannungen aufweisen, die Ergebnisse insgesamt jedoch ausreichend gut übereinstimmen.

Für die Abweichungen der Spannung in den Simulationen mit Elementen zweiter Ordnung gibt es zwei mögliche Begründungen. Es kann in Abaqus auch hier zu Änderungen im Gleichungssystem kommen, die die Effizienz oder Genauigkeit der Simulation verbessern. Ein anderer Grund könnte in den verwendeten Elementen zweiter Ordnung liegen. In MFEM werden für Simulationen mit Elementen zweiter Ordnung Quad9 Elemente verwendet. Diese Elemente haben neben den Knotenpunkten an den Elementecken auch an den Mittelpunkten der Kanten und im Mittelpunkt des Elements Knotenpunkte. Bei der Simulation



(a) Von-Mises-Spannungen der undeformierten Einheitszelle nach der Simulation in MFEM



(b) Von-Mises-Spannungen der undeformierten Einheitszelle nach der Simulation in Abaqus

Abbildung 3.25: Vergleich der Simulationsergebnisse der Von-Mises-Spannungen in MFEM und Abaqus

in Abaqus wurde bereits beschrieben, dass in Hyperworks nur Knotenpunkte in der Mitte der Kanten eingefügt werden und kein Knotenpunkt in der Mitte des Elements. Dadurch ergibt sich ein Quad8 Element. Es ist möglich, dass durch diese unterschiedliche Definition von Elementen zweiter Ordnung der Unterschied in den Ergebnissen der Spannungen entsteht.

Insgesamt passen die Verschiebungen und Spannungen ausreichend gut zusammen. Aus diesen Größen werden die mechanischen Eigenschaften berechnet. Darum kann davon ausgegangen werden, dass auch Kennwerte wie die benötigte Deformationskraft und die Poissonzahl ausreichend akkurat bestimmt werden. Die Berechnung dieser Größen ist unser Simulationsziel. Darum wird die MFEM-Simulation als korrekte Simulation der Einheitszelle verwendet.

## 3.4 Netzfeinheitsstudien

In diesem Kapitel werden die Netzfeinheitsstudien für die Einheitszellen durchgeführt. Diese Netzfeinheitsstudien dienen dazu, die optimale Netzfeinheit für die Simulation der Einheitszelle zu finden. Die optimale Netzfeinheit hält das Gleichgewicht zwischen der benötigten Simulationszeit und der Genauigkeit der simulierten Ergebnisse. Mit einem feineren Netz spiegeln die Simulationsergebnisse die Realität genauer wieder. Je mehr Elemente verwendet werden, desto größer wird das zu lösende Gleichungssystem, wodurch sich eine längere Simulationszeit ergibt.

Wie lange eine vertretbare Zeit pro Simulation ist, hängt von der Anzahl der durchzuführenden Simulationen ab. Im weiteren Verlauf dieser Arbeit werden die Simulationen dazu genutzt, eine Parameterstudie für mehrere Einheitszellen durchzuführen. Dazu sind pro Einheitszelle etwa 100 Simulationen nötig. Diese Information muss auch bei der Netzfeinheitsstudie berücksichtigt werden und wirkt sich limitierend auf die mögliche Simulationszeit aus.

Um herauszufinden, wie akkurat ein Simulationsergebnis ist, wird in der Netzfeinheitsstudie ein immer feiner werdendes Netz simuliert. Dabei konvergieren die Ergebnisse mit feiner werdendem Netz zu einem Wert hin. Wenn die Netzfeinheit beliebig groß wäre, ist das Ergebnis exakt. Daraus lässt sich schließen, dass dieser Wert, gegen den die Simulationsergebnisse konvergieren, das exakte Ergebnis ist.

In diesen Netzfeinheitsstudien werden die mechanischen Eigenschaften Poissonzahl, Deformationskraft und maximale Von-Mises-Spannung berechnet. Da diese Werte unsere gewünschten Simulationsergebnisse sind, werden diese auch als Kennwerte für die Parameterstudie verwendet. Es wird also eine Einheitszelle mit gleichbleibenden Parametern zuerst grob vernetzt und die Simulation durchgeführt. Schritt für Schritt wird die Netzfeinheit erhöht und die Simulationsergebnisse werden gespeichert. Diese Ergebnisse werden hier gezeigt. Die Geometrie der Einheitszelle bleibt bei der Änderung der Netzfeinheit gleich. Welche Parameter dabei gewählt wurden, ist von untergeordneter Bedeutung. Aus diesem Grund wird auf die genaue Geometrie nicht eingegangen.

### 3.4.1 Netzfeinheitsstudie Chi-Bar

Als erstes wird die Netzfeinheitsstudie der Einheitszelle Chi-Bar durchgeführt. In der Tabelle 3.1 sind die Ergebnisse der Simulationen eingetragen.

Elemente pro Splineparameter	Poissonzahl	Deformationskraft in N	max. Spannung in MPa	Simulationszeit in s
4	-0.7257	6.029	43.25	1
9	-0.7228	6.037	51.85	3
14	-0.7225	6.038	55.72	7
19	-0.7224	6.038	57.41	13
24	-0.7224	6.038	58.66	22
29	-0.7224	6.038	59.32	36
34	-0.7224	6.038	59.89	52
39	-0.7224	6.038	60.27	76
44	-0.7224	6.038	60.56	104

Tabelle 3.1: Ergebnisse Netzfeinheitsstudie Chi-Bar

Zur besseren Visualisierung werden die Daten in Abbildung 3.26 in einem Graphen dargestellt. Dabei sind die Werte zueinander in Relation gesetzt, um ein klares Bild zu bekommen, wie die einzelnen Werte im Vergleich zueinander konvergieren.

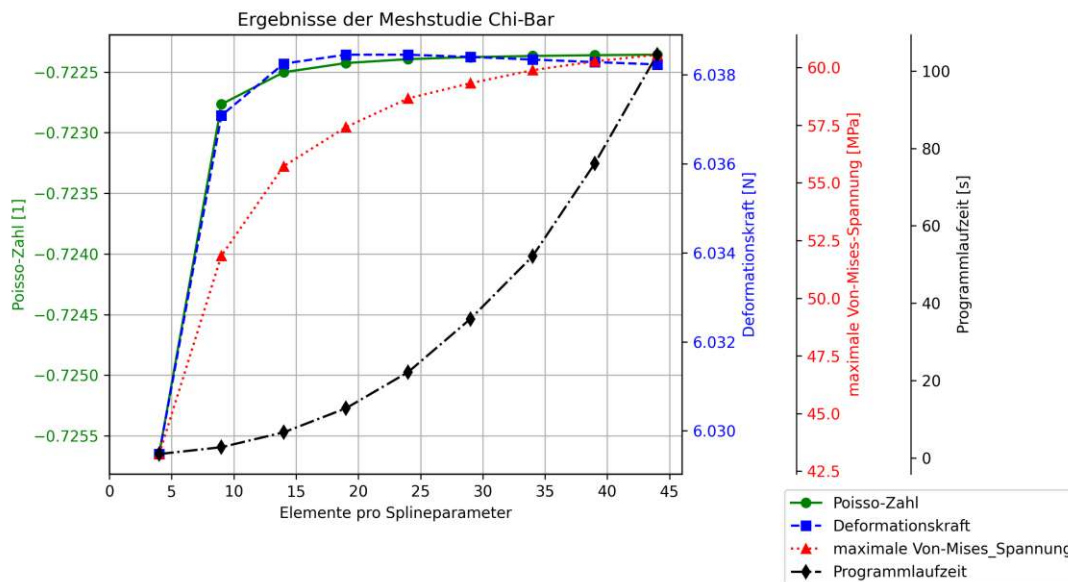


Abbildung 3.26: Graph der Daten der Netzfeinheitsstudie Chi-Bar

Aus dem Graphen in Abbildung 3.26 kann die optimale Anzahl an Elementen pro Splineparameter abgelesen werden. In diesem Fall werden 34 Elemente pro Splineparameter als Optimum gewählt, da die Simulationszeit von 52 Sekunden für die Parameterstudie angemessen ist und die Simulationsergebnisse bereits sehr gut konvergieren.

### 3.4.2 Netzfeinheitsstudie Chi-Bar-Flex

Die Ergebnisse der Netzfeinheitsstudie der Einheitszelle Chi-Bar-Flex sind in Tabelle 3.2 aufgelistet.

Elemente pro Splineparameter	Poissonzahl	Deformationskraft in N	max. Spannung in MPa	Simulationszeit in s
4	-0.7639	13.203	73.90	1
9	-0.7582	13.478	103.41	3
14	-0.7575	13.546	114.36	6
19	-0.7572	13.571	121.17	10
24	-0.7571	13.582	125.25	18
29	-0.7570	13.588	127.93	26
34	-0.7570	13.592	129.81	39
39	-0.75699	13.594	131.19	55
44	-0.75697	13.596	132.54	71

Tabelle 3.2: Ergebnisse Netzfeinheitsstudie Chi-Bar-Flex

Die Daten aus Tabelle 3.2 werden im Graph 3.27 im Verhältnis zueinander in Grafiken dargestellt.

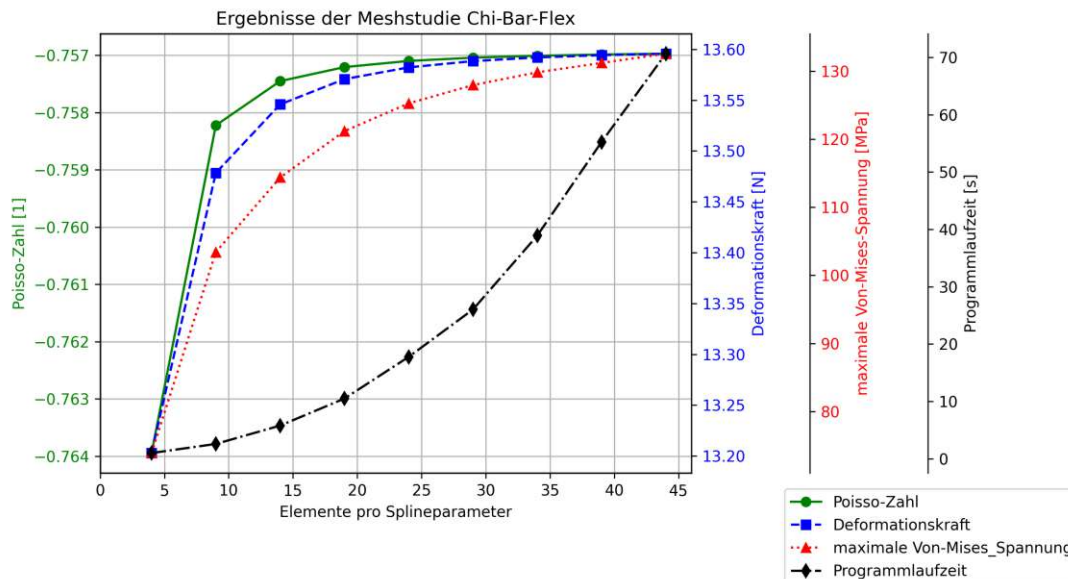


Abbildung 3.27: Graph der Daten der Netzfeinheitsstudie Chi-Bar-Flex

Die optimale Elementanzahl pro Splineparameter wurde mit 34 Elementen gewählt.

### 3.4.3 Netzfeinheitsstudie Chi

In Tabelle 3.3 sind die Ergebnisse der Simulationen der Netzfeinheitsstudie für die Einheitszelle Chi eingetragen.

Elemente pro Splineparameter	Poissonzahl	Deformationskraft in N	max. Spannung in MPa	Simulationszeit in s
4	0.7828	106.477	177.69	1
9	0.7844	77.554	216.26	3
14	0.7853	68.761	235.79	6
19	0.7858	64.461	266.46	12
24	0.7862	61.882	298.05	20
29	0.7865	60.153	326.44	32
34	0.7866	58.909	352.45	48
39	0.7868	57.970	376.59	70
44	0.7869	57.234	399.22	96

Tabelle 3.3: Ergebnisse Netzfeinheitsstudie Chi

Die in Tabelle 3.3 eingetragenen Werte werden im Graphen in Abbildung 3.28 im Verhältnis zueinander dargestellt.

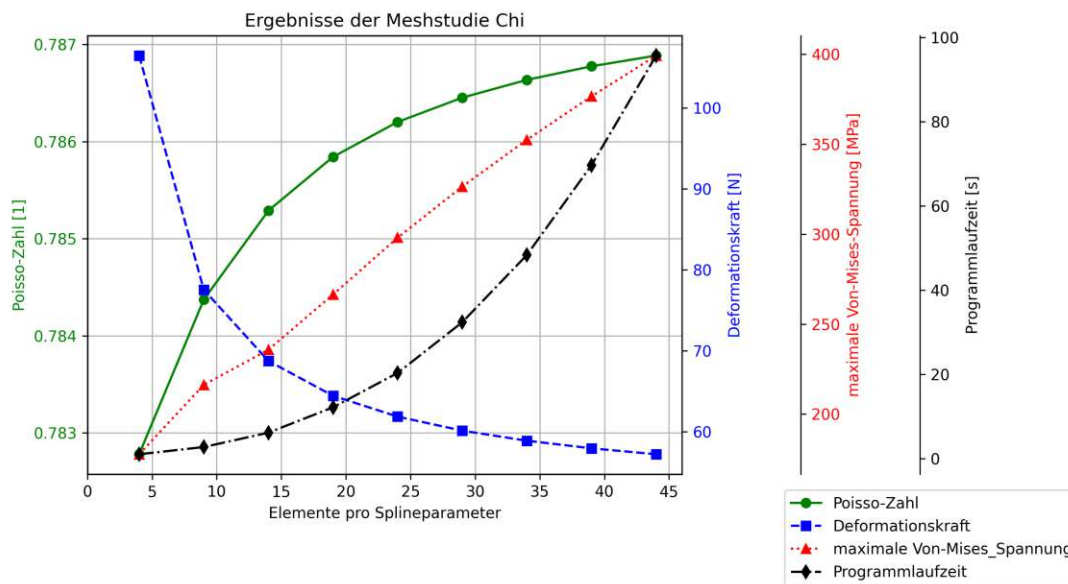


Abbildung 3.28: Graph der Daten der Netzfeinheitsstudie Chi

Im Vergleich mit den anderen beiden Einheitszellen zeigt der Graph in Abbildung 3.28, dass die maximale Von-Mises-Spannung nicht konvergiert. Grund dafür sind die nicht abgerundeten Kanten. Dieses Phänomen ist in Kapitel 3.1.3 genauer beschrieben. Das bedeutet, dass der Datensatz der maximalen Von-Mises-Spannungen bei der Ermittlung der optimalen Netzfeinheit ausgeschlossen werden muss.

