



Universität Karlsruhe (TH)
Forschungsuniversität · gegründet 1825

Institut für Mechanik
und
Fakultät für Mathematik

Modellierung gekrümmter Balken bei beliebigen großen Deformationen und Kontakt

Diplomarbeit
zur Erlangung des akademischen Grades
“Diplom-Technomathematiker”

von
Baldur Georg Allert
am
Institut für Mechanik
und
Fakultät für Mathematik
Universität Karlsruhe (TH)

Themensteller / Betreuer:
Prof. Dr.-Ing. Karl Schweizerhof
Dr. Alexander Konyukhov

Gutachter:
Prof. Dr.-Ing. Karl Schweizerhof
Prof. Dr. Andreas Rieder

Beginn der Diplomarbeit: 1. Oktober 2008
Abgabe der Diplomarbeit: 31. März 2009

Zusammenfassung

Ziel dieser Arbeit ist es, eine geometrisch exakte Balkentheorie herzuleiten und mittels Finiter Elemente zu implementieren. Dabei wird die kinematische Hypothese nach Timoshenko verwendet, die auch Schubverzerrungen miteinbezieht.

Die mittels dieser Modellierung entstehende Theorie ist nichtlinear. Außerdem müssen die Orientierungen der Balkenquerschnitte durch Rotationen dargestellt werden. Die Menge der Rotationen bildet jedoch keinen Vektorraum, sondern eine Mannigfaltigkeit. Ein Hauptaspekt der Arbeit liegt somit auf der mathematischen Darstellung von Rotationen. Außerdem muss ein geeigneter Ableitungsbegriff gefunden werden, um das Energiepotential minimieren zu können.

Das nichtlineare, kontinuierliche Problem wird durch die Finite Elemente Methode diskretisiert und mit einem Newton-Schema linearisiert. Finite Elemente Ansätze bis zu kubischer Ordnung werden dabei definiert. Dadurch besteht die Möglichkeit einer nahezu exakten Darstellung von gekrümmten Geometrien sowie der Verwendung von stetig differenzierbaren Splines.

Ein weiterer Gesichtspunkt dieser Arbeit ist die Berücksichtigung von Kontaktbedingungen. Es wird lediglich reibungsloser, punktwiser Kontakt zwischen Balken mit Kreisquerschnitten betrachtet.

Um den Einsatzbereich und die Grenzen der vorliegenden Implementierung zu bestimmen, werden einige Beispielrechnungen vorgestellt und analysiert.

Abstract

The aim of this thesis is the definition of a geometrically exact beam theory which is subsequently implemented by the finite element method. Here, the kinematic hypothesis according to Timoshenko is used which also takes shear strain into account.

The theory resulting from this modeling approach is nonlinear. In addition, the orientations of the cross-sections have to be described by rotations. However, the set of rotations does not form a vector space, but a manifold. A main aspect of this thesis consists of the mathematical representation of rotations. Moreover, an adequate term for the derivative has to be defined, to enable the minimization of the potential energy.

The nonlinear continuous problem is being discretized by the method of finite elements and linearized by a Newton scheme. This includes the definition of finite element approaches up to cubic order. Hence a nearly exact representation of curved geometry is possible, as well as the usage of continuously differentiable splines.

Another aspect of this thesis is the incorporation of conditions of contact. Note that only pointwise contact, disregarding friction, is considered.

A number of numerical examples, with and without contact, are computed and analyzed in order to determine the range of application of the present the approach.

Aufgabenstellung

Modellierung gekrümmter Balken bei beliebigen großen Deformationen und Kontakt

Die Modellierung beliebig gekrümmter Balken wird bei vielen Anwendungen benötigt. Dies sind Kabelstrukturen, wie auch Balken- und Fachwerkstrukturen mit beliebigen Querschnitten. Der Schwerpunkt der Arbeit soll auf der mathematisch-mechanischen Modellierung von stark deformierbaren dünnen Balken mit Kreisquerschnitt mit der Finite Elemente Methode liegen. Hierbei sollen beliebige Ansatzordnungen betrachtet werden. Folgende Einzelpunkte sind speziell zu bearbeiten.

1. Betrachtung unterschiedlicher Modelle für endliche Rotationen: Darstellung über Matrizen, Rotationsvektoren und Quaternionen.
2. Variationelle Formulierung unter Einbeziehung großer Rotationen.
3. Linearisierung im Rahmen einer iterativen Lösung des nichtlinearen Problems. Dabei sind unterschiedliche Vorgehensweisen bei der Aufdatierung der Rotationen zu untersuchen.
4. Implementierung in das institutseigene FE-Programm FEAP-MeKa, d.h. Balkenelement mit endlichen Rotationen höherer Ansatzordnung. Ein Element mit linearer Ansatzordnung und kleinen Rotationen liegt bereits vor.
5. Numerische Untersuchungen
 - (a) Balkenstrukturen ohne Kontakt
 - (b) Kontakt zwischen Balken - Nutzung vorhandener Kontaktalgorithmen
6. Diskussion der Ergebnisse

Eine Auswahl an relevantem Schrifttum ist angefügt. Die Arbeiten [Arg82, Sim86, Ibr97, IT02, KS05] sind im Literaturverzeichnis zu finden.

Erklärung

Hiermit versichere ich, dass ich diese Diplomarbeit selbständig verfasst und keine anderen, als die angegebenen Quellen und Hilfsmittel benutzt habe. Die Stellen meiner Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, habe ich in jedem Fall unter Angabe der Quelle kenntlich gemacht. Diese Arbeit hat in dieser oder einer ähnlichen Form noch nicht im Rahmen einer anderen Prüfung vorgelegen.

Karlsruhe, den 31. März 2009

(Baldur Allert)

Vorwort

Die vorliegende Diplomarbeit entstand von Oktober 2008 bis März 2009 am Institut für Mechanik der Universität Karlsruhe (TH).

Der Text wurde mit Hilfe des Textsatzprogrammes \LaTeX erstellt. Die Berechnungen in dieser Arbeit wurden mit dem Programm FEAP von R.L. Taylor durchgeführt. Der Programmcode, der hier vorgestellten Elementroutinen, ist ausschließlich in FORTRAN77 geschrieben. Da keine Programme zur grafischen Ausgabe vorhanden sind, die entsprechend hohe Ansatzordnungen unterstützen, wurden sämtliche Grafiken mit selbst implementierten Routinen in MATLAB erstellt.

Ich möchte mich bei Prof. Dr.-Ing. Karl Schweizerhof für die Vergabe und Betreuung der Diplomarbeit bedanken. Die sehr interessante Aufgabenstellung bot mir die Möglichkeit mechanische und mathematische Aspekte eng zu vereinen.

Dr. Alexander Konyukhov gilt mein spezieller Dank für seine sehr engagierte Betreuung der Diplomarbeit. Die zahlreichen Gespräche und vielseitigen Denkanstöße waren eine große Hilfe bei der Erstellung dieser Arbeit.

Außerdem möchte ich mich bei den Mitarbeitern des Instituts für Mechanik bedanken, die bei Problemen mit den Rechnern im Poolraum mit Rat und Tat zur Seite standen.

Ich danke meinen Freunden und besonders meiner Freundin für ihre moralische Unterstützung und dafür, dass sie während der vergangenen Monate auf viel gemeinsame Zeit verzichten mussten.

Mein ganz besonderer Dank gilt meiner Familie, die mir durch ihren fortwährenden Rückhalt diese Arbeit ermöglichte und sie mit großem Interesse verfolgte.

Karlsruhe im März 2009

Baldur Allert

Inhaltsverzeichnis

1	Einleitung	1
2	Mathematische Grundlagen	3
2.1	Allgemeine Bezeichnungen	3
2.2	Rotationen im dreidimensionalen Raum	3
2.2.1	Infinitesimale Rotationen	4
2.2.2	Endliche Rotationen der Rotationsgruppe SO_3	4
2.3	Die Gâteaux-Variation als Ableitungsbegriff	6
2.4	Quaternionen	8
2.4.1	Der Körper \mathbb{H}	8
2.4.2	Quaternionen zur Darstellung von Rotationen	10
2.5	Bézier-Kurven und Spline-Interpolation	12
2.5.1	Bernstein-Polynome	12
2.5.2	Darstellung von Kreisen durch rationale Bézier-Kurven	14
2.5.3	Spline Darstellung von C^1 -Kurven	15
3	Mechanisches Modell	17
3.1	Konfiguration	17
3.2	Gleichgewicht	19
3.3	Kinematik	21
3.4	Stoffgesetz	21
3.5	Explizite Form der Balkengleichungen	22
3.6	Prinzip der virtuellen Verschiebungen	23
3.7	Prinzip des minimalen Energiepotentials	24
4	Finite Elemente Methode	27
4.1	Allgemeines	27
4.2	Finite Elemente Diskretisierung	27
4.2.1	Die Ansatzfunktionen $\mathbf{A}, \mathbf{A}_{ref}$	29