Bei Betrachtung der benötigten Deformationskraft, um die Verformung zu erzielen, sieht man eine Konvergenz. Dabei nimmt jedoch im Vergleich zu den anderen beiden Simulationen die Kraft mit zunehmender Netzfeinheit ab. Das kann verschiedene Gründe haben.

Die Deformationskraft wird aus den Spannungen der Elemente, die am östlichen Rand angrenzen, berechnet. Wie genau und warum die Kraft auf diese Art berechnet werden muss, ist in Kapitel 3.1.3 erläutert. An den Enden der Arme der Chis ist auch eine spitze Kante vorhanden, an der die Spannung nicht korrekt berechnet wird. Da nun die Kraft aus diesen Spannungen berechnet wird, muss also trotz der

Konvergenz der Werte auch dieser Datensatz als unbrauchbar eingestuft und für die Netzfeinheitsstudie ausgeschlossen werden.

Der Datensatz der Poissonzahl ist der einzige, der für die Netzfeinheitsstudie brauchbar ist. Mit den Simulationsergebnissen der Poissonzahl wird die optimale Netzfeinheit gesucht. Auch hier werden 34 Elemente pro Splineparameter für die Simulationen der Parameterstudie gewählt.

## 3.5 Parameterstudien

Die Parameterstudie wird hier für die Einheitszellen Chi, Chi-Bar und Chi-Bar-Flex durchgeführt. Dabei wird untersucht, welchen Einfluss die Parameter auf die mechanischen Eigenschaften haben. In den drei gewählten gibt es die Dicke und den Verdrehwinkel als Parameter. Diese Parameter können auch innerhalb der Einheitszelle variiert werden.

Um eine Parameterstudie durchzuführen, müssen die Parameter in kleinen Schritten geändert werden. Über diese Schritte hinweg werden die mechanischen Eigenschaften gespeichert. Durch die feine Unterteilung der Parameter kann im Gesamtbild aller Simulationen die Tendenz der mechanischen Eigenschaften bei Änderungen der Parameter untersucht werden.

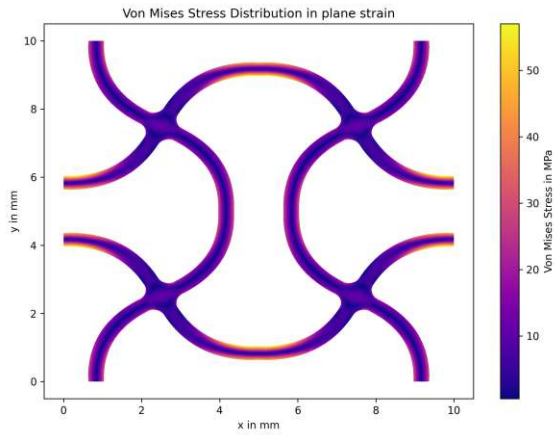
Um die Anzahl der Simulationen in einem bewältigbaren Rahmen zu halten, wird auf Parameteränderungen innerhalb der Einheitszelle verzichtet. Das bedeutet, dass die Dicke der Arme der Chis und die Verdrehwinkel dieser an allen Stellen gleich bleiben. Dadurch können diese beiden gleichbleibenden Parameter in feinen Schritten simuliert werden.

Die mechanischen Eigenschaften werden mithilfe von Konturdiagrammen dargestellt. Das Ziel dieser Konturdiagramme ist es, den Einfluss der beiden Parameter auf die mechanischen Eigenschaften abzubilden. Auf den Achsen ist die Variation der Parameter aufgetragen. Die Ergebnisse der mechanischen Eigenschaften sind farblich dargestellt. In dieser Weise wird für jede mechanische Eigenschaft ein Konturdiagramm erstellt.

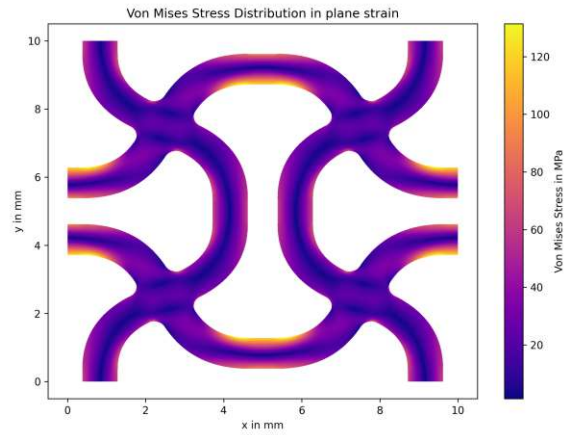
### 3.5.1 Parameterstudie Chi-Bar

Für jede Einheitszelle muss separat überprüft werden, in welchen Grenzen sich die Parameter bewegen sollen. Dabei ist darauf zu achten, dass der Flächenverbund nicht über die Grenzen der Einheitszelle hinausragt. Auch Überlappungen der Splines der Einheitszelle sind nicht zulässig. Es muss auch eine Untergrenze für beide Parameter definiert werden. Bei der unteren Grenze der Dicke ist durch die Fertigung eine Grenze gesetzt. Bei der Fertigung solcher Einheitszellen mit additiven Fertigungsmethoden ist eine minimal fertigbare Dicke gegeben.

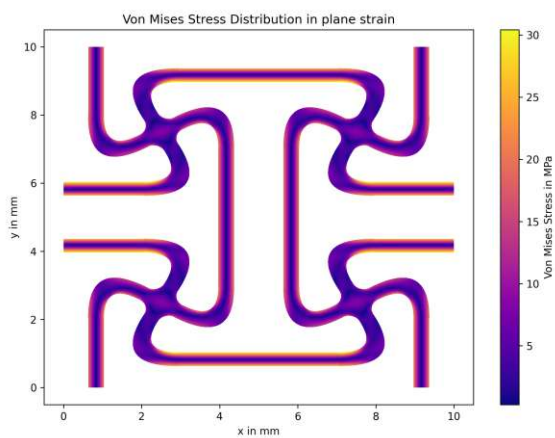
Im Falle der Einheitszelle Chi-Bar wird die Dicke der Arme der Chis zwischen 0.4mm und 0.9mm gewählt. Die Abmaße der Einheitszelle sind mit 10mm die Seitenlänge des Quadrates. Der Verdrehungswinkel  $\beta$  wird im Bereich von  $-\frac{\pi}{6}$  bis  $\frac{\pi}{6}$  gewählt. Für die vier Simulationen, die die Randwerte dieser Parametergrenzen abbilden, wurde die undeformierte Einheitszelle mit den berechneten Von-Mises-Spannungen in Abbildung 3.29 dargestellt.



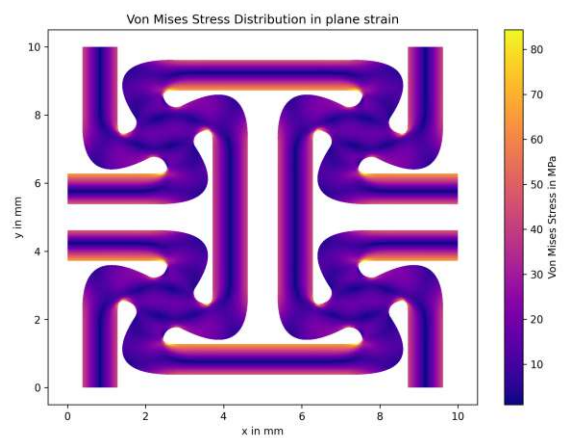
(a) Chi-Bar:  $t = 0.4\text{mm}$ ,  $\beta = -\frac{\pi}{6}$



(b) Chi-Bar:  $t = 0.4\text{mm}$ ,  $\beta = \frac{\pi}{6}$



(c) Chi-Bar:  $t = 0.9\text{mm}$ ,  $\beta = -\frac{\pi}{6}$



(d) Chi-Bar:  $t = 0.9\text{mm}$ ,  $\beta = \frac{\pi}{6}$

Abbildung 3.29: Grenzparameter der Parameterstudie für die Einheitszelle Chi-Bar

Bei der Dicke wurde der Wert in 0.05mm Schritten erhöht. Dadurch ergeben sich 11 Dicken, die simuliert werden. Beim Verdrehwinkel  $\beta$  wird der Winkel von  $\frac{\pi}{6}$  bis  $-\frac{\pi}{6}$  in 11 Winkelparameter unterteilt und simuliert. In Summe werden also 121 Simulationen durchgeführt, bei denen jeweils die vier oben genannten mechanischen Eigenschaften berechnet werden. Diese Daten werden im Anhang 4.3 in einer Tabelle aufgelistet.

Zur Veranschaulichung dieser Daten werden sie in der Abbildung 3.30 in Konturdiagrammen dargestellt. In diesen Konturdiagrammen sind die Ergebnisse der mechanischen Eigenschaften über die beiden Parameter farblich dargestellt.

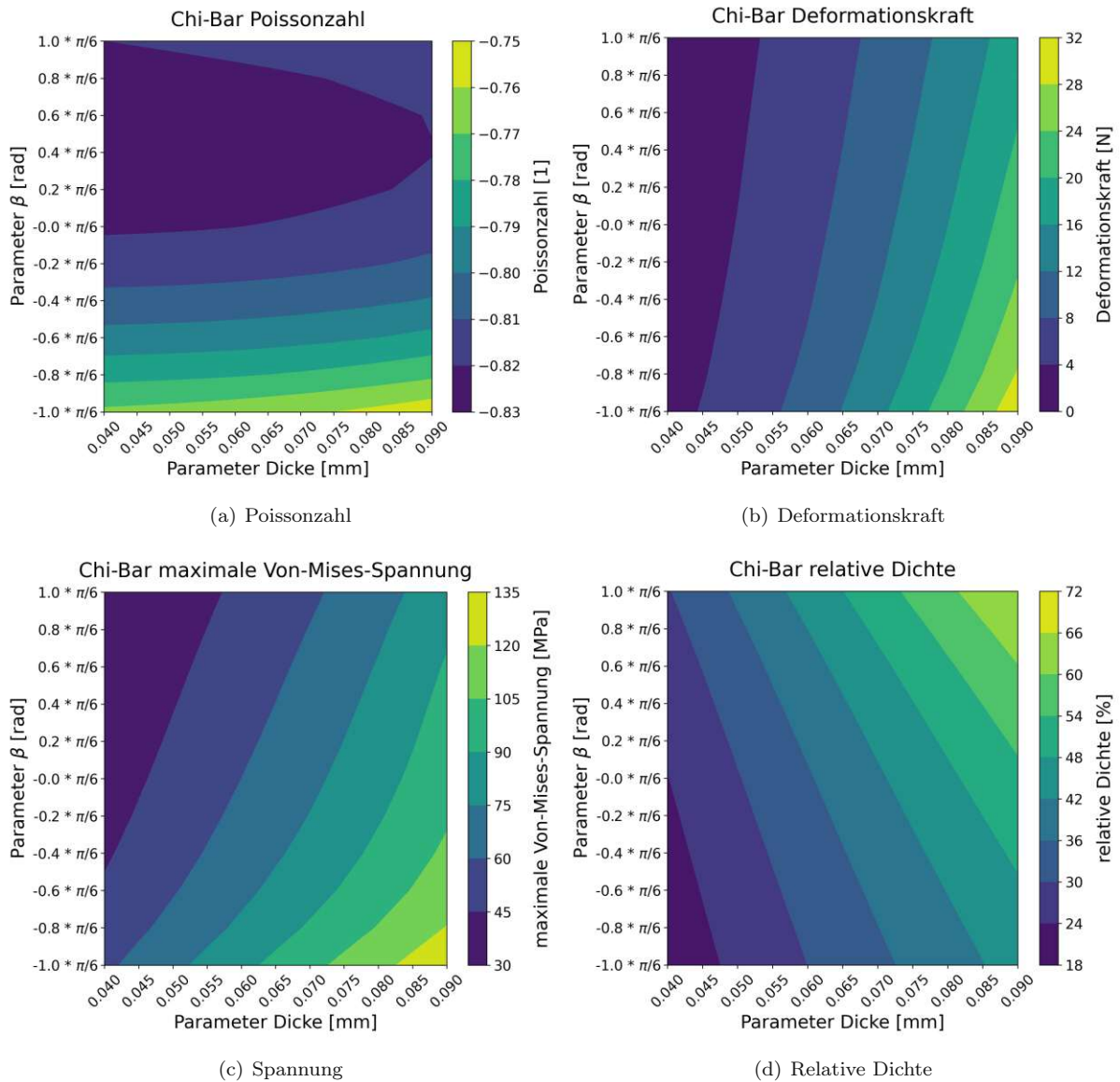
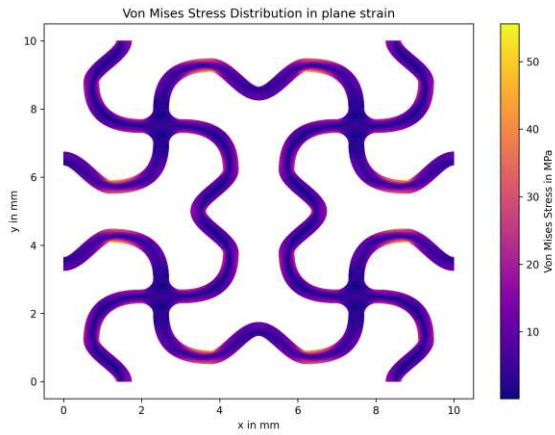


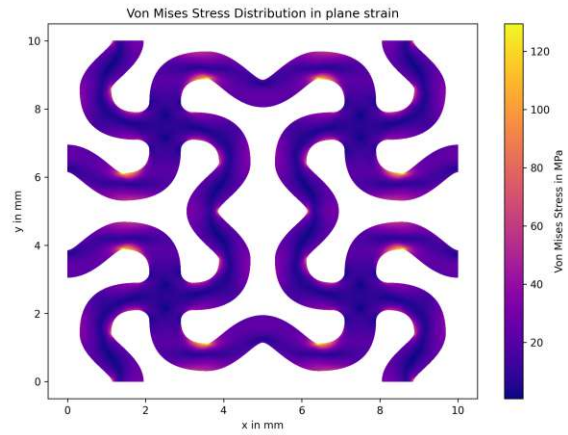
Abbildung 3.30: Konturdiagramme mit allen mechanischen Eigenschaften der Parameterstudie von Chi-Bar

### 3.5.2 Parameterstudie Chi-Bar-Flex

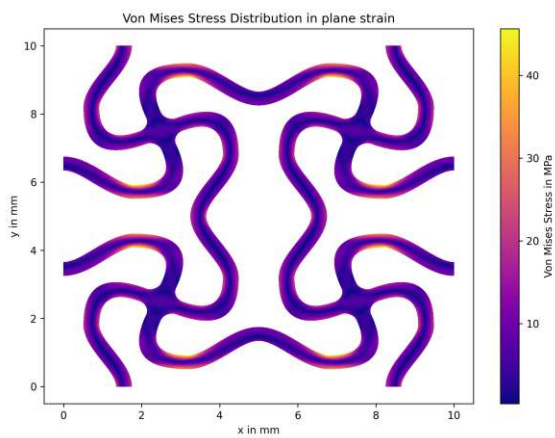
Für die Einheitszelle Chi-Bar-Flex werden die Grenzen der Parameter für die Parameterstudie wie folgt festgelegt. Die Dicke der Arme der Chis werden zwischen 0.4mm und 0.8mm gewählt. Die Abmaße der Einheitszelle sind 10mm Seitenlänge des Quadrates. Der Verdrehungswinkel  $\beta$  soll im Bereich von 0 bis  $\frac{\pi}{6}$  simuliert werden. Diese Grenzfälle der Parameter wurden undeformiert mit simulierten Von-Mises-Spannungen in Abbildung 3.31 abgebildet.



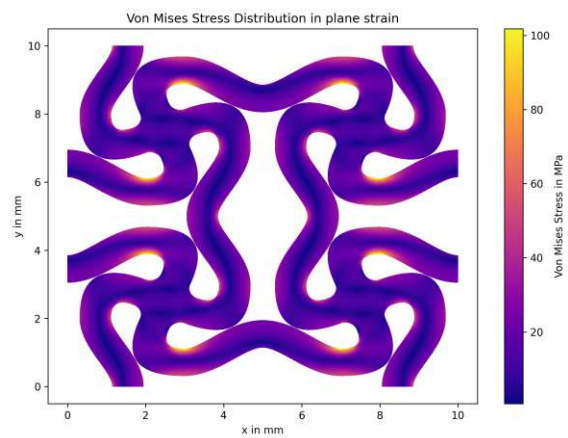
(a) Chi-Bar-Flex:  $t = 0.4mm$ ,  $\beta = 0$



(b) Chi-Bar-Flex:  $t = 0.8mm$ ,  $\beta = 0$



(c) Chi-Bar-Flex:  $t = 0.4mm$ ,  $\beta = \frac{\pi}{8}$



(d) Chi-Bar-Flex:  $t = 0.8mm$ ,  $\beta = \frac{\pi}{8}$

Abbildung 3.31: Grenzparameter der Parameterstudie für die Einheitszelle Chi-Bar-Flex

Die Parameterschritte sind im Dickenparameter 0.05mm weit, wodurch sich neun Simulationen für die Dickenparameter ergeben. Der Verdrehwinkel  $\beta$  wird von 0 bis  $\frac{\pi}{8}$  mit 11 Winkelparametern simuliert. Es ergeben sich 99 Simulationen für diese Parameterstudie. Alle Ergebnisse dieser Parameterstudie werden in der Tabelle im Anhang 4.3 aufgeführt. Diese Daten sind in Abbildung 3.32 in Konturdiagrammen dargestellt.

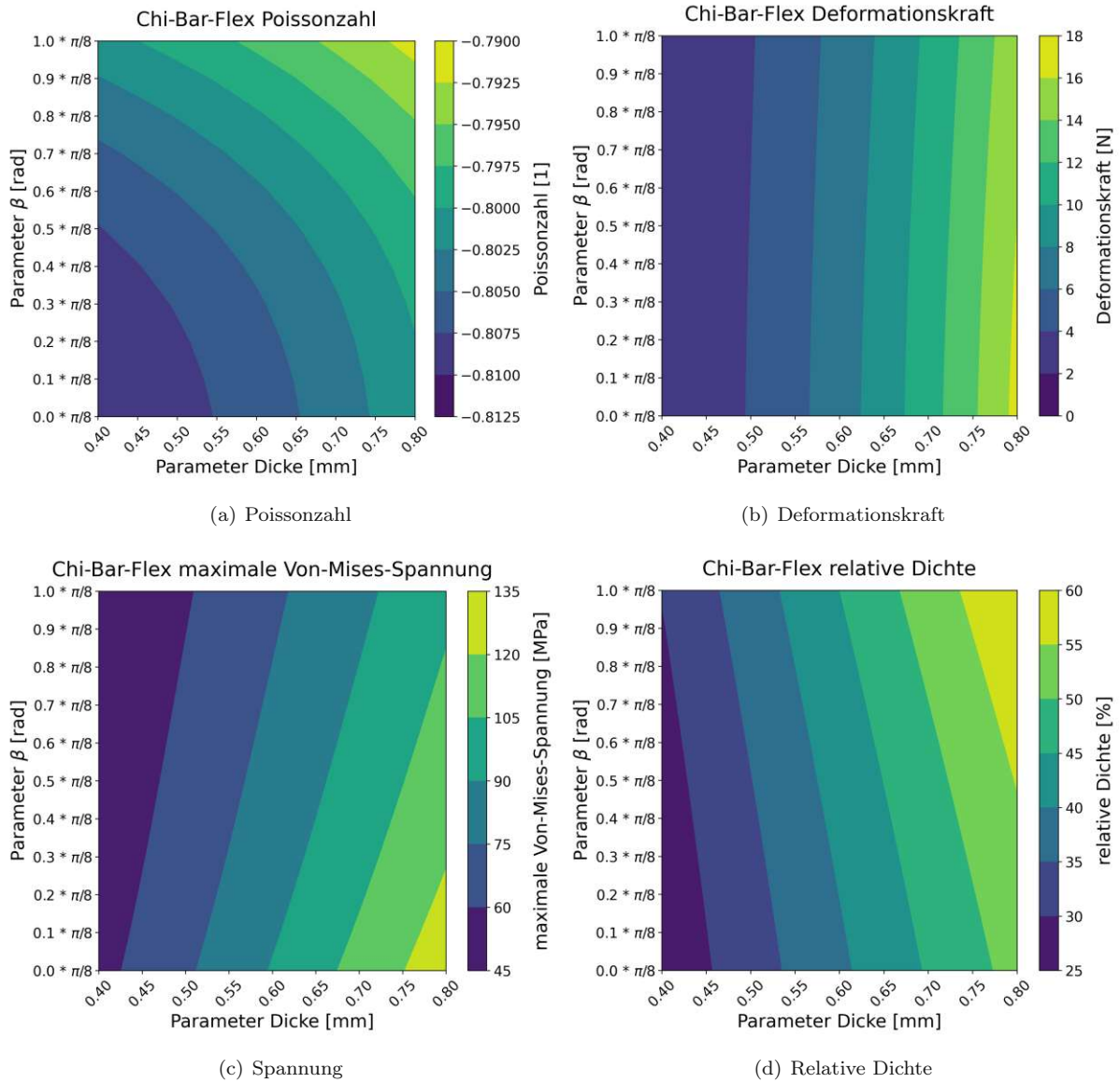


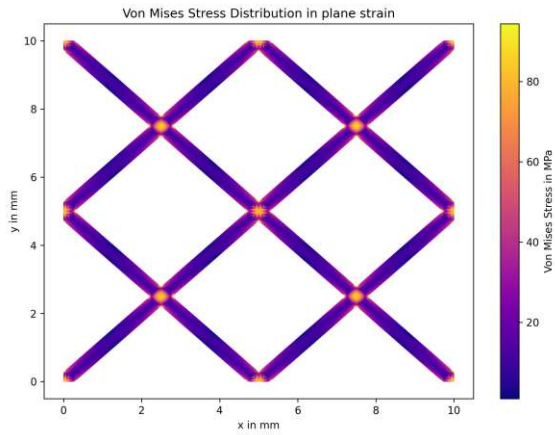
Abbildung 3.32: Konturdiagramme mit allen mechanischen Eigenschaften der Parameterstudie von Chi-Bar-Flex

### 3.5.3 Parameterstudie Chi

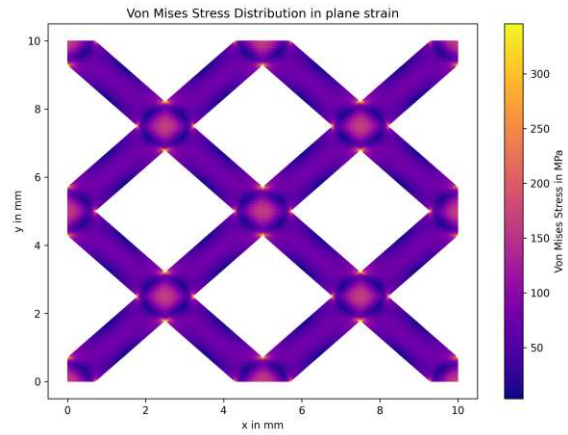
Für die Einheitszelle Chi wurden die Grenzen des Dickenparameters zwischen 0.4mm und 1mm gewählt. Auch hier hat die Einheitszelle eine Seitenlänge von 10mm.

Beim Verdrehwinkel wurden nur positive Winkel verwendet. Der Grund dafür ist, dass bei gleichbleibenden Parametern die Einheitszelle bei gleicher Dicke und demselben Winkelbetrag mit sich unterscheidenden Vorzeichen nur gespiegelt wird. Dementsprechend werden auch die mechanischen Eigenschaften dieser beiden Parameterkombinationen genau gleiche Ergebnisse liefern. Der Winkelparameter wird im Bereich von 0 bis  $\frac{\pi}{2}$  simuliert.

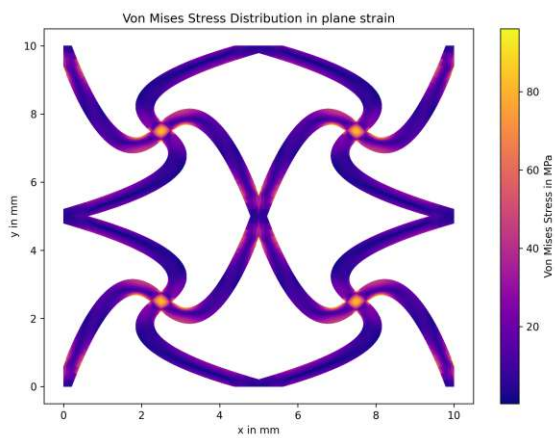
Die Grenzfälle der Parameter werden in Abbildung 3.33 dargestellt. Dabei ist auf der undeformierten Geometrie der Einheitszelle die Von-Mises-Spannungen abgebildet.



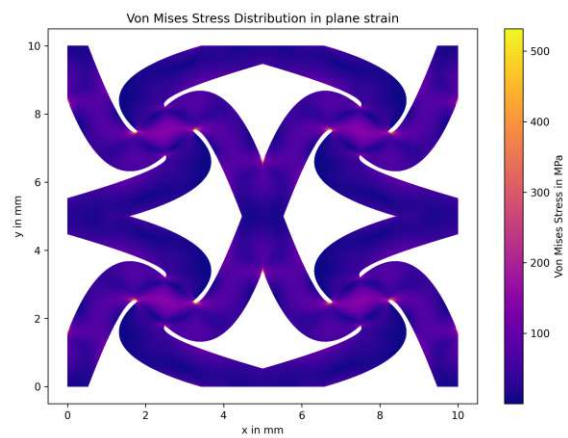
(a) Chi:  $t = 0.4mm$ ,  $\beta = 0$



(b) Chi-Bar-Flex:  $t = 1mm$ ,  $\beta = 0$



(c) Chi-Bar-Flex:  $t = 0.4mm$ ,  $\beta = \frac{\pi}{2}$



(d) Chi-Bar-Flex:  $t = 1mm$ ,  $\beta = \frac{\pi}{2}$

Abbildung 3.33: Grenzparameter der Parameterstudie für die Einheitszelle Chi

Die Parameterschritte betragen im Dickenparameter  $0.05mm$ . Der Winkelparameter  $\beta$  wird zwischen  $0$  und  $\frac{\pi}{2}$  mit  $9$  Werten simuliert. Dadurch werden  $13$  Dickenparameter mit jeweils  $9$  Winkelparametern simuliert. Es werden also für diese Parameterstudie  $117$  Simulationen durchgeführt.

Alle Ergebnisse dieser Parameterstudie sind im Anhang 4.3 in einer Tabelle zusammengefasst. Abbildung 3.34 zeigt diese Daten auch in einem Konturdiagramm.