4.2.2	Berechnung der Jacobimatrix J	30
4.2.3	Update der Verschiebung \mathbf{u} und der Rotationsmatrix $\mathbf{\Lambda}$	30
4.2.4	Alternative Ansätze zum Update der Rotationsmatrix $\mathbf{\Lambda}$	32
4.2.5	Update der Verzerrungen $\boldsymbol{\varepsilon}$ und $\boldsymbol{\kappa}$	33
4.2.6	Berechnung der Spannungsresultierenden \mathbf{n} und \mathbf{m}	35
4.2.7	Berechnung der tangentialen Steifigkeitsmatrix \mathbf{TANG}	35
4.2.8	Berechnung des Residuums \mathbf{RES}	37
4.2.9	Zusammenbau der globalen Steifigkeitsmatrix	38
4.3	Newton-Iteration	38
4.3.1	Schrittweise Lastauftragung	40
4.3.2	Eine Alternative zum Newton-Verfahren	40
5	Elementroutinen	41
5.1	2-Knoten Balkenelemente	41
5.1.1	Kleine Deformationen - <i>elmt33</i>	42
5.1.2	Endliche Deformationen - <i>elmt82</i>	43
5.2	Das 3-Knoten Balkenelement - <i>elmt83</i>	43
5.2.1	Lagrange Ansatz	43
5.2.2	Hierarchischer Ansatz	44
5.2.3	Rationale Bézier-Kurven	45
5.2.4	Kreisgeometrie	47
5.3	Das 4-Knoten Balkenelement - <i>elmt84</i>	48
5.3.1	Lagrange Ansatz	48
5.3.2	Hierarchischer Ansatz	49
5.3.3	Rationale Bézier-Kurven	49
5.3.4	C_1 -Spline Ansatz	50
5.4	Die Elemente <i>elmt86</i> , <i>elmt87</i> und <i>elmt88</i>	52
5.5	Die Kontaktelemente <i>elmt90</i> , <i>elmt91</i> und <i>elmt92</i>	52
6	Kontakt	53
6.1	Generelles Vorgehen	54
6.2	Abstandsberechnung	54
6.2.1	Geometrie und krummlinige Koordinatensysteme	54
6.2.2	Durchdringung und Kontaktkraft	57
6.3	Kinematik	57
6.4	Variationsformulierung und Linearisierung	58
6.5	Implementierung	59

6.5.1	Ansatzfunktionen	59
6.5.2	Globale Suche durch Newton-Iteration	60
6.5.3	Lokale Suche	60
6.5.4	Die Steifigkeitsmatrix $TANG_{kont}$	61
6.5.5	Das Residuum RES_{kont}	61
7	Elementtests	63
7.1	Gemeinsamkeiten mit der linearen Balkentheorie	63
7.1.1	Eingespannter gerader Balken mit Einzelmoment	64
7.1.2	Eingespannter gerader Balken mit Einzelkraft	66
7.1.3	Eingespannter Viertelkreisbogen mit Einzelkraft	69
7.1.4	Federmodell und Torsionsbelastung	72
7.2	Auf- und Abrollen von Balken	75
7.2.1	Aufrollen eines geraden Balkens	75
7.2.2	Abrollen eines Ringes	78
7.2.3	Aufrollen eines Balkens unter Störeinfluss	82
7.3	Ein dreidimensionaler Test nach Bathe	84
7.4	Virtueller Ringzugversuch	87
7.5	Kontaktprobleme	91
7.5.1	Kontakt zwischen Balken	91
7.5.2	Kontakt zwischen Ringen	95
7.5.3	Kontakt bei Knoten	98
8	Ergebnisse und Ausblick	103
8.1	Reduzierte Integration	103
8.2	Tangentiale Steifigkeitsmatrix $TANG$	104
8.3	Zahl der Iterationsschritte	105
8.4	Kontakt	105
8.5	Fazit	106
A	Rechenregeln und Beweise	107
A.1	Rechenregeln	107
A.2	Beweis zu Satz 2.3 der Rotationsabbildung	110
A.3	Beweis zu Satz 3.2 der endlichen Verzerrungen	112
A.4	Beweis zu Satz 3.3 der ersten Potentialvariation	113
A.5	Beweis zu Satz 3.3 der zweiten Potentialvariation	115
B	Materialparameter	117

Kapitel 1

Einleitung

Die in dieser Arbeit betrachtete, geometrisch exakte Balkentheorie wurde zuerst von Reissner [Rei72, Rei81] vorgestellt und ist daher auch nach ihm benannt. Von Bathe und Bolourchi [BB79] wurde eine Finite Elemente Implementierung noch ohne Newton-Schema erarbeitet. Zahlreiche Arbeiten von Simo [Sim85, Sim86] beinhalten ein Newton-Schema. Außerdem behandelt Simo [Sim92] die Frage nach einer entsprechenden Definition des Ableitungsoperators für Rotationen. Dabei bleibt die Referenzgeometrie jedoch auf gerade Balken beschränkt. Ibrahimbegovic [Ibr95, IFK95, Ibr97] berücksichtigt auch gekrümmte Referenzgeometrien. Er vergleicht außerdem die Verwendung von Rotationsvektoren gegenüber Rotationsmatrizen oder Quaternionen.

Zahlreiche Arbeiten zur Berücksichtigung von Kontakt bei mechanischen Problemen stammen von Wriggers, exemplarisch sei auf [Wri06] verwiesen. Eine veränderte Definition des Koordinatensystems an der Kontaktstelle von Konyukhov und Schweizerhof [KS05] führt zu einer geometrisch besser zu interpretierenden Steifigkeitsmatrix. Dieser Ansatz wird in der vorliegenden Arbeit berücksichtigt.

Aufbau der Arbeit

Die mathematischen Grundlagen zur Beschreibung von Rotationen werden in Kapitel 2 erläutert. Dabei gehen wir auf die Darstellung von Rotationen durch Rotationsvektoren, Matrizen und Quaternionen ein. Außerdem wird durch die kovariante Ableitung ein Ableitungsbegriff auf der Menge der Rotationen definiert. Daneben sind in diesem Kapitel einige Grundlagen zu rationalen Bézier-Kurven und kubischen Splines zu finden.

Im darauffolgenden Kapitel wird das mechanische Modell hergeleitet. Dabei wenden wir die Hypothese nach Timoshenko an. Neben der Kinematik und dem Stoffgesetz formulieren wir die Gleichgewichtsbedingungen. Diese definieren in ihrer schwachen, variationellen Formulierung das zu minimierende Energiepotential.

Dieses Problem wird in Kapitel 4 diskretisiert und linearisiert. Dabei wird die Finite Elemente Methode eingeführt und der allgemeine Aufbau der Elementroutinen dargestellt. Das so diskretisierte Problem wird durch das Newton-Verfahren linearisiert und kann somit durch ein lineares Gleichungssystem beschrieben und gelöst werden.

Danach wird im fünften Kapitel der exakte Aufbau der verschiedenen Elementroutinen und die Wahl der Ansatzfunktionen erläutert. Elemente höherer Ansatzordnung ermöglichen es, neben polynomialen Kurven auch rationale Funktionen, kubische Splines oder nichtlineare Kreisgeometrien einzubinden.

Das sechste Kapitel stellt eine Theorie zur Berücksichtigung von Nebenbedingungen durch Kontakt zwischen Balken vor. Zusätzlich zur der Abstandsberechnung und der Kinematik wird auch die Variationsformulierung dargestellt. In diesem Kapitel finden sich auch Einzelheiten zur Implementierung von Kontaktelementen, analog zu den Balkenelementen in Kapitel 4.

Verschiedene Elementtests und Beispielprobleme, mit und ohne Kontakt, werden in Kapitel 7 berechnet. Dabei können manche Ergebnisse mit analytischen Lösungen oder mit Vergleichswerten aus der Literatur verglichen werden.

Zuletzt ist in Kapitel 8 ein Fazit der hier vorgestellten Theorie und der numerischen Implementierung gegeben. Probleme und Grenzen der hier implementierten Finiten Elemente werden aufgezeigt und mögliche Lösungsansätze vorgestellt.

In Anhang A sind Ergänzungen zu den mathematischen Grundlagen und einige Beweise abgedruckt. Hinweise zur Verwendung der Materialparameter in den Elementroutinen sind in Anhang B zu finden.

Kapitel 2

Mathematische Grundlagen

In diesem Kapitel werden die mathematischen Definitionen und Sätze hergeleitet, die später für die Berechnung des mechanischen Problems benötigt werden.

Zuerst werden einige Bezeichnungen festgelegt. Der nächste Abschnitt behandelt die Darstellung von Rotationen mittels Rotationsmatrizen. Daraufhin definieren wir mit der kovarianten Ableitung einen Ableitungsbegriff auf der Menge der Rotationen. Im nachfolgenden Abschnitt wird die Theorie der Quaternionen betrachtet. Schließlich definieren wir Bézier-Kurven und kubische Splines. Zusätzliche Rechenformeln und längere Beweise sind auch im Anhang A zu finden.

2.1 Allgemeine Bezeichnungen

Wir führen folgende allgemeine Bezeichnungen ein. Leider können diese nicht immer streng eingehalten werden. Ausnahmen sind jedoch entsprechend gekennzeichnet.

Skalare und Mengen werden mit dünn gedruckten Buchstaben bezeichnet, z.B. a, B, c, D oder α, β, γ . Vektoren sowie vektorwertige Funktionen sind fett gedruckte Kleinbuchstaben, z.B. $\mathbf{a}, \mathbf{b}, \mathbf{c}$ oder $\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}$ und soweit nicht anders vermerkt, als Spaltenvektoren zu verstehen. Matrizen werden hingegen mit fett gedruckten Großbuchstaben bezeichnet, z.B. $\mathbf{A}, \mathbf{B}, \mathbf{C}$ oder $\boldsymbol{\Lambda}, \boldsymbol{\Omega}$.