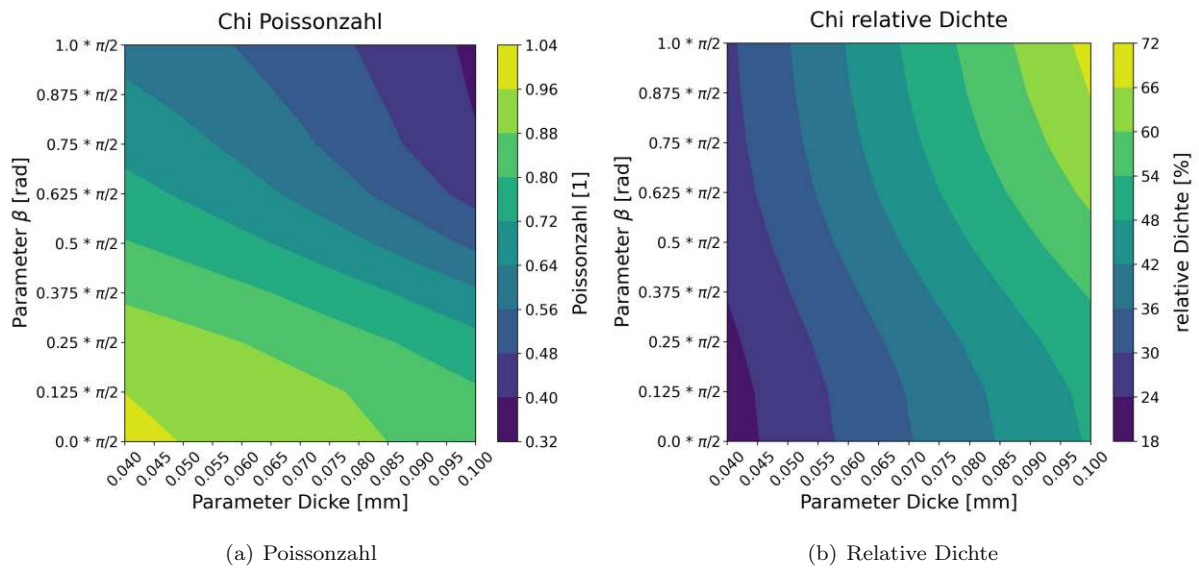


Abbildung 3.34: Konturdiagramme mit allen mechanischen Eigenschaften der Parameterstudie von Chi

Wie bereits aus Kapitel 3.1.3 hervorgeht, sind die Ergebnisse der maximalen Von-Mises-Spannung und der aus den Spannungen berechneten Deformationskraft nicht repräsentativ und wurden deshalb nicht dargestellt.

# Kapitel 4

## Ergebnisdiskussion

In diesem Kapitel werden die Ergebnisse der Simulationen nochmals genauer dargestellt und eingeordnet.

### 4.1 Einfluss der Parameter auf die mechanischen Eigenschaften

Die Ergebnisse der Parameterstudien werden untersucht. Ziel ist es für die Einheitszellen Zusammenhänge zwischen den Parametern und den mechanischen Eigenschaften zu erkennen. Diese Zusammenhänge werden analysiert und das Potenzial der Variabilität der Einheitszelle wird herausgearbeitet.

Da ein Ziel der Erstellung dieser Einheitszelle die Steuerung der Verformbarkeit und der Steifigkeit war, wird hier besonderes Augenmerk auf die Poissonzahl und die benötigte Deformationskraft gelegt. Die Poissonzahl spiegelt das Verhalten bei Verformung wieder. Die Deformationskraft zeigt, wie steif die Einheitszelle ist. Um mit den Daten, die in den Konturdiagrammen in Kapitel 3.5 bereits dargestellt wurden, eine genaue Aussage treffen zu können, werden diese nach mechanischen Eigenschaften getrennt gezeigt und untersucht. Aus diesen Konturdiagrammen können die Einflüsse der Parameter abgelesen werden.

#### 4.1.1 Einfluss der Parameter in Einheitszelle Chi-Bar

Die folgenden Abbildungen 3.30 (a), 3.30 (b), 3.30 (c) und 3.30 (d) zeigen die Änderungen einer mechanischen Eigenschaft bei Variation der Parameter. In diesen Konturdiagrammen wird auf den Achsen jeweils ein Parameter variiert und die Größe der untersuchten mechanischen Eigenschaften werden farblich dargestellt.

In der Abbildung 3.30 (a) ist die Poissonzahl der Einheitszelle Chi-Bar in einem Konturdiagramm zu sehen. Die Änderung des Winkels  $\beta$  hat einen starken Einfluss auf die Poissonzahl.

Im Bereich  $\beta = -\frac{\pi}{6}$  bis  $\beta = 0$  des Winkelparameters nimmt die Poissonzahl mit steigendem Winkel stark ab. Für Anwendungen in der Optimierung von Gitterstrukturen, bei denen bei gegebener Belastung eine bestimmte Verformung vorliegen soll, hat dieser Parameterbereich das größte Potenzial, diese Eigenschaften zu variieren. Im Bereich, in dem der Winkel positiv ist, steigt die Poissonzahl wieder leicht an.

Die Dicke hat einen geringen Einfluss auf die Poissonzahl. Hier steigt die Poissonzahl bei allen Winkeln annähernd linear mit der Dicke.

Im Konturdiagramm 3.30 (b) ist zu sehen, dass der Verdrehungswinkel einen geringen Einfluss auf die Deformationskraft hat. Mit Erhöhung des Winkels sinkt die Deformationskraft und bei hohen Dickenparametern ist ein steilerer Abfall der Kraft mit einer Vergrößerung des Winkels ersichtlich. Das deutet daher, dass eine Versteifung der Struktur erfolgt, wenn der Winkel niedriger ist. Diese Versteifung ist jedoch bei den dünnen Einheitszellen deutlich weniger ausgeprägt.

Die Dicke hat einen sehr starken Einfluss auf die Kraft. Je höher die Dicke der Einheitszellen, desto mehr Kraft wird für die Verformung benötigt.

Im Konturdiagramm in Abbildung 3.30 (d) ist zu sehen, dass der Anteil, den der Flächenverbund in der Einheitszelle einnimmt, mit steigender Dicke und Winkel annähernd linear wächst.

Die maximale Von-Mises-Spannung, die in der Einheitszelle entsteht, ist im Konturdiagramm in Abbildung 3.30 (c) zu sehen. Die Spannung wächst bei steigender Dicke annähernd linear. Bei steigendem Winkel fällt sie ab. Das deutet darauf hin, dass mit steigender Dicke die Einheitszelle steifer wird. Bei steigendem Winkel nimmt die Steifigkeit der Einheitszelle ab.

Eine mögliche Erklärung ist, dass durch größeren Widerstand in der Einheitszelle gegen die immer gleiche Deformation aller Simulationen eine höhere Spannung in das Material eingetragen wird.

#### 4.1.2 Einfluss der Parameter in Einheitszelle Chi-Bar-Flex

Die folgenden Abbildungen 3.32 (a), 3.32 (b), 3.32 (c) und 3.32 (d) zeigen die Änderungen einer mechanischen Eigenschaft bei Variation der Parameter. In diesen Konturdiagrammen wird auf den Achsen jeweils ein Parameter variiert und die Größe der untersuchten mechanischen Eigenschaften werden farblich dargestellt.

In dieser Parameterstudie wurde ein deutlich kleinerer und ausschließlich positiver Winkelbereich abgedeckt. Im Konturdiagramm in Abbildung 3.32 (a) ist die Parameterstudie der Poissonzahl abgebildet. Es ist ein leichter Anstieg der Poissonzahl sowohl mit Anstieg des Winkels als auch der Dicke abzulesen. Hier sieht man auch die Ähnlichkeit, die diese Einheitszelle zur Einheitszelle Chi-Bar hat. Denn im Bereich  $\beta = 0$  bis  $\beta = \frac{\pi}{8}$ , in dem sich diese Parameterstudie bewegt, ist das Verhalten der Poissonzahl in Abbildung 3.30 (a) sehr ähnlich.

Im Konturdiagramm 3.32 (b) sind die Daten zur benötigten Deformationskraft abgebildet. Hier sieht man bei den höchsten Dickenparametern einen leichten Abfall der Kraft mit steigendem Winkel. Der Einfluss bleibt aber allgemein sehr gering.

In Konturdiagramm 3.32 (d) ist zu sehen, dass die relative Dichte mit Winkel und Dicke linear ansteigt.

Ähnlich wie in der Einheitszelle Chi-Bar ist im Konturdiagramm der maximalen Von-Mises-Spannung 3.32 (c) ein Anstieg der Spannung mit Anstieg der Dicke zu sehen. Die Spannung nimmt jedoch mit zunehmendem Winkel ab. Das deutet wieder auf eine Versteifung der Einheitszelle bei hoher Dicke und kleinem Winkel hin.

#### 4.1.3 Einfluss der Parameter in Einheitszelle Chi

Die folgenden Abbildungen 3.34 (a) und 3.34 (b) zeigen die Änderungen einer mechanischen Eigenschaft bei Variation der Parameter. In diesen Konturdiagrammen wird auf den Achsen jeweils ein Parameter variiert und die Werte der untersuchten mechanischen Eigenschaften sind farblich dargestellt.

Im Vergleich zu den beiden vorangegangenen Einheitszellen sind für diese Einheitszelle die Werte der Poissonzahl in Abbildung 3.34 (a) ausschließlich positiv. Das bedeutet, dass die Dicke der Einheitszelle bei Verlängerung zunimmt.

Mit dieser Einheitszelle kann eine sehr hohe Bandbreite an Poissonzahlen erreicht werden. Die Poissonzahl fällt hierbei mit steigendem Winkel und mit steigender Dicke ab. Mit dieser hohen Variabilität eignet sich diese Einheitszelle gut für Anwendungen in Gitterstrukturen, die eine positive Poissonzahl auf Makroebene benötigen, dabei jedoch punktuell mit der Ausdehnung dieser Struktur variieren sollen.

Im Konturdiagramm der relativen Dichte in Abbildung 3.34 (b) ist zu sehen, dass die relative Dichte mit steigender Dicke und steigendem Winkel zunimmt.

## 4.2 Vergleich der realisierbaren Bandbreite der mechanischen Eigenschaften

Nach der Beschreibung der Einflüsse der Parameter auf die mechanischen Eigenschaften erfolgt in diesem Kapitel eine Visualisierung der Bandbreite, die diese mechanischen Eigenschaften annehmen können. Das ermöglicht einen Vergleich der Bandbreiten der einzelnen mechanischen Eigenschaften der Einheitszellen. Die Werte von Spannung und Deformationskraft wurden für die Einheitszelle Chi nicht dargestellt, da sie keine aussagekräftigen Ergebnisse sind.

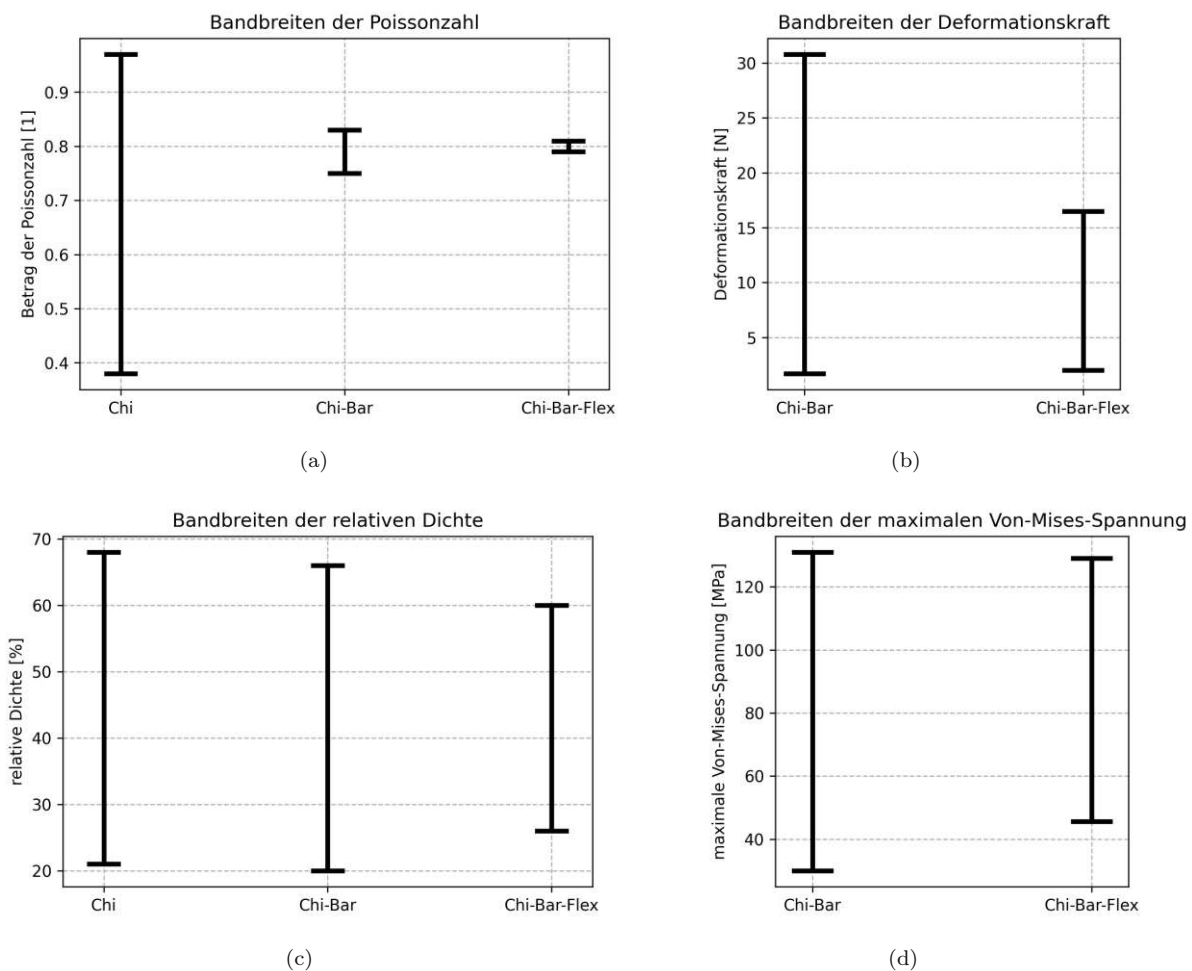


Abbildung 4.1: Bandbreiten der mechanischen Eigenschaften der Einheitszellen

Die Bandbreiten der Poissonzahlen in Abbildung 4.1 (a) weisen die größten Unterschiede auf. Vor allem die Einheitszelle Chi deckt dort eine sehr hohe Bandbreite ab. Die Einheitszelle Chi-Bar-Flex hingegen deckt in allen Bereichen einen deutlich kleineren Bereich ab. Das liegt daran, dass die Bereiche zur Variation der Parameter kleiner ausfallen als bei der artverwandten Einheitszelle Chi-Bar.