Die zu einem Vektor $\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \in \mathbb{R}^3$ zugehörige schiefsymmetrische Matrix $\hat{\mathbf{A}} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix}$ wird mit einem Dach kenntlich gemacht.

Die Einheitsmatrix im $\mathbb{R}^{n \times n}$ wird mit \mathbf{I}_n bezeichnet, Nullvektoren und die Nullmatrix mit $\mathbf{0}$.

Quaternionen sind durch Kleinbuchstaben in Frakturschrift, z.B. $\mathfrak{q}, \mathfrak{q}_\Lambda, \mathfrak{r}$ gekennzeichnet.

\mathbb{R} bezeichnet die reellen Zahlen, \mathbb{Z} die ganzen Zahlen, \mathbb{N}_0 und \mathbb{N}^+ die natürlichen Zahlen, mit bzw. ohne 0.

In dieser Arbeit wird die Eulersche Summenkonvention nicht angewendet. Sämtliche Additionen sind vielmehr explizit angegeben.

2.2 Rotationen im dreidimensionalen Raum

In diesem Abschnitt leiten wir die theoretischen Grundlagen zur Beschreibung von Rotationen her, die wir später zur Berechnung der Orientierung der lokalen Koordinatensysteme benötigen.

Die hier betrachteten Rotationen drehen immer um den Ursprung des globalen Koordinatensystems. Eine Rotation um einen anderen Punkt kann in eine Rotation um den Ursprung und eine Translation aufgeteilt werden. Somit stellt dies keine Einschränkung dar. Wir müssen zwischen infinitesimal kleinen Rotationen und endlichen Rotationen unterscheiden. Da die Menge der endlichen Rotationen keinen Vektorraum bildet und nicht kommutiert, können viele als selbstverständlich angesehene Sätze und Formeln hier nicht angewandt werden.

2.2.1 Infinitesimale Rotationen

Bei infinitesimal kleinen Rotationen gelten zahlreiche Vereinfachungen, die bei endlichen Rotationen nicht weiter gemacht werden dürfen.

Insbesondere nehmen wir bei Hintereinanderausführung von infinitesimalen Rotationen an, dass jede der Rotationen auf die Ausgangslage angewandt wird. Dadurch ist die Reihenfolge der Rotationen vertauschbar. Außerdem können wir somit jede infinitesimale Rotation als Kombination dreier Rotationen um die kartesischen Koordinatenachsen \mathbf{e}_i ($i = 1, 2, 3$) auffassen. Dies definiert eine mögliche Darstellung für infinitesimale Rotationen

$$\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix},$$

wobei w_i den Drehwinkel im Bogenmaß um die Koordinatenachse \mathbf{e}_i angibt. Eine Drehung um den Winkel $w_i = 2\pi$ entspricht einer vollen Drehung in die Ausgangslage zurück.

Offensichtlich ist $\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ das neutrale Element. Die infinitesimal kleinen Rotationen sind isomorph zu \mathbb{R}^3 . Die Hintereinanderausführung entspricht der Vektoraddition. Im Folgenden bezeichnen wir mit $\theta := \|\mathbf{w}\| := \sqrt{w_1^2 + w_2^2 + w_3^2}$ die euklidische Norm von \mathbf{w} .

Bei infinitesimalen Rotationen gibt \mathbf{w} auch die Richtung der Rotationsachse an. Der Winkel der Rotation entspricht dabei θ .

2.2.2 Endliche Rotationen der Rotationsgruppe SO_3

Bei endlichen Rotationen gelten die im vorherigen Abschnitt gemachten Vereinfachungen nicht. Insbesondere ist die Reihenfolge der Rotationen nicht vertauschbar. Der Vektor \mathbf{w} gibt hier die Rotationsachse an, $\theta = \|\mathbf{w}\|$ den zugehörigen Winkel. Dies ist eine erste Möglichkeit zur Darstellung von endlichen Rotationen. Diese Darstellung hat den Vorteil, dass nur drei reelle Skalare zur Beschreibung einer Rotation benötigt werden. Allerdings ist die Beschreibung durch den Rotationsvektor \mathbf{w} nicht eindeutig, d.h. $\mathbf{w} \neq \mathbf{v}$ beschreiben unter Umständen die gleiche Rotation.

Auch die Darstellung über Eulersche Winkel benötigt nur drei reelle Skalare, wir gehen auf diese Darstellung jedoch nicht weiter ein. Es kann gezeigt werden, dass alle Darstellungen von Rotationen, die mit drei Skalaren auskommen, nicht singularitätenfrei sein können.

Jede Rotation kann allgemein als Abbildung betrachtet werden, die jedem $\mathbf{x} \in \mathbb{R}^3$ ein $\mathbf{y} \in \mathbb{R}^3$ zuordnet. Da der Ursprung ein Fixpunkt jeder Rotation ist, kann leicht nachgeprüft werden, dass Rotationen lineare Abbildungen sind und somit als Matrix dargestellt werden können. Die Menge der Rotationen \mathbf{A} ist somit eine Teilmenge von $\mathbb{R}^{3 \times 3}$.

Rotationen sind Isometrien, da sie langlen- und winkeltreu sind. Nach [RT00], Kapitel 21, gilt deshalb

$$\mathbf{\Lambda}\mathbf{\Lambda}^T = \mathbf{I}_3, \quad \det \mathbf{\Lambda} = 1.$$

Definition 2.1 (SO_3 und so_3):

Es gilt:

- Die Menge $SO_3 := \{\mathbf{\Lambda} \in \mathbb{R}^{3 \times 3} : \mathbf{\Lambda}^T \mathbf{\Lambda} = \mathbf{I}_3, \quad \det \mathbf{\Lambda} = 1\}$ heist Menge der Rotationen ber \mathbb{R}^3 .
- Die Teilmenge aller schiefsymmetrischen Matrizen in $\mathbb{R}^{3 \times 3}$ bezeichnen wir im Folgenden mit $so_3 := \{\widehat{\mathbf{W}} \in \mathbb{R}^{3 \times 3} : \widehat{\mathbf{W}}^T = -\widehat{\mathbf{W}}\}$.

Satz 2.2 (Eigenschaften von SO_3):

Es gilt:

- SO_3 ist eine nicht abelsche Gruppe bzgl. der Matrizenmultiplikation. Geometrisch entspricht dies der Hintereinanderausfhrung der Rotationen.
- SO_3 ist eine differenzierbare Mannigfaltigkeit, aber kein Vektorraum.
- Sei $\mathbf{\Lambda} \in SO_3$, dann ist der Tangentialraum $T_{\mathbf{\Lambda}}SO_3$ zu SO_3 an der Stelle $\mathbf{\Lambda}$ gegeben durch $T_{\mathbf{\Lambda}}SO_3 := \{\widehat{\mathbf{W}}\mathbf{\Lambda} : \widehat{\mathbf{W}} \in so_3\}$. Dieser ist isomorph zu \mathbb{R}^3 .

Die Beweise sind z.B. bei Knorrer [Kn96] zu finden.

Der Tangentialraum $T_{\mathbf{I}_3}SO_3$ der Identitat besteht gerade aus den schiefsymmetrischen Matrizen. Der folgende Satz setzt den Rotationsvektor $\mathbf{w} \in \mathbb{R}^3$ mit der Rotationsabbildung $\mathbf{\Lambda} \in SO_3$ in Beziehung.

Satz 2.3 (Rotationsabbildung):

Sei $\mathbf{w} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix} \in \mathbb{R}^3$ ein Rotationsvektor und $\widehat{\mathbf{W}} := \begin{pmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{pmatrix}$ die zugehrige schiefsymmetrische Matrix, sowie $\theta = \|\mathbf{w}\|$, dann ist

$$\mathbf{\Lambda} = \exp \widehat{\mathbf{W}} \in SO_3$$

die zugehrige Rotationsabbildung. Auerdem gelten folgende Identitaten, die auch als Rodrigues-Formel bezeichnet werden.

$$\mathbf{\Lambda} = \exp \widehat{\mathbf{W}} \tag{2.1}$$

$$= \mathbf{I}_3 + \frac{\sin \theta}{\theta} \widehat{\mathbf{W}} + \frac{1 - \cos \theta}{\theta^2} \widehat{\mathbf{W}}^2 \tag{2.2}$$

$$= \mathbf{I}_3 + \frac{\sin \theta}{\theta} \widehat{\mathbf{W}} + \frac{1}{2} \left(\frac{\sin(\theta/2)}{\theta/2} \right)^2 \widehat{\mathbf{W}}^2 \tag{2.3}$$

$$= \cos \theta \mathbf{I}_3 + \frac{\sin \theta}{\theta} \widehat{\mathbf{W}} + \frac{1 - \cos \theta}{\theta^2} \mathbf{w}\mathbf{w}^T \tag{2.4}$$

Neben dem in Anhang A dargestellten geometrischen Beweis kann die Aussage auch mit Hilfe der Lie-Theorie bewiesen werden.

Bei mehreren Rotationen ist die Reihenfolge der Ausführung entscheidend. Wir unterscheiden dabei Rotationen um raumfeste Achsen und Rotationen um körperfeste Achsen. Für die spätere Anwendung benötigen wir raumfeste Achsen.

Seien $\Lambda_1, \Lambda_2 \in SO_3$ zwei Rotationen. Dann ist die kombinierte Rotation “ Λ_1 dann Λ_2 ” gegeben durch

$$\Lambda_{12} = \Lambda_2 \Lambda_1.$$

Zusätzliche Rotationen müssen also immer von rechts hinzumultipliziert werden. Dies ist für die spätere Aufdatierung inkrementeller Rotationen wichtig.

Ein weiterer wichtiger Punkt ist die Definition einer Metrik auf der Menge der Rotationen. Hierfür gibt es verschiedene Möglichkeiten. Einerseits kann eine auf $\mathbb{R}^{3 \times 3}$ definierte Metrik, wie z.B. die Spektralnorm, auf die Teilmenge SO_3 eingeschränkt werden. Andererseits existiert die Möglichkeit auf Mannigfaltigkeiten eine Riemannsche Metrik zu definieren. Dabei wird eine auf den Tangentialräumen existierende Metrik durch die Exponentialabbildung auf die Mannigfaltigkeit übertragen. Für eine genauere Betrachtung sei auf Schottenloher [Sch95], Anhang G, verwiesen.

2.3 Die Gâteaux-Variation als Ableitungsbegriff

Um später das Funktional der potentiellen Energie zu minimieren, müssen wir einen geeigneten Ableitungsbegriff definieren. Zuerst definieren wir mit der Gâteaux-Variation eine Verallgemeinerung der Richtungsableitung des \mathbb{R}^n auf beliebige, auch unendlichdimensionale, Vektorräume.

Definition 2.4 (Gâteaux-Variation):

Sei V ein reeller Vektorraum, $D \subset V$. Gegeben sind ein Funktional $\Pi : D \rightarrow \mathbb{R}$ sowie Punkte $\mathbf{x} \in D$ und $\mathbf{y} \in V$, sodass gilt

$$\exists \alpha_0 > 0 \forall \alpha \in (-\alpha_0, \alpha_0) : \mathbf{x} + \alpha \mathbf{y} \in D.$$

Dann definieren wir die Gâteaux-Variation von Π an der Stelle \mathbf{x} in Richtung \mathbf{y} durch

$$\delta \Pi(\mathbf{x}, \mathbf{y}) := \lim_{\alpha \rightarrow 0} \frac{1}{\alpha} (\Pi(\mathbf{x} + \alpha \mathbf{y}) - \Pi(\mathbf{x})). \quad (2.5)$$

Dies ist auch bei Meyberg [MV03], Kapitel 13, beschrieben.

Allerdings ist das später zu untersuchende Energiepotential Π nicht auf einem Vektorraum, sondern auf einer Mannigfaltigkeit $M = \mathbb{R}^3 \times SO_3$ definiert. Wenn wir \mathbf{y} als Richtung der Ableitung auffassen, dann liegt \mathbf{y} somit im zugehörigen Tangentialraum $T_{\mathbf{x}}M$ des Punktes \mathbf{x} . Im Allgemeinen gilt jedoch $\forall \alpha \neq 0 : \mathbf{x} + \alpha \mathbf{y} \notin M$. Somit ist die Gâteaux-Variation in der obigen Form nicht auf Mannigfaltigkeiten definierbar. Wir suchen nach einer Möglichkeit, den obigen Ableitungsbegriff auf differenzierbare Mannigfaltigkeiten zu erweitern, sodass der Spezialfall einer Mannigfaltigkeit, die gleichzeitig ein Vektorraum ist, mit der obigen Definition übereinstimmt. Der Artikel von Simo [Sim92] dient dabei neben Schottenloher [Sch95], Anhänge M und G, als Grundlage.

Um die Struktur der Mannigfaltigkeit M und des Tangentialraums $T_{\mathbf{x}}M$ zu verdeutlichen, werden Elemente $x \in M$ dünn gedruckt, Vektoren $\mathbf{y} \in T_{\mathbf{x}}M$ jedoch fett.

Definition 2.5 (Kovariante Ableitung oder Lie-Ableitung):

Sei M eine differenzierbare Mannigfaltigkeit, $x \in M$ und $T_x M$ der Tangentialraum an der Stelle x .

Sei $\mathbf{y} \in T_x M$ beliebig und $\epsilon > 0$. Wir wählen eine glatte Kurve

$$\gamma : \begin{cases} [-\epsilon, \epsilon] & \rightarrow M \\ \alpha & \mapsto \gamma(\alpha) \end{cases}$$

mit

$$\gamma(0) = x \qquad \lim_{\alpha \rightarrow 0} \frac{\partial}{\partial \alpha} \gamma(\alpha) = \mathbf{y}.$$

Sei $\Pi : M \rightarrow \mathbb{R}$ ein Funktional, dann ist die erste kovariante Ableitung definiert durch

$$\delta \Pi(x, \mathbf{y}) := \lim_{\alpha \rightarrow 0} \frac{\partial}{\partial \alpha} \Pi(\gamma(\alpha)). \quad (2.6)$$

Die Wohldefiniertheit, d.h. die Unabhängigkeit der Ableitung von der gewählten Kurve γ ist noch zu zeigen. Außerdem ist der Begriff einer glatten Kurve hier nicht definiert. In der oben angegebenen Literatur [Sch95] sind diese Punkte jedoch ausführlich dargestellt.

Bemerkung 2.6

Im Spezialfall, dass die differenzierbare Mannigfaltigkeit auch ein Vektorraum ist, können wir für die Kurve $\gamma = \alpha \mathbf{y}$ wählen. Die Definition der kovarianten Ableitung entspricht dann jener der Gâteaux-Variation. Deshalb bezeichnen wir die kovariante Ableitung auch als Variation, um später die in der mechanischen Modellbildung gebräuchlichen Begriffe, wie z.B. das Variationsprinzip, weiter verwenden zu können.

Im Folgenden bezeichnen die Operatoren δ und Δ sowohl die kovariante Ableitung bzw. die Gâteaux-Variation, als auch die Richtungen, bzgl. der die Ableitung definiert ist. Diese, aus mathematischer Sicht weniger angebrachte Schreibweise, ermöglicht es somit, die in der Mechanik üblichen Bezeichnungen zu übernehmen.

Satz 2.7

Die kovariante Ableitung ist ein linearer Operator. Außerdem gilt die Produktregel.

$$\begin{aligned} \delta(\Pi + \Omega)(x, \mathbf{y}) &= \delta \Pi(x, \mathbf{y}) + \delta \Omega(x, \mathbf{y}) \\ \delta(c\Pi)(x, \mathbf{y}) &= c \delta \Pi(x, \mathbf{y}) \\ \delta(\Pi\Omega)(x, \mathbf{y}) &= \delta \Pi(x, \mathbf{y}) \Omega + \Pi \delta \Omega(x, \mathbf{y}) \end{aligned}$$

Dies ist bei Schottenloher [Sch95] in Anhang M dargestellt.

Um später das Funktional Π minimieren zu können, benötigen wir auch die zweite kovariante Ableitung.

Definition 2.8 (Zweite Ableitung):

Seien M , x , $T_x M$, \mathbf{y} , γ , α , ϵ wie oben. Außerdem sei $\mathbf{z} \in T_x M$. Wir wählen eine weitere glatte Kurve

$$\tilde{\gamma} : \begin{cases} [-\epsilon, \epsilon] & \rightarrow M \\ \beta & \mapsto \tilde{\gamma}(\beta) \end{cases}$$

mit

$$\tilde{\gamma}(0) = x \qquad \lim_{\beta \rightarrow 0} \frac{\partial}{\partial \beta} \tilde{\gamma}(\beta) = z.$$

Sei $\Pi : M \rightarrow \mathbb{R}$ ein Funktional, dann ist die zweite kovariante Ableitung definiert durch

$$\Delta \delta \Pi(x, \mathbf{y}, z) := \lim_{\beta \rightarrow 0} \frac{\partial}{\partial \beta} \delta \Pi(\tilde{\gamma}(\beta), \mathbf{y}). \quad (2.7)$$

Bemerkung 2.9

Nach [Sim92] gilt im Allgemeinen

$$\Delta \delta \Pi(x, \mathbf{y}, z) \neq \Delta \delta \Pi(x, z, \mathbf{y}),$$

d.h. die Reihenfolge der Ableitungen ist erheblich. Im Spezialfall $\delta \Pi(x, \mathbf{y}) = 0$ folgt allerdings Gleichheit.

2.4 Quaternionen

Quaternionen sind eine Erweiterung der reellen Zahlen, sogenannte hyperkomplexe Zahlen, die mehr als eine imaginäre Einheit haben. Bei Quaternionen gibt es drei imaginäre Einheiten, sodass sie mit einem 4-zeiligen Vektor vollständig beschrieben werden können. Sie eignen sich in besonderem Maße, um Rotationen darzustellen, wozu sie im Folgenden auch genutzt werden.

Grundlage des folgenden Kapitels ist, neben dem Buch von Kuipers [Kui99], Kapitel 5, auch die Arbeit von Argyris [Arg82]. Dort wird die Verwendung von Quaternionen zur Darstellung von Rotationen im Anhang behandelt. Außerdem ist eine zur späteren Implementierung wichtige Formel aus [Sim86] entnommen.

Wir bezeichnen Quaternionen mit einem Kleinbuchstaben in Frakturschrift, meist ein \mathfrak{q}_Λ mit tiefgestellter zugehöriger Rotationsmatrix Λ .