Es gibt, wie in Kapitel 3.2.3 angedeutet, auch die Möglichkeit, die Position des Armes und dadurch die Biegung, die die Arme am äußeren Ende aufweisen, als zusätzliche Parameter zu definieren. Dadurch kann das Potenzial dieser Einheitszelle noch deutlich gesteigert werden.

### 4.3 Zusammenfassung und Ausblick

Ziel dieser Arbeit ist es, eine zweidimensionale Gitterstruktur zu erstellen. Die sich wiederholenden Einheitszellen dieser Gitterstruktur können in ihrer Geometrie mithilfe von Parametern variiert werden. Die Geometrie der Einheitszellen wird aus mehreren Splines zusammengesetzt. Es wurden drei solcher Einheitszellen entworfen. Mithilfe einer Finite-Elemente-Simulation können die mechanischen Eigenschaften dieser Einheitszellen bei bestimmten Parametern simuliert werden. All diese Schritte wurden mithilfe von Open-Source-Python-Bibliotheken realisiert.

In Zukunft könnten die Programme zur Erstellung der Geometrie und zur Simulation der mechanischen Eigenschaften öffentlich zugänglich gemacht werden. Dazu muss eine benutzerfreundlichere und intuitive Eingabemöglichkeit der benötigten Daten erarbeitet werden. Mithilfe dieser Software können dann Gitterstrukturen erstellt werden. Ein Aspekt, der in dieser Arbeit nicht erläutert wurde, ist, dass die Geometrie der Gitterstruktur nicht rechteckig sein muss. Splinepy bietet die Möglichkeit, die Form der Gitterstruktur durch komplexe zweidimensionale Splines zu definieren. Dadurch wird der mögliche Einsatzbereich deutlich erweitert. Es soll auch die Funktion, die gewünschten mechanischen Eigenschaften über die Fläche der Gitterstruktur anzugeben, implementiert werden. Durch die Wandelbarkeit der Einheitszellen kann die Gitterstruktur so aufgebaut werden, dass sie die gewünschten lokalen Werte der mechanischen Eigenschaften bestmöglich erfüllt. Durch diese Erweiterung ist die Einsatzmöglichkeit in Kombination mit der additiven Fertigungsmethode sehr vielversprechend.

# Literaturverzeichnis

- [1] R. Abeyaratne. *Continuum Mechanics*, volume II of *Lecture Notes on the Mechanics of Solids*. 2012.
- [2] Altair Engineering Inc. *Altair HyperWorks*. Altair Engineering Inc., Troy, Michigan, USA, 2025.
- [3] F. Andersson. *Bézier and B-spline Technology*. 2003.
- [4] K. J. Bathe. *Finite Element Procedures*. Prentice Hall, 1996.
- [5] J. Bühring. *Effektive Steifigkeiten und das Stabilitätsverhalten von Gitterstrukturen*. Shaker Verlag, 2024.
- [6] L. Chen. Variational formulation of linear elasticity, 2025.
- [7] S. Elgeti. *Skript zur Veranstaltung Einführung in die Finite Elemente Methode*. TU Wien, 2021.
- [8] A. Buffa et al. *Approximation estimates for isogeometric spaces in multipatch geometries*. Numerical Methods for Partial Differential Equations, 2015.
- [9] H. Man et al. *Towards Automatic Stress Analysis using Scaled Boundary Finite Element Method with Quadtree Mesh of High-order Elements*. UNSW Technical Report, 2014.
- [10] I. Thawon et al. *Mechanical anisotropy of 2D lattice structures*. The 12th International Conference on Mechanical Engineering, 2022.
- [11] M. Fleisch et al. *Functional mechanical metamaterial with independently tunable stiffness in the three spatial directions*. Materials Today Advances, 2021.
- [12] M. Fleisch et al. *Chiral-based mechanical metamaterial with tunable normal-strain shear coupling effect*. Engineering Structures, 2023.
- [13] M. Munford et al. Prediction of anisotropic mechanical properties for lattice structures. *Additive Manufacturing*, Vol. 32, 2020.
- [14] O. A. González et al. *Efficient recovery-based error estimation for the smoothed finite element method for smooth and singular linear elasticity*. Computational Mechanics, 2012.
- [15] R. Anderson et al. MFEM: A Modular Finite Element Methods Library. *Computers Mathematics with Applications*, Vol. 81, 2021.
- [16] T. Hirschler et al. *SLM lattice structures: Properties, performance, applications and challenges*. Materials Design, 2019.
- [17] L. Jaewook. *gustaf: Loyal butler for numerical-analysis geometry processing and visualization*. [urlhttps://pypi.org/project/gustaf/](https://pypi.org/project/gustaf/), 2022. MIT License.
- [18] L. Jaewook. *splinepy: Library for prototyping spline geometries of arbitrary dimensions and degrees*. [urlhttps://pypi.org/project/splinepy/](https://pypi.org/project/splinepy/), 2025. MIT License.

- [19] T. Munhowen. *Modeling and Simulation of Superelastic Lattice Structures*. TU Wien, 2025.
- [20] E. Q. Sun. *Shear Locking and Hourglassing in MSC Nastran, ABAQUS, and ANSYS*. [https://moodle.umontpellier.fr/pluginfile.php/480056/mod\\_resource/content/0/Sun-ShearLocking-Hourglassing.pdf](https://moodle.umontpellier.fr/pluginfile.php/480056/mod_resource/content/0/Sun-ShearLocking-Hourglassing.pdf).
- [21] M. Cenanovic und C. J. Jonsson. *Basic FEM*. Jönköping University URL: <https://fea.ju.se/WeakFormElasticity2D/>, 2024.
- [22] D.A. Ham und C.J. Cotter. *Finite Elements: Analysis and Implementation*. Imperial College London, 2024.
- [23] E. Dimas und D. Briassoulis. *3D geometric modelling based on NURBS: a review*. Advances in Engineering Software, 1996.
- [24] L. D. Landau und E. M. Lifshitz. *Theory of Elasticity*, volume 7 of *Course of Theoretical Physics*. Pergamon Press, 3 edition, 1986.
- [25] A. Dimitrov; H. Andrä und E. Schnack. *Efficient computation of order and mode of corner singularities in 3D-elasticity*. International Journal for Numerical Methods in Engineering, 2001.
- [26] Liu G.R und Quek S.S. *The Finite Element Method: A Practical Course*. Butterworth-Heinemann, 2013.
- [27] R. Piltner und R. L. Taylor. *A systematic construction of B-bar functions for linear and non-linear mixed-enhanced finite elements for plane elasticity problems*. International Journal for Numerical Methods in Engineering, 1999.
- [28] A. Collin; G. Sangalli und T. Takacs. *Analysis-suitable  $G^1$  multi-patch parametrizations for  $C^1$  isogeometric spaces*. arXiv preprint, 2016.
- [29] M. Smith; Z. Guan und W. J. Cantwell. *Finite element modelling of the compressive response of lattice structures manufactured using the selective laser melting technique*. International Journal of Mechanical Sciences, February 2013.
- [30] L. Piegl und W. Tiller. *The NURBS Book*. Springer, 2 edition, 1997. Kapitel 2 und 3 sind frei online zugänglich.
- [31] A. Walzl. *Potentiale und Restriktionen von Gitterstrukturen für die additive Fertigung*. Montanuniversität Leoben, 2018.
- [32] T. Wester. *Nature Teaching Structures*. International Journal of Space Structures, 2002.

# Abbildungsverzeichnis

1.1	Gitterstruktur aufgebaut mit Einheitszellen . . . . .	1
2.1	Bézier-Spline dritten Grades [3] . . . . .	4
2.2	Abbildung eines zweidimensionalen Bézier-Splines vom Parameterraum in den physikalischen Raum [3] . . . . .	6
2.3	Einheitszelle als Flächenverbund aus Bézier-Splines . . . . .	7
2.4	Eine aus Einheitszellen aufgebaute Gitterstruktur . . . . .	8
2.5	Eindimensionale Einheitszelle . . . . .	9
2.6	Flächenverbund einer Einheitszelle mit bemaßten Parametern . . . . .	10
2.7	Gitterstruktur mit variablen Parametern . . . . .	10
2.8	Zweidimensionaler Spline mit Vernetzungspunkten aus den Splineparametern und Finite-Elemente-Netz . . . . .	11
2.9	Finite-Elemente-Netz der Einheitszelle . . . . .	12
2.10	Referenzelement für rechteckige lineare Elemente [7] . . . . .	15
2.11	Ansatzfunktionen für vier lineare rechteckige Elemente [7] . . . . .	16
2.12	Beispielelement mit Randbedingungen . . . . .	19
2.13	zweidimensionaler Spannungszustand mit spannungsfreier z-Achse [7] . . . . .	19
2.14	zweidimensionaler Spannungszustand mit verzerrungsfreier z-Achse [7] . . . . .	20
2.15	vollständig periodische Randbedingung [19] . . . . .	23
2.16	vereinfachte periodische Randbedingung [19] . . . . .	24
2.17	Finite-Elemente-Netz mit Zwangsbedingungen am nördlichen Rand . . . . .	25
2.18	Beispielelement mit Randbedingungen und Zwangsbedingungen . . . . .	27
2.19	Vergleich der Verformung der drei Fälle der Randbedingungen . . . . .	28
3.1	Änderung der Geometrie eines Splines mithilfe eines Parameters . . . . .	33
3.2	Einfluss von zwei Parametern auf die mechanischen Eigenschaften einer Einheitszelle . . . . .	34
3.3	Einheitszellen bestehend aus nur einem Kreuz vier Mal wiederholt . . . . .	36
3.4	Abbildung der Ausdehnung der Einheitszelle auf die Gitterstruktur zum Ablesen der Parameter . . . . .	38
3.5	Quadratische Gitterstruktur mit neun Einheitszellen . . . . .	39
3.6	Erstellen von Rundungen an Ecken in der Einheitszelle . . . . .	40
3.7	Positive Richtung der Parameter der Einheitszelle Chi . . . . .	41
3.8	Einheitszelle Chi mit gleichbleibenden Parametern über die Zelle . . . . .	42
3.9	Markierte Punkte an denen in der Einheitszelle die Parameter abgelesen werden . . . . .	43
3.10	Einheitszelle Chi mit verschiedenen Parameterverteilungen . . . . .	44
3.11	Positive Richtung der Parameter der Einheitszelle Chi-Bar . . . . .	45
3.12	Einheitszelle Chi-Bar mit gleichbleibenden Parametern über die Zelle . . . . .	46
3.13	Markierte Punkte an denen in der Einheitszelle die Parameter abgelesen werden . . . . .	46

3.14	Einheitszelle Chi-Bar mit verschiedenen Parameterverteilungen . . . . .	47
3.15	Rundung in der Einheitszelle Chi-Bar . . . . .	48
3.16	Positive Richtung der Parameter der Einheitszelle Chi-Bar-Flex . . . . .	49
3.17	Einheitszelle Chi mit gleichbleibenden Parametern über die Zelle . . . . .	49
3.18	Detailabbildung eines Armes des Chis das in der Chi-Bar-Flex-Einheitszelle verwendet wird . . .	50
3.19	Punkte, an denen in der Einheitszelle Chi-Bar-Flex die Parameter abgelesen werden . . . . .	50
3.20	Einheitszelle Chi-Bar-Flex mit verschiedenen Parameterverteilungen . . . . .	51
3.21	Finite-Elemente-Netz der Beispielsimulation . . . . .	53
3.22	Simulationsergebnisse der Einheitszelle Chi-Bar-Flex nach Verformung . . . . .	54
3.23	Verteilung der Von-Mises-Spannungen in der undeformierten Einheitszelle . . . . .	54
3.24	Vergleich der Ergebnisse der Deformationen in MFEM und Abaqus . . . . .	56
3.25	Vergleich der Simulationsergebnisse der Von-Mises-Spannungen in MFEM und Abaqus . . . . .	58
3.26	Graph der Daten der Netzfeinheitsstudie Chi-Bar . . . . .	60
3.27	Graph der Daten der Netzfeinheitsstudie Chi-Bar-Flex . . . . .	61
3.28	Graph der Daten der Netzfeinheitsstudie Chi . . . . .	62
3.29	Grenzparameter der Parameterstudie für die Einheitszelle Chi-Bar . . . . .	64
3.30	Konturdiagramme mit allen mechanischen Eigenschaften der Parameterstudie von Chi-Bar . . . . .	65
3.31	Grenzparameter der Parameterstudie für die Einheitszelle Chi-Bar-Flex . . . . .	66
3.32	Konturdiagramme mit allen mechanischen Eigenschaften der Parameterstudie von Chi-Bar-Flex .	67
3.33	Grenzparameter der Parameterstudie für die Einheitszelle Chi . . . . .	68
3.34	Konturdiagramme mit allen mechanischen Eigenschaften der Parameterstudie von Chi . . . . .	69
4.1	Bandbreiten der mechanischen Eigenschaften der Einheitszellen . . . . .	72
2	Nummerierung Kontrollpunkte und bemaßte Variablen für die Berechnung von Chi . . . . .	82
3	Nummerierung Kontrollpunkte und bemaßte Variablen für die Berechnung von Chi-Bar . . . . .	84
4	Nummerierung Kontrollpunkte und bemaßte Variablen für die Berechnung von Chi-Bar-Flex . . .	86

# Tabellenverzeichnis

3.1	Ergebnisse Netzfeinheitsstudie Chi-Bar . . . . .	60
3.2	Ergebnisse Netzfeinheitsstudie Chi-Bar-Flex . . . . .	61
3.3	Ergebnisse Netzfeinheitsstudie Chi . . . . .	62

# Anhang

## MFEM-kompatible Definition des Finite-Elemente-Netzes

MFEM mesh v1.0

dimension

2

elements

1

1 3 0 1 3 2

boundary

4

1 1 0 1

2 1 1 3

3 1 3 2

4 1 2 0

vertices

4

2

0.0 0.0

10.0 0.0

0.0 10.0

10.0 10.0

## Python-Programm zur MFEM-Implementierung des Beispielproblems

```
import numpy as np
import mfem.ser as mfem
import pypardiso
import scipy.sparse as sparse

element_order = 1
Fx = 100

mesh = mfem.Mesh("test_mesh_multipatch.mesh")

fecollection = mfem.H1_FECollection(element_order, mesh.Dimension())
fespace = mfem.FiniteElementSpace(mesh, fecollection, mesh.Dimension())

max_n_atr = mesh.bdr_attributes.Max()
essential_bdr_S = mfem.intArray([0]*max_n_atr)
essential_bdr_S[0] = 1
essential_bdr_E = mfem.intArray([0]*max_n_atr)
essential_bdr_E[1] = 1
essential_bdr_N = mfem.intArray([0]*max_n_atr)
essential_bdr_N[2] = 1
essential_bdr_W = mfem.intArray([0]*max_n_atr)
essential_bdr_W[3] = 1

essential_tdof_S = mfem.intArray()
fespace.GetEssentialTrueDofs(essential_bdr_S, essential_tdof_S, component=1)
essential_tdof_E = mfem.intArray()
fespace.GetEssentialTrueDofs(essential_bdr_E, essential_tdof_E, component=0)
essential_tdof_N = mfem.intArray()
fespace.GetEssentialTrueDofs(essential_bdr_N, essential_tdof_N, component=1)
essential_tdof_W = mfem.intArray()
fespace.GetEssentialTrueDofs(essential_bdr_W, essential_tdof_W, component=0)

essential_tdof_list = mfem.intArray()
essential_tdof_list.Append(essential_tdof_S[0])
essential_tdof_list.Append(essential_tdof_W)

u = mfem.GridFunction(fespace)
u.Assign(0.0)

b = mfem.LinearForm(fespace)
b.Assemble()

for dof in essential_tdof_E:
    b[dof] += Fx
```

```

Lambda = 121154
Mu = 80769
Lambda_ccf = mfem.ConstantCoefficient(Lambda)
Mu_ccf =mfem. ConstantCoefficient(Mu)

a = mfem.BilinearForm(fespace)
a.AddDomainIntegrator(mfem.ElasticityIntegrator(Lambda_ccf, Mu_ccf))
a.Assemble()

K = mfem.OperatorPtr()
U = mfem.Vector()
F = mfem.Vector()
a.FormLinearSystem(essential_tdof_list, u, b, K, U, F)

KK = mfem.OperatorHandle2SparseMatrix(K)

w, h = KK.Width(), KK.Height()
I = KK.GetIArray()
J = KK.GetJArray()
data = KK.GetDataArray()

KK_sparse = sparse.csr_matrix((data, J, I), shape=(h, w))

print(KK_sparse.toarray()) # prints full dense matrix

F_array = F.GetDataArray() # MFEM → NumPy view
print(F_array)

KK_dens = KK.ToDenseMatrix()

KK_sparse = sparse.csr_matrix(
    KK_dens.GetDataArray()
)
F_sparse = F.GetDataArray()

u_sol = pypardiso.spsolve(KK_sparse, F_sparse)

```

## Gleichungen der Einheitszellen

In diesem Anhang werden die Gleichungen der Positionen der Kontrollpunkte, welche die Gitterstrukturen der Einheitszellen bilden, definiert. Diese Einheitszellen werden in Kapitel 3.2 genauer beschrieben.

Aufgebaut wird dieser Anhang so, dass die beiden Abbildungen die nummerierten Kontrollpunkte zeigen. Dadurch können die beiden Gleichungen in x- und y-Richtung dem entsprechenden Kontrollpunkt zugeordnet werden. In der zweiten Abbildung werden immer die Parameter dargestellt, die für die Erstellung der Gitterstruktur eingelesen werden. Zusätzlich werden auch alle anderen variabel gehaltenen Größen, die zur Berechnung der Gleichungen benötigt werden, eingezeichnet, welche aber über die gesamte Einheitszelle (und dadurch auch über die gesamte Makrostruktur) gleich bleiben.

### Chi-Einheitszelle

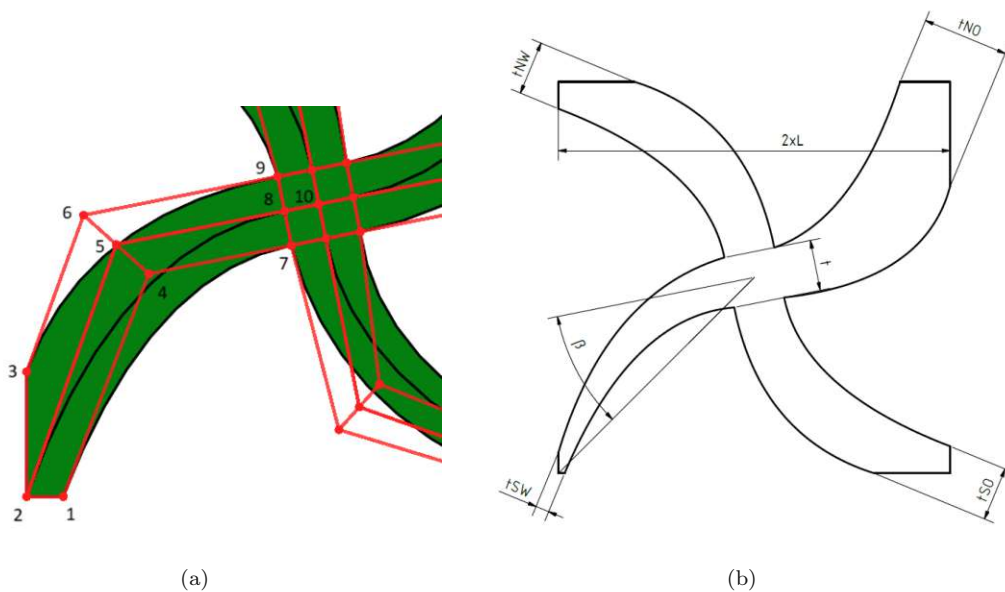


Abbildung 2: Nummerierung Kontrollpunkte und bemaßte Variablen für die Berechnung von Chi

$$r = \frac{\sqrt{0.125} \cdot L}{0.5}$$

$$a = r \cdot \cos\left(\beta + \frac{\pi}{4}\right)$$

$$b = r \cdot \sin\left(\beta + \frac{\pi}{4}\right)$$

$$\gamma = \arctan\left(\frac{L - (r \cdot \cos(\beta + \frac{\pi}{4}))}{L - (r \cdot \sin(\beta + \frac{\pi}{4}))}\right) - \frac{\pi}{4}$$

$$\gamma_{R1} = \arctan\left(\frac{L - (r \cdot \cos(\beta_{SW} + \frac{\pi}{4}))}{L - (r \cdot \sin(\beta_{SW} + \frac{\pi}{4}))}\right) - \frac{\pi}{4}$$

$$\delta = \pi - \gamma_{R1} - \beta_{SW}$$

$$t_{SW\_ratio\_middle\_CP} = \frac{(t_{SW} - t) \cdot \text{Length\_ratio} + t}{t}$$

$$N_{1x} = -L + \frac{t_{SW}}{2 \cdot \sin(\gamma_{R1} + \frac{\pi}{4})}$$

$$N_{1y} = -L$$

$$N_{2x} = -L$$

$$N_{2y} = -L$$

$$N_{3x} = -L$$

$$N_{3y} = -L + \frac{t_{SW}}{2 \cdot \sin(-\gamma_{R1} + \frac{\pi}{4})}$$

$$N_{4x} = -b + \frac{t_{SW\_ratio\_middle\_CP} \cdot t}{2 \cdot \sin(\frac{\delta}{2})} \cdot \sin\left(\frac{3\pi}{4} - \frac{\delta}{2} - \beta\right)$$

$$N_{4y} = -a - \frac{t_{SW\_ratio\_middle\_CP} \cdot t}{2 \cdot \sin(\frac{\delta}{2})} \cdot \cos\left(\frac{3\pi}{4} - \frac{\delta}{2} - \beta\right)$$

$$N_{5x} = -b$$

$$N_{5y} = -a$$

$$N_{6x} = -b - \frac{t_{SW\_ratio\_middle\_CP} \cdot t}{2 \cdot \sin(\frac{\delta}{2})} \cdot \sin\left(\frac{3\pi}{4} - \frac{\delta}{2} - \beta\right)$$

$$N_{6y} = -a + \frac{t_{SW\_ratio\_middle\_CP} \cdot t}{2 \cdot \sin(\frac{\delta}{2})} \cdot \cos\left(\frac{3\pi}{4} - \frac{\delta}{2} - \beta\right)$$

$$N_{7x} = -\frac{t \cdot \sin(\beta)}{\sqrt{2}}$$

$$N_{7y} = -\frac{t \cdot \cos(\beta)}{\sqrt{2}}$$

$$N_{8x} = -\frac{t \cdot \sin(\beta + \frac{\pi}{4})}{2}$$

$$N_{8y} = -\frac{t \cdot \cos(\beta + \frac{\pi}{4})}{2}$$

$$N_{9x} = -\frac{t \cdot \sin(\pi - \beta - \frac{\pi}{2})}{\sqrt{2}}$$

$$N_{9y} = + \frac{t \cdot \cos(\pi - \beta - \frac{\pi}{2})}{\sqrt{2}}$$

$$N_{10x} = 0$$

$$N_{10y} = 0$$

### Chi-Bar-Einheitszelle

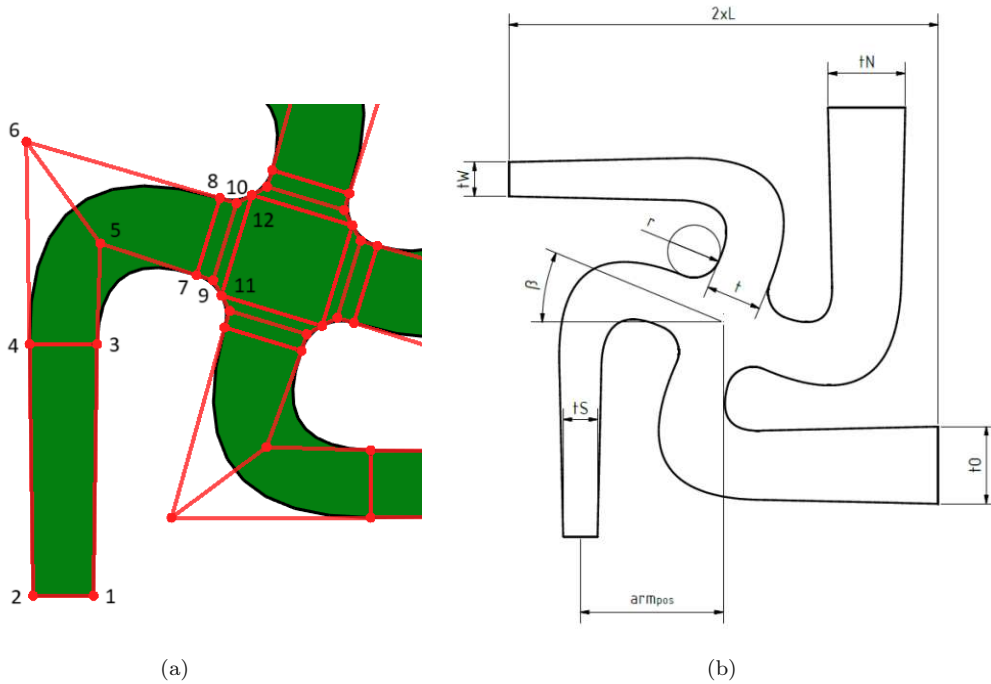


Abbildung 3: Nummerierung Kontrollpunkte und bemaßte Variablen für die Berechnung von Chi-Bar

$$l_{ges1} = \frac{arm_{pos}}{\cos(\beta)} - \frac{t}{2} + L + arm_{pos} \cdot \tan(\beta)$$

$$\delta t_{S,knick} = \frac{(t_S - t) \left( \frac{arm_{pos}}{\cos(\beta)} - \frac{t}{2} \right) / l_{ges1} + t}{t}$$

$$\delta t_{S,gerade} = \frac{(t_S - t) \left( \frac{arm_{pos}}{\cos(\beta)} - \frac{t}{2} \right) \cdot 2 / l_{ges1} + t}{t}$$

$$N_{1x} = -arm_{pos} + \frac{t_S}{2}$$

$$N_{1y} = -L$$

$$N_{2x} = -arm_{pos} - \frac{t_S}{2}$$

$$N_{2y} = -L$$

$$N_{3x} = -arm_{pos} + \delta t_{S,gerade} \cdot \frac{t}{2}$$

$$N_{3y} = \left( arm_{pos} \cdot \tan(\beta) - \frac{arm_{pos}}{\cos(\beta)} + \frac{t}{2} \right) + r$$

$$N_{4x} = -arm_{pos} - \delta t_{S,gerade} \cdot \frac{t}{2}$$

$$N_{4y} = \left( arm_{pos} \cdot \tan(\beta) - \frac{arm_{pos}}{\cos(\beta)} + \frac{t}{2} \right) + r$$

$$N_{5x} = -arm_{pos} + \delta t_{S,knick} \cdot \frac{t}{2}$$

$$N_{5y} = \left( arm_{pos} - \frac{t}{2} \right) \cdot \tan(\beta) - \delta t_{S,knick} \cdot \frac{t}{2 \cdot \sin\left(\frac{\pi}{2} - \beta\right)}$$

$$N_{6x} = -arm_{pos} - \delta t_{S,knick} \cdot \frac{t}{2}$$

$$N_{6y} = \left( arm_{pos} + \frac{t}{2} \right) \cdot \tan(\beta) + \delta t_{S,knick} \cdot \frac{t}{2 \cdot \sin\left(\frac{\pi}{2} - \beta\right)}$$

$$N_{7x} = -\frac{t}{\sqrt{2}} \cdot \cos\left(-\frac{\pi}{4} + \beta\right) - r \cdot \cos(\beta)$$