2.4.1 Der Körper \mathbb{H}

Definition 2.10 (Quaternion):

Seien i, j, k drei imaginäre Einheiten für die gelte

$$i^2 = j^2 = k^2 = ijk = -1. \quad (2.8)$$

Seien $q_0, q_1, q_2, q_3 \in \mathbb{R}$. Dann heißt

$$\mathfrak{q} := q_0 + q_1 i + q_2 j + q_3 k \quad (2.9)$$

eine Quaternion. Das Zeichen $+$ bezeichnet hier nicht die Addition von reellen Zahlen, sondern dient im Moment nur als einfaches Symbol.

Die Menge aller Quaternionen \mathfrak{q} bezeichnen wir mit \mathbb{H} .

Wir definieren ähnlich dem Realteil und Imaginärteil einer komplexen Zahl $c \in \mathbb{C}$ außerdem den Skalarteil und den Vektorteil einer Quaternion

$$\operatorname{Re} \mathfrak{q} := q_0 \in \mathbb{R} \qquad \operatorname{Im} \mathfrak{q} := \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} \in \mathbb{R}^3.$$

Zu beachten ist, dass $\operatorname{Re} \mathfrak{q}$ einen Skalar darstellt. Dahingegen ist $\operatorname{Im} \mathfrak{q}$ trotz der gleichen Schreibweise ein Vektor. Somit können wir jede Quaternion $\mathfrak{q} \in \mathbb{H}$ schreiben als

$$\mathfrak{q} = \operatorname{Re} \mathfrak{q} + (\operatorname{Im} \mathfrak{q})^T \begin{pmatrix} i \\ j \\ k \end{pmatrix}.$$

Definition 2.11 (Rechenoperationen):

Seien $\mathfrak{q} = q_0 + q_1i + q_2j + q_3k$, $\mathfrak{r} = r_0 + r_1i + r_2j + r_3k \in \mathbb{H}$ und $a \in \mathbb{R}$. Dann definieren wir folgende Rechenoperationen für Quaternionen.

- Addition:

$$\mathfrak{q} + \mathfrak{r} := (q_0 + r_0) + (q_1 + r_1)i + (q_2 + r_2)j + (q_3 + r_3)k$$

- Skalarmultiplikation:

$$a\mathfrak{q} := (aq_0) + (aq_1)i + (aq_2)j + (aq_3)k$$

- Multiplikation:

$$\begin{aligned} \mathfrak{q} \mathfrak{r} &:= \operatorname{Re} \mathfrak{q} \operatorname{Re} \mathfrak{r} - (\operatorname{Im} \mathfrak{q})^T \operatorname{Im} \mathfrak{r} + (\operatorname{Re} \mathfrak{q} \operatorname{Im} \mathfrak{r} + \operatorname{Re} \mathfrak{r} \operatorname{Im} \mathfrak{q} + \operatorname{Im} \mathfrak{q} \times \operatorname{Im} \mathfrak{r})^T \begin{pmatrix} i \\ j \\ k \end{pmatrix} \\ &= (q_0r_0 - q_1r_1 - q_2r_2 - q_3r_3) \\ &\quad + (q_0r_1 + q_1r_0 + q_2r_3 - q_3r_2)i \\ &\quad + (q_0r_2 + q_2r_0 + q_3r_1 - q_1r_3)j \\ &\quad + (q_0r_3 + q_3r_0 + q_1r_2 - q_2r_1)k \end{aligned}$$

Die Multiplikation in \mathbb{H} ist, wie das Vektorkreuzprodukt in \mathbb{R}^3 über welches sie definiert ist, nicht kommutativ.

- Konjugierte Quaternion:

$$\bar{\mathfrak{q}} := \operatorname{Re} \mathfrak{q} + (-\operatorname{Im} \mathfrak{q})^T \begin{pmatrix} i \\ j \\ k \end{pmatrix} = q_0 + (-q_1)i + (-q_2)j + (-q_3)k$$

Addition, Skalarmultiplikation und Multiplikation sind so definiert, dass es unerheblich ist, ob wir auf der Menge der Quaternionen rechnen oder die Darstellung (2.9) verwenden und wie gewohnt mit reellen Zahlen unter Verwendung der Beziehungen (2.8) rechnen. Deshalb können wir, wie in der Definition für das Pluszeichen bereits geschehen, für beide Mengen die gleichen Operatorzeichen benutzen und müssen diese nicht streng auseinander halten, um Verwechslungen zu vermeiden.

Die Menge der Quaternionen \mathbb{H} bildet zusammen mit der Addition und der Skalarmultiplikation einen Vektorraum. \mathbb{H} ist offensichtlich isomorph zu \mathbb{R}^4 .

Satz 2.12 (Norm von Quaternionen):

Sei $\mathfrak{q} = q_0 + q_1i + q_2j + q_3k \in \mathbb{H}$ eine Quaternion, dann ist durch

$$|\mathfrak{q}| := \sqrt{\mathfrak{q} \bar{\mathfrak{q}}}$$

eine Norm gegeben. Es gilt

$$|\mathfrak{q}|^2 = q_0^2 + q_1^2 + q_2^2 + q_3^2. \quad (2.10)$$

Beweis. Wir zeigen zunächst Gleichung (2.10), dann folgen die Normaxiome (Definitheit, Homogenität und Dreiecksungleichung) direkt.

$$\begin{aligned} |\mathfrak{q}|^2 &= \mathfrak{q} \bar{\mathfrak{q}} \\ &= (\operatorname{Re} \mathfrak{q})^2 + (\operatorname{Im} \mathfrak{q})^T \operatorname{Im} \mathfrak{q} + \left(\underbrace{-\operatorname{Re} \mathfrak{q} \operatorname{Im} \mathfrak{q} + \operatorname{Re} \mathfrak{q} \operatorname{Im} \mathfrak{q}}_{=0} - \underbrace{\operatorname{Im} \mathfrak{q} \times \operatorname{Im} \mathfrak{q}}_{=0} \right)^T \begin{pmatrix} i \\ j \\ k \end{pmatrix} \\ &= q_0^2 + q_1^2 + q_2^2 + q_3^2 \end{aligned}$$

□

Satz 2.13 (Inverse Quaternionen):

Es gilt:

- Die zu $0 \neq \mathfrak{q} \in \mathbb{H}$ multiplikativ inverse Quaternion ist gegeben durch

$$\mathfrak{q}^{-1} = \frac{1}{|\mathfrak{q}|^2} \bar{\mathfrak{q}}.$$

- Seien $0 \neq \mathfrak{q}, \mathfrak{r} \in \mathbb{H}$, dann gilt

$$(\mathfrak{q}\mathfrak{r})^{-1} = \mathfrak{r}^{-1}\mathfrak{q}^{-1}.$$

- \mathbb{H} bildet zusammen mit der oben definierten Addition und Multiplikation einen Körper.

Beweis. Dass \mathfrak{q}^{-1} die Inverse ist, kann leicht nachgerechnet werden.

$$\mathfrak{q} \mathfrak{q}^{-1} = \mathfrak{q}^{-1} \mathfrak{q} = \frac{1}{|\mathfrak{q}|^2} \mathfrak{q} \bar{\mathfrak{q}} = \frac{|\mathfrak{q}|^2}{|\mathfrak{q}|^2} = 1$$

Die Produktformel muss offensichtlich nur für die Konjugation gezeigt werden.

$$\begin{aligned} \bar{\mathfrak{q}} \bar{\mathfrak{r}} &= \operatorname{Re} \mathfrak{q} \operatorname{Re} \mathfrak{r} - (\operatorname{Im} \mathfrak{q})^T \operatorname{Im} \mathfrak{r} - (\operatorname{Re} \mathfrak{q} \operatorname{Im} \mathfrak{r} + \operatorname{Re} \mathfrak{r} \operatorname{Im} \mathfrak{q} + \operatorname{Im} \mathfrak{q} \times \operatorname{Im} \mathfrak{r})^T \begin{pmatrix} i \\ j \\ k \end{pmatrix} \\ &= \operatorname{Re} \mathfrak{r} \operatorname{Re} \mathfrak{q} - (\operatorname{Im} \mathfrak{r})^T \operatorname{Im} \mathfrak{q} + (-\operatorname{Re} \mathfrak{r} \operatorname{Im} \mathfrak{q} - \operatorname{Re} \mathfrak{q} \operatorname{Im} \mathfrak{r} + \operatorname{Im} \mathfrak{r} \times \operatorname{Im} \mathfrak{q})^T \begin{pmatrix} i \\ j \\ k \end{pmatrix} \\ &= \bar{\mathfrak{r}} \bar{\mathfrak{q}} \end{aligned}$$

Die Körperaxiome prüfen wir an dieser Stelle nicht einzeln nach.

□

2.4.2 Quaternionen zur Darstellung von Rotationen

Einheitsquaternionen \mathbf{u} , d.h. Quaternionen mit Norm 1, eignen sich zur Darstellung von Rotationen.

Lemma 2.14 (Darstellung von Einheitsquaternionen):

Sei $\alpha \in [0, 2\pi]$ und $\mathbf{x} \in \mathbb{R}^3$ ein beliebiger Einheitsvektor, dann ist

$$\mathbf{u} := \cos \frac{\alpha}{2} + \sin \frac{\alpha}{2} \mathbf{x}$$

eine Einheitsquaternion. Die Abbildung ist surjektiv, aber nicht injektiv, d.h. alle Einheitsquaternionen können auf diese Weise dargestellt werden.