$$N_{7y} = \frac{t}{\sqrt{2}} \cdot \sin\left(-\frac{\pi}{4} + \beta\right) + r \cdot \sin(\beta)$$

$$N_{8x} = -\frac{t}{\sqrt{2}} \cdot \cos\left(\frac{\pi}{4} + \beta\right) - r \cdot \cos(\beta)$$

$$N_{8y} = \frac{t}{\sqrt{2}} \cdot \sin\left(\frac{\pi}{4} + \beta\right) + r \cdot \sin(\beta)$$

$$N_{9x} = -\frac{t}{\sqrt{2}} \cdot \cos\left(-\frac{\pi}{4} + \beta\right) - r \cdot (2 - \sqrt{2}) \cdot \cos(\beta)$$

$$N_{9y} = \frac{t}{\sqrt{2}} \cdot \sin\left(-\frac{\pi}{4} + \beta\right) + r \cdot (2 - \sqrt{2}) \cdot \sin(\beta)$$

$$N_{10x} = -\frac{t}{\sqrt{2}} \cdot \cos\left(\frac{\pi}{4} + \beta\right) - r \cdot (2 - \sqrt{2}) \cdot \cos(\beta)$$

$$N_{10y} = \frac{t}{\sqrt{2}} \cdot \sin\left(\frac{\pi}{4} + \beta\right) + r \cdot (2 - \sqrt{2}) \cdot \sin(\beta)$$

$$N_{1S} = -\left( \frac{t}{\sqrt{2}} + r \cdot (\sqrt{2} - 1) \right) \cdot \cos\left(-\frac{\pi}{4} + \beta\right)$$

$$N_{11y} = \left( \frac{t}{\sqrt{2}} + r \cdot (\sqrt{2} - 1) \right) \cdot \sin \left( -\frac{\pi}{4} + \beta \right)$$

$$N_{12x} = - \left( \frac{t}{\sqrt{2}} + r \cdot (\sqrt{2} - 1) \right) \cdot \cos \left( \frac{\pi}{4} + \beta \right)$$

$$N_{12y} = \left( \frac{t}{\sqrt{2}} + r \cdot (\sqrt{2} - 1) \right) \cdot \sin \left( \frac{\pi}{4} + \beta \right)$$

### Chi-Bar-Flex-Einheitszelle

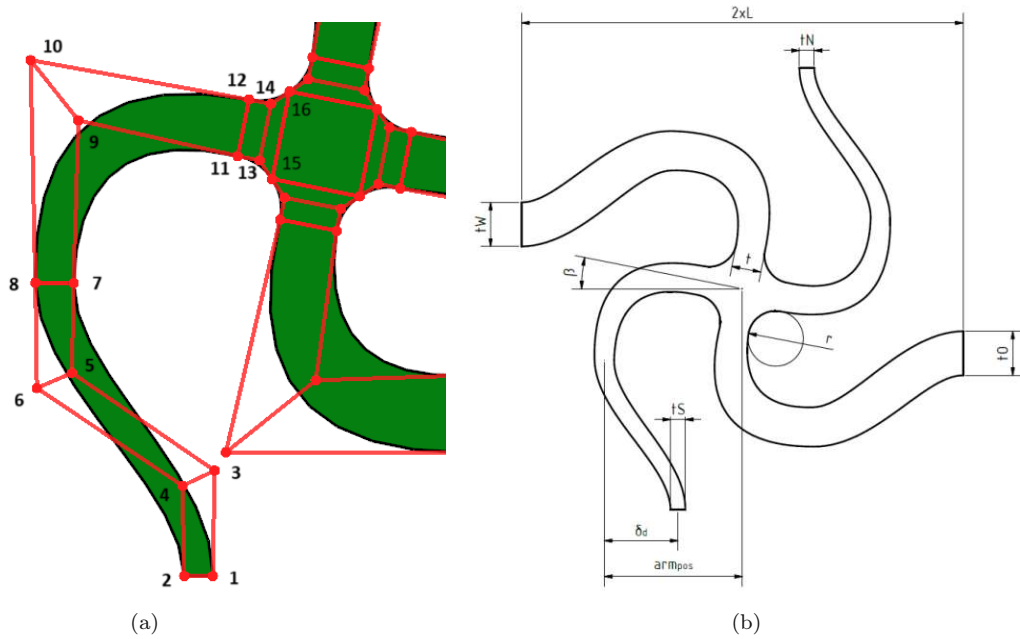


Abbildung 4: Nummerierung Kontrollpunkte und bemaßte Variablen für die Berechnung von Chi-Bar-Flex

$$l_{\text{ges1}} = \frac{arm_{\text{pos}}}{\cos(\beta)} - \frac{t}{2} + L + arm_{\text{pos}} \times \tan(\beta)$$

$$\delta t_{S,\text{knick}} = \frac{(t_S - t) \left( \frac{arm_{\text{pos}}}{\cos(\beta)} - \frac{t}{2} \right) \div l_{\text{ges1}} + t}{t}$$

$$\delta t_{S,\text{gerade}} = \frac{(t_S - t) \left( \left( \frac{arm_{\text{pos}}}{\cos(\beta)} - \frac{t}{2} \right) \times 2 \div l_{\text{ges1}} \right) + t}{t}$$

$$h = arm_{\text{pos}} \tan(\beta) - \frac{arm_{\text{pos}}}{\cos(\beta)} + \frac{t}{2} + r$$

$$k = \frac{3 \delta_d}{L + h}$$

$$N_{1x} = -arm_{\text{pos}} + \frac{t_S}{2} + \delta_d$$

$$N_{1y} = -L$$

$$N_{2x} = -arm_{pos} - \frac{tS}{2} + \delta_d$$

$$N_{2y} = -L$$

$$N_{3x} = -arm_{pos} + \frac{\delta t_{S,gerade} \times t}{6} + \frac{2 \times tS}{6} + \delta_d$$

$$N_{3y} = -\frac{2L}{3} + \frac{1}{3}h + \frac{0.5 tS \left( -1 + \frac{1}{\cos(\arctan(k))} \right)}{k}$$

$$N_{4x} = -arm_{pos} - \frac{\delta t_{S,gerade} \times t}{6} - \frac{2 \times tS}{6} + \delta_d$$

$$N_{4y} = -\frac{2L}{3} + \frac{1}{3}h - \frac{0.5 tS \left( -1 + \frac{1}{\cos(\arctan(k))} \right)}{k}$$

$$N_{5x} = -arm_{pos} + \frac{2 \times \delta t_{S,gerade} \times t}{6} + \frac{tS}{6}$$

$$N_{5y} = -\frac{L}{3} + \frac{2}{3}h + \frac{0.5 tS \left( -1 + \frac{1}{\cos(\arctan(k))} \right)}{k}$$

$$N_{6x} = -arm_{pos} - \frac{2 \times \delta t_{S,gerade} \times t}{6} - \frac{tS}{6}$$

$$N_{6y} = -\frac{L}{3} + \frac{2}{3}h - \frac{0.5 tS \left( -1 + \frac{1}{\cos(\arctan(k))} \right)}{k}$$

$$N_{7x} = -arm_{pos} + \frac{\delta t_{S,gerade} \times t}{2}$$

$$N_{7y} = \left( arm_{pos} \tan(\beta) - \frac{arm_{pos}}{\cos(\beta)} + \frac{t}{2} \right) + r$$

$$N_{8x} = -arm_{pos} - \frac{\delta t_{S,gerade} \times t}{2}$$

$$N_{8y} = \left( arm_{pos} \tan(\beta) - \frac{arm_{pos}}{\cos(\beta)} + \frac{t}{2} \right) + r$$

$$N_{9x} = -arm_{pos} + \frac{\delta t_{S,knick} \times t}{2}$$

$$N_{9y} = \left( arm_{pos} - \frac{t}{2} \right) \tan(\beta) - \frac{\delta t_{S,knick} \times t}{2 \sin\left(\frac{\pi}{2} - \beta\right)}$$

$$N_{10x} = -arm_{pos} - \frac{\delta t_{S,knick} \times t}{2}$$

$$N_{10y} = \left( arm_{pos} + \frac{t}{2} \right) \tan(\beta) + \frac{\delta t_{S,knick} \times t}{2 \sin\left(\frac{\pi}{2} - \beta\right)}$$

$$N_{11x} = -\frac{t}{\sqrt{2}} \cos\left(-\frac{\pi}{4} + \beta\right) - r \cos(\beta)$$

$$N_{11y} = \frac{t}{\sqrt{2}} \sin\left(-\frac{\pi}{4} + \beta\right) + r \sin(\beta)$$

$$N_{12x} = -\frac{t}{\sqrt{2}} \cos\left(\frac{\pi}{4} + \beta\right) - r \cos(\beta)$$

$$N_{12y} = \frac{t}{\sqrt{2}} \sin\left(\frac{\pi}{4} + \beta\right) + r \sin(\beta)$$

$$N_{13x} = -\frac{t}{\sqrt{2}} \cos\left(-\frac{\pi}{4} + \beta\right) - r(2 - \sqrt{2}) \cos(\beta)$$

$$N_{13y} = \frac{t}{\sqrt{2}} \sin\left(-\frac{\pi}{4} + \beta\right) + r(2 - \sqrt{2}) \sin(\beta)$$

$$N_{14x} = -\frac{t}{\sqrt{2}} \cos\left(\frac{\pi}{4} + \beta\right) - r(2 - \sqrt{2}) \cos(\beta)$$

$$N_{14y} = \frac{t}{\sqrt{2}} \sin\left(\frac{\pi}{4} + \beta\right) + r(2 - \sqrt{2}) \sin(\beta)$$

$$N_{15x} = -\left(\frac{t}{\sqrt{2}} + r(\sqrt{2} - 1)\right) \cos\left(-\frac{\pi}{4} + \beta\right)$$

$$N_{15y} = \left(\frac{t}{\sqrt{2}} + r(\sqrt{2} - 1)\right) \sin\left(-\frac{\pi}{4} + \beta\right)$$

$$N_{16x} = -\left(\frac{t}{\sqrt{2}} + r(\sqrt{2} - 1)\right) \cos\left(\frac{\pi}{4} + \beta\right)$$

$$N_{16y} = \left(\frac{t}{\sqrt{2}} + r(\sqrt{2} - 1)\right) \sin\left(\frac{\pi}{4} + \beta\right)$$

## Ergebnisse der Simulationen der Parameterstudien

In diesem Anhang werden alle Ergebnisse der mechanischen Eigenschaften, die durch die Simulationen im Zuge der Parameterstudien der einzelnen Einheitszellen berechnet worden sind, in Tabellen aufgelistet.

Um Platz zu sparen, wurden die Variablen nur abgekürzt in die Tabellen eingetragen.

Die Parameter, die für die Erstellung der Gitterstrukturen verwendet wurden, sind in der obersten Reihe und in der Spalte ganz links eingetragen. In der linken Spalte sind die Winkel eingetragen, die für den Parameter des Verdrehungswinkels simuliert wurden. Einheit ist Radianen.

In der obersten Reihe ist die Dicke der Arme der Chis aufgelistet, mit der die Simulation durchgeführt wurde. Diese Dicke ist in mm eingetragen, wobei die Seitenlänge der Einheitszelle immer 10mm beträgt.

Pro Simulation werden vier Werte abgespeichert, welche die mechanischen Eigenschaften der Einheitszelle zeigen. Diese sind abgekürzt in die zweite Spalte von links eingetragen. Was die Abkürzungen bedeuten ist Folgendes:

P ... Poissonzahl der Einheitszelle [1]

F ... Kraft, die für die Deformation benötigt wird [N]

S ... Maximal vorkommende gemittelte Von-Mises-Spannung [MPa]

D ... Dichte oder Anteil der Einheitszelle der von der Gitterstruktur eingenommen wird [%]

## Einheitszelle Chi

	t	0.4	0.45	0.5	0.55	0.6	0.65	0.7
0/8 * Pi/2	P	0.974	0.966	0.958	0.95	0.94	0.93	0.918
	F	16.6	21.6	27.7	35.0	43.7	53.9	65.7
	S	94.4	110.0	127.0	146.0	164.0	184.0	205.0
	D	21.3	23.8	26.3	28.7	31.1	33.4	35.7
1/8 * Pi/2	P	0.959	0.952	0.943	0.934	0.924	0.913	0.901
	F	16.2	21.3	27.7	35.4	44.5	55.3	68.0
	S	130.0	152.0	175.0	198.0	223.0	248.0	274.0
	D	21.7	24.3	26.7	29.2	31.6	34.0	36.3
2/8 * Pi/2	P	0.922	0.913	0.903	0.892	0.88	0.867	0.853
	F	15.0	20.6	27.6	36.2	46.7	59.3	74.3
	S	160.0	186.0	214.0	242.0	271.0	301.0	332.0
	D	22.8	25.4	28.0	30.6	33.1	35.5	37.9
3/8 * Pi/2	P	0.867	0.856	0.844	0.831	0.816	0.8	0.783
	F	13.7	19.8	27.7	37.8	50.2	65.5	84.0
	S	182.0	212.0	244.0	276.0	310.0	346.0	382.0
	D	24.2	27.0	29.8	32.5	35.1	37.7	40.3
4/8 * Pi/2	P	0.805	0.791	0.775	0.758	0.74	0.72	0.698
	F	12.6	19.0	27.6	38.6	52.6	70.0	91.4
	S	187.0	220.0	255.0	290.0	328.0	366.0	406.0
	D	25.7	28.7	31.6	34.5	37.3	40.1	42.8
5/8 * Pi/2	P	0.745	0.729	0.711	0.691	0.67	0.647	0.623
	F	11.0	17.0	25.0	35.4	48.7	65.3	85.8
	S	173.0	204.0	237.0	271.0	306.0	343.0	382.0
	D	27.0	30.2	33.2	36.3	39.3	42.2	45.1
6/8 * Pi/2	P	0.695	0.679	0.661	0.641	0.62	0.597	0.573
	F	9.3	14.3	20.9	29.4	40.0	53.3	69.4
	S	146.0	172.0	200.0	228.0	257.0	287.0	319.0
	D	27.9	31.2	34.4	37.6	40.7	43.8	46.8
7/8 * Pi/2	P	0.653	0.637	0.621	0.603	0.584	0.564	0.543
	F	8.14	12.2	17.5	24.1	32.3	42.2	54.0
	S	117.0	138.0	160.0	182.0	204.0	227.0	251.0
	D	28.5	31.8	35.2	38.5	41.7	44.9	48.1
8/8 * Pi/2	P	0.616	0.602	0.587	0.571	0.554	0.536	0.517
	F	7.84	11.4	15.8	21.2	27.7	35.4	44.4
	S	96.0	112.0	128.0	150.0	173.0	199.0	229.0
	D	28.8	32.3	35.7	39.1	42.4	45.7	49.0