Beweis. Dass $|u| = 1$ folgt durch einfaches Nachrechnen. Die Surjektivität bleibt zu zeigen. Sei u eine beliebige Einheitsquaternion, dann existiert ein eindeutiges $\alpha \in [0, 2\pi]$ mit $\operatorname{Re} u = \cos \frac{\alpha}{2}$. Im Fall $\alpha \neq 0$ ist $\mathbf{x} = \frac{1}{\sin \frac{\alpha}{2}} \operatorname{Im} u$. Für $\alpha = 0$ können wir \mathbf{x} beliebig wählen. Deshalb gilt keine Injektivität. Genauer hierzu ist in [Kui99], Kapitel 5.8.2, zu finden. \square

Der folgende Satz dient als Grundlage dafür, dass Quaternionen zur Darstellung von Rotationen geeignet sind.

Satz 2.15 (Rotationsabbildung durch Quaternionen):

Sei $\mathbf{w} \in \mathbb{R}^3$ ein Rotationsvektor, $\theta = \|\mathbf{w}\|$ und $\mathbf{\Lambda} := \exp \widehat{\mathbf{W}}$ die zugehörige Rotationsmatrix. Wir definieren, wie oben $\mathbf{u}_{\mathbf{\Lambda}} := \cos \frac{\theta}{2} + \frac{1}{\theta} \sin \frac{\theta}{2} \mathbf{w}$, falls $\mathbf{w} \neq 0$. Im Fall $\mathbf{w} = 0$ setzen wir $\mathbf{u}_{\mathbf{\Lambda}} := 1$. Dann gilt für alle $\mathbf{y} \in \mathbb{R}^3$

$$(\mathbf{\Lambda} \mathbf{y})^T \begin{pmatrix} i \\ j \\ k \end{pmatrix} = \mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\eta} \overline{\mathbf{u}_{\mathbf{\Lambda}}}. \quad (2.11)$$

Dabei bezeichnet $\boldsymbol{\eta}$ die Quaternion mit Vektorteil \mathbf{y} , d.h. $\operatorname{Re} \boldsymbol{\eta} = 0$, $\operatorname{Im} \boldsymbol{\eta} = \mathbf{y}$.

Beweis. An dieser Stelle ist nur eine kurze Beweisskizze gegeben. Der vollständige Beweis kann in [Kui99], Kapitel 5.15 nachgelesen werden.

Wir zeigen zuerst, dass $\mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\eta} \overline{\mathbf{u}_{\mathbf{\Lambda}}}$ einen Vektor beschreibt, d.h. $\operatorname{Im}(\mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\eta} \overline{\mathbf{u}_{\mathbf{\Lambda}}}) = 0$. Danach ist nachzuprüfen, dass $\mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\eta} \overline{\mathbf{u}_{\mathbf{\Lambda}}}$ ein linearer Operator in \mathbf{y} ist und dass die Länge von \mathbf{y} erhalten bleibt $\|\mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\eta} \overline{\mathbf{u}_{\mathbf{\Lambda}}}\| = \|\mathbf{y}\|$.

Wir zerlegen nun $\mathbf{y} = \boldsymbol{\tau} + \boldsymbol{\nu}$ in einen Tangentialteil $\boldsymbol{\tau} \parallel \mathbf{w}$ und einen Normalteil $\boldsymbol{\nu} \perp \mathbf{w}$. Es folgt, dass $\mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\tau} \overline{\mathbf{u}_{\mathbf{\Lambda}}} = \boldsymbol{\tau}$. Die Binormale $\boldsymbol{\beta}$ ist über $\boldsymbol{\beta} = \boldsymbol{\tau} \times \boldsymbol{\nu}$ definiert. Zuletzt ist noch zu zeigen, dass $\mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\nu} \overline{\mathbf{u}_{\mathbf{\Lambda}}} = \cos \theta \boldsymbol{\nu} + \sin \theta \boldsymbol{\beta}$ eine Rotation von $\boldsymbol{\nu}$ um den Winkel θ beschreibt. \square

Lemma 2.16

Sei $\mathbf{u}_{\mathbf{\Lambda}} = u_0 + u_1 i + u_2 j + u_3 k \in \mathbb{H}$ eine Einheitsquaternion, dann ist die zugehörige Rotationsmatrix $\mathbf{\Lambda}$ gegeben durch

$$\mathbf{\Lambda} = \begin{pmatrix} 2(u_0^2 + u_1^2) - 1 & 2(u_1 u_2 - u_0 u_3) & 2(u_1 u_3 + u_0 u_2) \\ 2(u_1 u_2 + u_0 u_3) & 2(u_0^2 + u_2^2) - 1 & 2(u_2 u_3 - u_0 u_1) \\ 2(u_1 u_3 - u_0 u_2) & 2(u_2 u_3 + u_0 u_1) & 2(u_0^2 + u_3^2) - 1 \end{pmatrix}.$$

Zugehörig heißt hier, dass entsprechend Gleichung (2.11) die gleiche Rotation ausgeführt wird.

Beweis. Wir wählen ein beliebiges $\mathbf{x} \in \mathbb{R}^3$ und setzen sowohl $\mathbf{u}_{\mathbf{\Lambda}}$, als auch $\mathbf{\Lambda}$ in (2.11) ein.

$$\mathbf{\Lambda} \mathbf{x} = \begin{pmatrix} (2u_0^2 + 2u_1^2 - 1)x_1 + 2(u_1 u_2 - u_0 u_3)x_2 + 2(u_1 u_3 + u_0 u_2)x_3 \\ (2u_0^2 + 2u_2^2 - 1)x_2 + 2(u_2 u_3 - u_0 u_1)x_3 + 2(u_1 u_2 + u_0 u_3)x_1 \\ (2u_0^2 + 2u_3^2 - 1)x_3 + 2(u_1 u_3 - u_0 u_2)x_1 + 2(u_2 u_3 + u_0 u_1)x_2 \end{pmatrix}$$

$$\begin{aligned} \mathbf{u}_{\mathbf{\Lambda}} \boldsymbol{\eta} \overline{\mathbf{u}_{\mathbf{\Lambda}}} &= (u_0 + u_1 i + u_2 j + u_3 k)(x_1 i + x_2 j + x_3 k)(u_0 - u_1 i - u_2 j - u_3 k) \\ &= ((u_0^2 + u_1^2 - u_2^2 - u_3^2)x_1 + 2(u_1 u_2 - u_0 u_3)x_2 + 2(u_0 u_2 + u_1 u_3)x_3) i \\ &\quad + ((u_0^2 - u_1^2 + u_2^2 - u_3^2)x_2 + 2(u_2 u_3 - u_0 u_1)x_3 + 2(u_0 u_3 + u_1 u_2)x_1) j \\ &\quad + ((u_0^2 + u_1^2 - u_2^2 + u_3^2)x_3 + 2(u_1 u_3 - u_0 u_2)x_1 + 2(u_0 u_1 + u_2 u_3)x_2) k \end{aligned}$$

Da $\mathbf{u}_{\mathbf{\Lambda}}$ normiert ist, folgt die Aussage direkt. \square

2.5 Bézier-Kurven und Spline-Interpolation

Bézier-Kurven und Splines werden für die Ansatzfunktionen von Finiten Elementen höherer Ordnung benötigt. Mit rationalen Bézier-Kurven können Deformationen und insbesondere Geometrien exakt dargestellt werden, die durch Polynome nur unzureichend approximiert werden können. Kubische Splines ermöglichen es, stetig differenzierbare Funktionen durch Finite Elemente darzustellen. Die Theorie zu Bézier-Kurven ist [PT97] entnommen.

2.5.1 Bernstein-Polynome

Zuerst definieren wir die zu Grunde liegenden Bernstein-Polynome.

Definition 2.17 (Bernstein-Polynom):

Sei $t \in [0, 1]$ und $n \in \mathbb{N}_0$, dann ist das i -te Bernstein-Polynom $B_{i,n}$ der Ordnung n definiert durch

$$B_{i,n}(t) := \begin{cases} 0 & \text{falls } i < 0 \\ \frac{n!}{i!(n-i)!} t^i (1-t)^{n-i} & \text{falls } i = 0, \dots, n \\ 0 & \text{falls } i > n. \end{cases} \quad (2.12)$$

Satz 2.18 (Eigenschaften der Bernstein-Polynome):

Es gilt:

- Nichtnegativität:
 $B_{i,n} \geq 0 \quad n \in \mathbb{N}_0 \quad i = 0, \dots, n \quad t \in [0, 1]$
- Zerlegung der Einsfunktion:
 $\sum_{i=0}^n B_{i,n}(t) = 1 \quad t \in [0, 1]$
- Werte an den Endpunkten:
 $B_{0,n}(0) = B_{n,n}(1) = 1 \quad n \in \mathbb{N}_0$
- Rekursivität:
 $B_{i,n}(t) = (1-t)B_{i,n-1}(t) + tB_{i-1,n-1}(t) \quad n \in \mathbb{N}^+ \quad i = 0, \dots, n \quad t \in [0, 1]$
- Ableitungen:
 $\frac{\partial}{\partial t} B_{i,n}(t) = n(B_{i-1,n-1}(t) - B_{i,n-1}(t)) \quad n \in \mathbb{N}^+ \quad i = 0, \dots, n \quad t \in [0, 1]$

Beweis. Die Nichtnegativität und die Werte an den Endpunkten sind trivial, die Zerlegung der Einsfunktion folgt aus der binomischen Formel. Die Rekursivität und die Ableitungen folgen durch einfaches Nachrechnen. \square

Aus Gründen der Vollständigkeit definieren wir zuerst polynomiale Bézier-Kurven.

Definition 2.19 (Polynomiale Bézier-Kurve):

Sei $n \in \mathbb{N}_0$. Gegeben sind die Kontrollpunkte $\mathbf{P}_i \in \mathbb{R}^3$ ($i = 0, \dots, n$). Dann ist die polynomiale Bézier-Kurve $\mathbf{C} \subset \mathbb{R}^3$ vom Grad n gegeben durch

$$\mathbf{C}(t) := \sum_{i=0}^n B_{i,n}(t) \mathbf{P}_i \quad t \in [0, 1].$$

Analog hierzu können wir rationale Bézier-Kurven definieren.

Definition 2.20 (Rationale Bézier-Kurve):

Sei $n \in \mathbb{N}_0$. Gegeben sind die Kontrollpunkte $\mathbf{P}_i \in \mathbb{R}^3$ ($i = 0, \dots, n$) und die Gewichte $w_i > 0$ ($i = 0, \dots, n$). Dann ist die rationale Bézier-Kurve $\mathbf{C} \subset \mathbb{R}^3$ vom Grad n gegeben durch

$$\mathbf{C}(t) := \frac{\sum_{i=0}^n B_{i,n}(t) w_i \mathbf{P}_i}{\sum_{i=0}^n B_{i,n}(t) w_i} \quad t \in [0, 1].$$

Im Sonderfall, dass alle Gewichte w_i gleich sind, d.h. $w_0 = \dots = w_n$, entspricht die rationale Bézier-Kurve einer polynomialen Bézier-Kurve. Dies folgt durch die Eigenschaft der Zerlegung der Einsfunktion der Bernstein-Polynome.

Satz 2.21 (Eigenschaften rationaler Bézier-Kurven):

Es gilt:

- **Konvexe Hülle:**
Die rationale Bézier-Kurve liegt vollständig in der konvexen Hülle ihrer Kontrollpunkte \mathbf{P}_i .
- **Invarianz unter linearen Transformationen:**
Lineare Transformationen der rationalen Bézier-Kurve, wie Rotationen, Translationen und Skalierungen, hängen nur von den Transformationen der Kontrollpunkte ab.
- **Interpolation der Endpunkte:**
 $\mathbf{C}(0) = \mathbf{P}_0$ und $\mathbf{C}(1) = \mathbf{P}_n$
- **Ableitungen:**
Die k -te Ableitung an der Stelle $t = 0$ bzw. $t = 1$ hängt nur von den $k + 1$ ersten bzw. letzten Punkten und Gewichten ab. Insbesondere ist $\frac{\partial}{\partial t} \mathbf{C}(0)$ parallel zu $\mathbf{P}_1 - \mathbf{P}_0$ und $\frac{\partial}{\partial t} \mathbf{C}(1)$ parallel zu $\mathbf{P}_n - \mathbf{P}_{n-1}$.

Beweis. Die Eigenschaft der konvexen Hülle folgt aus der Nichtnegativität und der Zerlegung der Einsfunktion.

Da die rationale Bézier-Kurve eine Linearkombination der Kontrollpunkte darstellt, gilt die Invarianz unter linearen Transformationen.

Die Interpolation der Endpunkte folgt aus den Werten der Bernstein-Polynome an den Stellen $t = 0$ und $t = 1$.

Die Ableitungen an den Endpunkten können unter mehrmaliger Ausnutzung der Ableitungsregel und der Interpolation der Endpunkte für polynomiale Bézier-Kurven berechnet werden. Bei rationalen Bézier-Kurven kann der Beweis, unter Beachtung der Quotientenregel und mit Hilfe der Aussage für polynomiale Bézier-Kurven, analog geführt werden. \square

Die Parallelität der ersten Ableitung an den Endpunkten zur Geraden durch die beiden ersten bzw. letzten Punkte ist wichtig für die geeignete Wahl der Kontrollpunkte um eine C^1 -Kontinuität zumindest in der Referenzkonfiguration sicherzustellen.

2.5.2 Darstellung von Kreisen durch rationale Bézier-Kurven

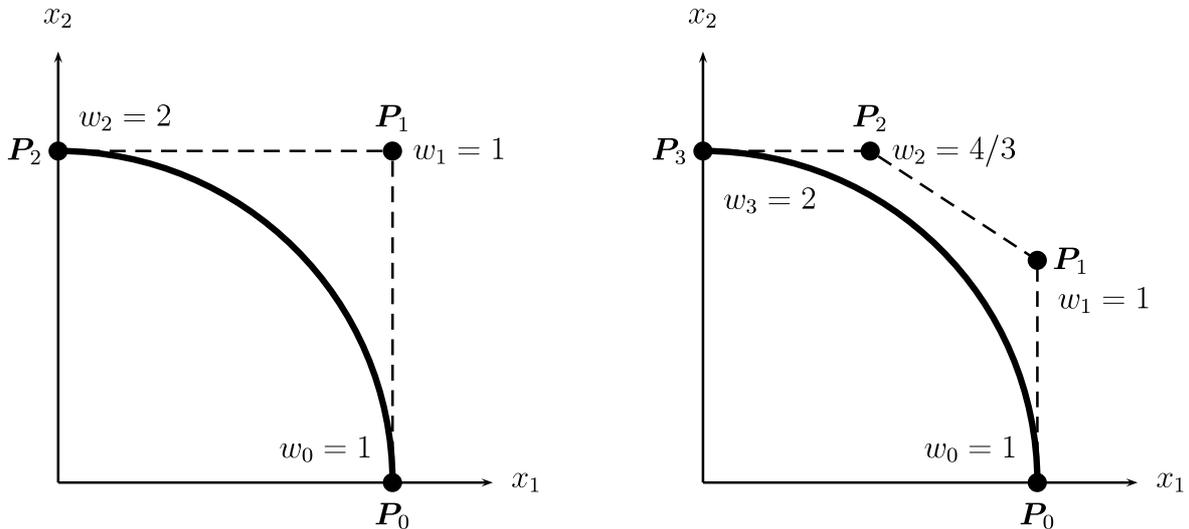


Abbildung 2.1: Darstellung eines Kreisbogens durch rationale Funktionen mit $n = 2$ (links) und $n = 3$ (rechts)

Ein Viertelbogen des Einheitskreises im \mathbb{R}^2 kann beispielsweise über den Winkel ψ parametrisiert werden.

$$x_1 = \cos(\psi), \quad x_2 = \sin(\psi) \quad \psi \in \left[0, \frac{\pi}{2}\right]$$

Eine alternative Parametrisierung des gleichen Kreisbogens ist durch die folgenden rationalen Funktionen möglich.

$$x_1 = \frac{1-t^2}{1+t^2}, \quad x_2 = \frac{2t}{1+t^2} \quad t \in [0, 1]$$

Da die Polynome im Nenner gleich sind, können wir den Kreisbogen durch eine rationale Bézier-Kurve mit mindestens 3 Kontrollpunkten P_i darstellen. Wir berechnen nun die Gewichte und die Lage der Kontrollpunkte bzgl. der Ordnungen $n = 2$ sowie $n = 3$.

Im Fall $n = 2$ ist die Lage der Kontrollpunkte leicht zu bestimmen. P_0 liegt am Bogenanfang, also bei $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$; P_2 am Bogenende, bei $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Der mittlere Kontrollpunkt P_1 muss nach Satz 2.21, Aussage 4, bei $\begin{pmatrix} 1 \\ 1 \end{pmatrix}$ liegen. Zur Berechnung der Gewichte nutzen wir die Nennerfunktion $1+t^2$. Diese muss als Linearkombination aus den Bernstein-Polynomen und den Gewichten dargestellt werden.

$$1+t^2 = \sum_{i=0}^2 B_{i,2}(t) w_i = (1-t)^2 w_0 + t(1-t) w_1 + t^2 w_2$$

Durch Einsetzen verschiedener Werte für t folgt

$$w_0 = 1, \quad w_1 = 1, \quad w_2 = 2.$$

Für $n = 3$ bestimmen wir zuerst die Gewichte, um die Nennerfunktion $1 + t^2$ darstellen zu können.

$$1 + t^2 = \sum_{i=0}^3 B_{i,3}(t) w_i = (1-t)^3 w_0 + t(1-t)^2 w_1 + t^2(1-t) w_2 + t^3 w_3$$

Durch Einsetzen verschiedener Werte für t folgt diesmal

$$w_0 = 1, \quad w_1 = 1, \quad w_2 = \frac{4}{3}, \quad w_3 = 2.$$

Die Kontrollpunkte haben folgende Koordinaten $\mathbf{P}_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $\mathbf{P}_3 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Durch Satz 2.21 folgt außerdem $\mathbf{P}_1 = \begin{pmatrix} 1 \\ \alpha \end{pmatrix}$, $\mathbf{P}_2 = \begin{pmatrix} \beta \\ 1 \end{pmatrix}$ mit $\alpha, \beta \in \mathbb{R}$. Um α und β zu bestimmen, müssen wir die beiden Zählerfunktionen der Parametrisierung $1 - t^2$ und $2t$ durch den Ausdruck im Zähler der Bézier-Kurve darstellen. Durch Einsetzen der schon bekannten Koordinaten der Kontrollpunkte folgt $\alpha = \frac{2}{3}$ und $\beta = \frac{1}{2}$.