	t	0.75	0.8	0.85	0.9	0.95	1.0
0/8 * Pi/2	P	0.906	0.893	0.879	0.865	0.85	0.834
	F	79.4	95.0	113.0	133.0	155.0	180.0
	S	226.0	249.0	272.0	296.0	320.0	346.0
	D	37.9	40.1	42.3	44.4	46.5	48.6
1/8 * Pi/2	P	0.888	0.874	0.859	0.844	0.828	0.811
	F	82.7	99.6	119.0	141.0	165.0	192.0
	S	301.0	329.0	358.0	387.0	417.0	448.0
	D	38.6	40.8	43.0	45.1	47.3	49.3
2/8 * Pi/2	P	0.838	0.822	0.804	0.786	0.767	0.747
	F	91.8	112.0	136.0	163.0	193.0	227.0
	S	364.0	397.0	431.0	466.0	500.0	535.0
	D	40.3	42.6	44.9	47.1	49.3	51.5
3/8 * Pi/2	P	0.764	0.744	0.723	0.7	0.677	0.652
	F	106.0	132.0	162.0	198.0	238.0	283.0
	S	419.0	457.0	496.0	534.0	572.0	609.0
	D	42.8	45.2	47.6	50.0	52.3	54.5
4/8 * Pi/2	P	0.675	0.651	0.625	0.598	0.57	0.541
	F	117.0	148.0	185.0	227.0	276.0	332.0
	S	447.0	488.0	529.0	568.0	607.0	662.0
	D	45.5	48.1	50.6	53.1	55.6	58.0
5/8 * Pi/2	P	0.597	0.57	0.542	0.512	0.482	0.451
	F	111.0	141.0	176.0	217.0	265.0	319.0
	S	421.0	460.0	500.0	538.0	575.0	614.0
	D	47.9	50.7	53.4	56.1	58.7	61.2
6/8 * Pi/2	P	0.548	0.522	0.494	0.466	0.436	0.406
	F	88.9	112.0	140.0	171.0	208.0	250.0
	S	351.0	384.0	418.0	452.0	499.0	564.0
	D	49.8	52.7	55.6	58.5	61.2	64.0
7/8 * Pi/2	P	0.52	0.497	0.472	0.447	0.42	0.393
	F	68.0	84.4	103.0	125.0	150.0	178.0
	S	279.0	319.0	363.0	413.0	474.0	544.0
	D	51.2	54.3	57.3	60.3	63.2	66.2
8/8 * Pi/2	P	0.498	0.477	0.455	0.432	0.408	0.383
	F	54.8	66.7	80.2	95.3	112.0	131.0
	S	265.0	305.0	348.0	396.0	461.0	532.0
	D	52.2	55.4	58.6	61.7	64.8	67.8

## Einheitszelle Chi-Bar

	t	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9
$-5/5 * \text{Pi}/6$	P	-0.768	-0.767	-0.767	-0.766	-0.765	-0.763	-0.762	-0.76	-0.758	-0.756	-0.754
	F	2.96	4.17	5.67	7.49	9.64	12.1	15.0	18.3	22.0	26.2	30.8
	S	57.0	64.2	71.5	78.8	86.1	93.5	101.0	109.0	116.0	124.0	131.0
	D	20.3	22.8	25.2	27.6	30.0	32.4	34.8	37.1	39.5	41.8	44.1
$-4/5 * \text{Pi}/6$	P	-0.783	-0.783	-0.782	-0.782	-0.781	-0.78	-0.779	-0.777	-0.776	-0.774	-0.772
	F	2.69	3.8	5.17	6.83	8.8	11.1	13.8	16.8	20.2	24.0	28.3
	S	51.0	57.6	64.2	70.9	77.7	84.6	91.6	98.6	106.0	113.0	121.0
	D	20.9	23.5	26.0	28.5	31.0	33.4	35.9	38.3	40.8	43.2	45.6
$-3/5 * \text{Pi}/6$	P	-0.796	-0.796	-0.796	-0.795	-0.795	-0.794	-0.793	-0.792	-0.791	-0.789	-0.787
	F	2.5	3.53	4.8	6.35	8.18	10.3	12.8	15.6	18.8	22.4	26.3
	S	46.7	52.8	58.9	65.2	71.6	78.1	84.8	91.5	98.4	106.0	113.0
	D	21.6	24.2	26.8	29.4	32.0	34.5	37.1	39.6	42.2	44.7	47.2
$-2/5 * \text{Pi}/6$	P	-0.807	-0.807	-0.807	-0.806	-0.806	-0.805	-0.804	-0.803	-0.802	-0.801	-0.799
	F	2.34	3.31	4.51	5.96	7.69	9.7	12.0	14.7	17.7	21.0	24.8
	S	43.3	49.1	54.9	60.9	67.0	73.2	79.6	86.1	93.0	100.0	108.0
	D	22.3	25.0	27.7	30.4	33.1	35.7	38.4	41.0	43.7	46.3	48.9
$-1/5 * \text{Pi}/6$	P	-0.815	-0.815	-0.815	-0.815	-0.814	-0.814	-0.813	-0.812	-0.811	-0.81	-0.808
	F	2.22	3.14	4.27	5.65	7.28	9.19	11.4	13.9	16.7	19.9	23.5
	S	40.7	46.1	51.7	57.4	63.2	69.2	75.4	81.9	88.7	95.8	103.0
	D	23.1	25.9	28.7	31.4	34.2	37.0	39.8	42.5	45.3	48.0	50.7
$0/5 * \text{Pi}/6$	P	-0.821	-0.821	-0.821	-0.821	-0.82	-0.82	-0.819	-0.818	-0.817	-0.816	-0.815
	F	2.11	2.99	4.07	5.38	6.93	8.75	10.8	13.2	15.9	19.0	22.4
	S	38.5	43.6	48.9	54.4	60.0	65.8	71.9	78.3	85.1	92.2	100.0
	D	23.9	26.8	29.7	32.6	35.5	38.4	41.3	44.1	47.0	49.9	52.7
$1/5 * \text{Pi}/6$	P	-0.825	-0.825	-0.825	-0.824	-0.824	-0.823	-0.823	-0.822	-0.821	-0.82	-0.819
	F	2.02	2.86	3.89	5.14	6.63	8.36	10.4	12.7	15.2	18.1	21.4
	S	36.6	41.5	46.6	51.8	57.2	62.8	68.8	75.1	81.8	89.1	97.0
	D	24.8	27.8	30.8	33.8	36.9	39.9	42.9	45.9	48.9	51.9	54.9
$2/5 * \text{Pi}/6$	P	-0.827	-0.827	-0.826	-0.826	-0.825	-0.825	-0.824	-0.823	-0.822	-0.821	-0.82
	F	1.94	2.74	3.73	4.93	6.35	8.01	9.93	12.1	14.6	17.4	20.5
	S	34.8	39.5	44.4	49.5	54.6	60.1	66.0	72.2	78.9	86.2	94.2
	D	25.8	28.9	32.1	35.2	38.4	41.5	44.7	47.8	51.0	54.1	57.2
$3/5 * \text{Pi}/6$	P	-0.827	-0.826	-0.826	-0.825	-0.825	-0.824	-0.823	-0.823	-0.822	-0.821	-0.82
	F	1.86	2.63	3.58	4.73	6.1	7.69	9.54	11.6	14.0	16.7	19.7
	S	33.3	37.8	42.4	47.2	52.2	57.5	63.2	69.2	75.9	83.2	91.3
	D	26.9	30.2	33.5	36.8	40.1	43.4	46.7	50.0	53.3	56.6	59.8
$4/5 * \text{Pi}/6$	P	-0.824	-0.824	-0.823	-0.823	-0.822	-0.821	-0.821	-0.82	-0.819	-0.818	-0.817
	F	1.79	2.53	3.44	4.55	5.86	7.39	9.16	11.2	13.5	16.0	18.9
	S	31.8	36.1	40.5	45.1	49.9	55.0	60.4	66.3	72.9	80.0	88.0
	D	28.1	31.6	35.1	38.6	42.0	45.5	49.0	52.4	55.9	59.3	62.8
$5/5 * \text{Pi}/6$	P	-0.82	-0.819	-0.819	-0.818	-0.817	-0.817	-0.816	-0.815	-0.814	-0.813	-0.812
	F	1.72	2.43	3.31	4.37	5.63	7.11	8.81	10.7	12.9	15.4	18.2
	S	30.4	34.5	38.7	43.0	47.5	52.4	57.6	63.3	69.6	76.6	84.4
	D	29.6	33.3	36.9	40.6	44.2	47.9	51.6	55.2	58.9	62.5	66.2

### Einheitszelle Chi-Bar-Flex

	t	0.4	0.45	0.5	0.55	0.6	0.65	0.7	0.75	0.8
0/10 * Pi/8	P	-0.81	-0.809	-0.808	-0.807	-0.806	-0.805	-0.804	-0.802	-0.801
	F	2.12	3.02	4.13	5.48	7.09	8.98	11.2	13.7	16.5
	S	55.6	64.1	72.9	81.8	91.0	100.0	110.0	120.0	129.0
	D	26.4	29.6	32.8	35.9	39.1	42.3	45.4	48.6	51.7
1/10 * Pi/8	P	-0.81	-0.809	-0.808	-0.807	-0.806	-0.805	-0.804	-0.802	-0.8
	F	2.12	3.01	4.12	5.46	7.06	8.95	11.1	13.6	16.5
	S	54.1	62.3	70.8	79.4	88.3	97.3	107.0	116.0	126.0
	D	26.7	29.9	33.1	36.4	39.6	42.8	46.0	49.2	52.3
2/10 * Pi/8	P	-0.809	-0.809	-0.808	-0.807	-0.806	-0.805	-0.803	-0.802	-0.8
	F	2.11	3.0	4.1	5.44	7.03	8.9	11.1	13.6	16.4
	S	52.8	60.8	69.0	77.3	85.9	94.7	104.0	113.0	122.0
	D	27.0	30.3	33.5	36.8	40.1	43.3	46.5	49.8	53.0
3/10 * Pi/8	P	-0.809	-0.808	-0.807	-0.806	-0.805	-0.804	-0.803	-0.801	-0.8
	F	2.1	2.98	4.08	5.41	6.99	8.85	11.0	13.5	16.3
	S	51.7	59.4	67.3	75.4	83.7	92.2	101.0	110.0	119.0
	D	27.3	30.6	34.0	37.3	40.6	43.9	47.2	50.4	53.7
4/10 * Pi/8	P	-0.808	-0.808	-0.807	-0.806	-0.805	-0.803	-0.802	-0.801	-0.799
	F	2.09	2.97	4.06	5.38	6.95	8.8	10.9	13.4	16.2
	S	50.6	58.1	65.8	73.7	81.8	90.0	98.4	107.0	116.0
	D	27.6	31.0	34.4	37.8	41.1	44.5	47.8	51.1	54.4
5/10 * Pi/8	P	-0.808	-0.807	-0.806	-0.805	-0.804	-0.803	-0.801	-0.8	-0.798
	F	2.08	2.95	4.03	5.34	6.9	8.73	10.9	13.3	16.0
	S	49.7	57.0	64.4	72.1	79.9	88.0	96.2	105.0	113.0
	D	28.0	31.4	34.9	38.3	41.7	45.1	48.5	51.8	55.2
6/10 * Pi/8	P	-0.807	-0.806	-0.805	-0.804	-0.803	-0.802	-0.8	-0.799	-0.797
	F	2.07	2.93	4.0	5.3	6.85	8.67	10.8	13.2	15.9
	S	48.8	55.9	63.2	70.6	78.3	86.1	94.0	102.0	111.0
	D	28.4	31.9	35.4	38.8	42.3	45.7	49.2	52.6	56.0
7/10 * Pi/8	P	-0.806	-0.805	-0.804	-0.803	-0.802	-0.8	-0.799	-0.798	-0.796
	F	2.05	2.91	3.97	5.26	6.79	8.59	10.7	13.1	15.8
	S	47.9	54.9	62.0	69.2	76.7	84.3	92.1	100.0	108.0
	D	28.8	32.3	35.9	39.4	42.9	46.4	49.9	53.4	56.9
8/10 * Pi/8	P	-0.804	-0.803	-0.802	-0.801	-0.8	-0.799	-0.798	-0.796	-0.795
	F	2.03	2.88	3.94	5.21	6.74	8.52	10.6	12.9	15.6
	S	47.1	53.9	60.8	67.9	75.2	82.6	90.2	98.0	106.0
	D	29.2	32.8	36.4	40.0	43.6	47.1	50.7	54.2	57.8
9/10 * Pi/8	P	-0.803	-0.802	-0.801	-0.8	-0.799	-0.798	-0.796	-0.795	-0.793
	F	2.02	2.86	3.9	5.17	6.67	8.44	10.5	12.8	15.5
	S	46.4	53.0	59.8	66.7	73.8	81.0	88.4	96.0	104.0
	D	29.7	33.3	37.0	40.6	44.3	47.9	51.5	55.1	58.7
10/10 * Pi/8	P	-0.801	-0.8	-0.799	-0.798	-0.797	-0.796	-0.794	-0.793	-0.792
	F	2.0	2.83	3.87	5.12	6.61	8.36	10.4	12.7	15.3
	S	45.6	52.1	58.7	65.5	72.4	79.4	86.7	94.1	102.0
	D	30.2	33.9	37.6	41.3	45.0	48.7	52.4	56.1	59.7