Falls keine Viertelkreise, sondern z.B. Achtelkreise dargestellt werden sollen, muss die gesamte Rechnung erneut durchgeführt werden, da sich die Gewichte und natürlich auch die Kontrollpunkte ändern.

2.5.3 Spline Darstellung von C^1 -Kurven

Ziel dieses Abschnittes ist es, zu einer gegebenen Menge an Punkten \mathbf{P}_k , $k = 0, \dots, n$ eine C^1 -kontinuierliche Kurve zu finden, die stückweise polynomial ist und durch sämtliche Punkte verläuft. Dies ist beispielsweise durch kubische Splines erfüllt. Wir konstruieren im Folgenden kubische Splines unter Verwendung der oben definierten Bernstein-Polynome dritter Ordnung.

Wir betrachten den k -ten Abschnitt der Kurve \mathbf{C}_k zwischen den beiden aufeinander folgenden Punkten $\mathbf{P}_k, \mathbf{P}_{k+1} \in \mathbb{R}^3$. Die Kurve \mathbf{C}_k kann mittels Bernstein-Polynomen (2.12) und zwei zusätzlichen Kontrollpunkten $\mathbf{Q}_k, \mathbf{R}_k \in \mathbb{R}^3$ dargestellt werden als

$$\mathbf{C}_k(t) = B_{0,3}(t)\mathbf{P}_k + B_{1,3}(t)\mathbf{Q}_k + B_{2,3}(t)\mathbf{R}_k + B_{3,3}(t)\mathbf{P}_{k+1} \quad t \in [0, 1].$$

Diese zusätzlichen Kontrollpunkte \mathbf{Q}_k und \mathbf{R}_k sind jedoch nicht bekannt. Wir berechnen die Ableitungen \mathbf{M}_k an den Stellen $t = 0$ und $t = 1$, d.h. in den Punkten \mathbf{P}_k und \mathbf{P}_{k+1} .

$$\mathbf{M}_k := \frac{\partial}{\partial t} \mathbf{C}_k(0) = 3(\mathbf{Q}_k - \mathbf{P}_k) \quad \mathbf{M}_{k+1} := \frac{\partial}{\partial t} \mathbf{C}_k(1) = 3(\mathbf{P}_{k+1} - \mathbf{R}_k)$$

Die zusätzlichen Kontrollpunkte $\mathbf{Q}_k, \mathbf{R}_k$ können nun durch die Ableitungen \mathbf{M}_k und \mathbf{M}_{k+1} ersetzt werden.

$$\mathbf{C}_k(t) = B_{0,3}(t)\mathbf{P}_k + B_{1,3}(t) \left(\frac{1}{3}\mathbf{M}_k + \mathbf{P}_k \right) + B_{2,3}(t) \left(\mathbf{P}_{k+1} - \frac{1}{3}\mathbf{M}_{k+1} \right) + B_{3,3}(t)\mathbf{P}_{k+1}$$

Wenn wir nun am Punkt \mathbf{P}_k in den links und rechts angrenzenden Abschnitten \mathbf{C}_{k-1} und \mathbf{C}_k jeweils die gleiche Ableitung \mathbf{M}_k einsetzen, ist die C^1 -Kontinuität gewährleistet. \mathbf{M}_k selbst ist, im Gegensatz zu den Punkten \mathbf{P}_k , nicht gegeben, kann jedoch folgendermaßen approximiert werden.

$$\mathbf{M}_k := \frac{1}{2}(-\mathbf{P}_{k-1} + \mathbf{P}_{k+1})$$

Diese Möglichkeit der Approximation folgt zum Einen anschaulich, kann andererseits aber auch formal hergeleitet werden. Dabei interpolieren wir die drei Punkte P_{k-1} , P_k und P_{k+1} durch eine quadratisches Polynom und werten dessen Ableitung am mittleren Punkt P_k aus. Im Falle äquidistanter Punkte und konstanter Krümmung ist die Richtung der Ableitung exakt.

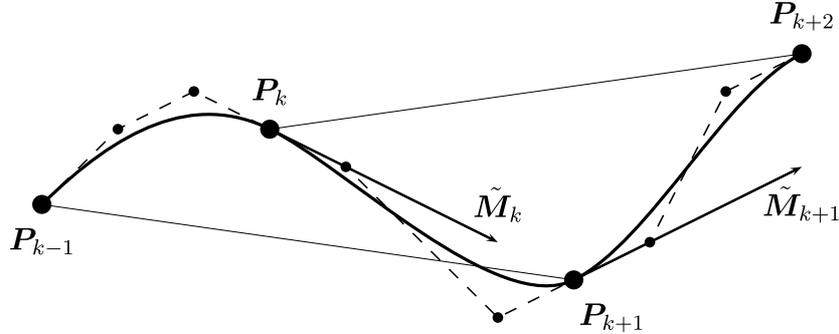


Abbildung 2.2: C^1 -Spline mit Approximation der Ableitungen

Die Kurve $C_k(t)$ hat nun die Form

$$C_k(t) = B_{0,3}(t)P_k + B_{1,3}(t) \left(-\frac{1}{6}P_{k-1} + P_k + \frac{1}{6}P_{k+1} \right) \\ + B_{2,3}(t) \left(\frac{1}{6}P_k + P_{k+1} - \frac{1}{6}P_{k+2} \right) + B_{3,3}(t)P_{k+1} \quad t \in [0, 1].$$

Im Falle $k = 0$ und $k = n - 1$ können wir allerdings obige Formel nicht anwenden, da P_{-1} und P_{n+1} nicht existieren. Um M_0 zu berechnen, interpolieren wir die drei Punkte P_0 , P_1 und P_2 wieder durch ein quadratisches Polynom und setzen M_0 gleich dessen Ableitung am Punkt P_0 . Analog verfahren wir mit den drei letzten Punkten P_{n-2} , P_{n-1} und P_n . Es folgt dann

$$M_0 := \frac{1}{2}(-3P_0 + 4P_1 - P_2) \quad M_n := \frac{1}{2}(P_{n-2} - 4P_{n-1} + 3P_n).$$

Die zugehörigen Kurven C_0 und C_{n-1} sind somit

$$C_0(t) = B_{0,3}(t)P_0 + B_{1,3}(t) \left(-\frac{1}{2}P_0 + \frac{3+2}{3}P_1 - \frac{1}{6}P_2 \right) \\ + B_{2,3}(t) \left(\frac{1}{6}P_0 + P_1 - \frac{1}{6}P_2 \right) + B_{3,3}(t)P_1 \\ C_{n-1}(t) = B_{0,3}(t)P_{n-1} + B_{1,3}(t) \left(-\frac{1}{6}P_{n-2} + P_{n-1} + \frac{1}{6}P_n \right) \\ + B_{2,3}(t) \left(-\frac{1}{6}P_{n-2} + \frac{3+2}{3}P_{n-1} - \frac{1}{2}P_n \right) + B_{3,3}(t)P_n.$$

Die Kurve $C := \bigcup_{k=0}^{n-1} C_k$ stellt nun den gewünschten C^1 -Spline dar.

Kapitel 3

Mechanisches Modell

In diesem Kapitel leiten wir das mechanische Modell der nichtlinearen Balkentheorie für endliche Rotationen her. Als kinematische Annahme dient die Hypothese nach Timoshenko, d.h. Querschnitte bleiben auch in der verformten Konfiguration eben, jedoch sind sie im Gegensatz zur Referenzkonfiguration nicht mehr senkrecht zur Balkenschwerachse. Als Grundlage dienen die Arbeiten von Ibrahimbegovic [Ibr95], Simo [Sim85] und Reissner [Rei72].

Im ersten Abschnitt definieren wir die zu Grunde liegende Konfiguration und erläutern die Hypothese nach Timoshenko. Danach werden die Gleichungen für das Gleichgewicht, die Kinematik und das Stoffgesetz hergeleitet bzw. vorgestellt. Am Ende dieses Kapitels wird das Variationsprinzip zur Lösung der Differentialgleichungen hergeleitet.

3.1 Konfiguration

Wir definieren ein globales Koordinatensystem durch die orthonormalen Basisvektoren \mathbf{e}_i ($i = 1, 2, 3$). Sei L_0 die Länge des Balkens in der Referenzkonfiguration. Der Balken wird durch

$$\Phi_0(s_0) = (\mathbf{u}_0(s_0), \mathbf{\Lambda}_0(s_0)) \quad s_0 \in [0, L_0]$$

vollständig beschrieben. Hierbei beschreibt die nach der Bogenlänge parametrisierte Kurve $\mathbf{u}_0(s_0)$ den Verlauf der Balkenschwerachse und die Rotationsmatrix $\mathbf{\Lambda}_0(s_0)$ die Verdrehung der Balkenquerschnitte.

Wir definieren in jedem Punkt $s_0 \in [0, L_0]$ das lokale Koordinatensystem \mathbf{g}_i durch $\mathbf{g}_i = \mathbf{\Lambda}_0 \mathbf{e}_i$ ($i = 1, 2, 3$). Der Vektor \mathbf{g}_1 steht dabei senkrecht auf der Balkenquerschnittsfläche, wohingegen \mathbf{g}_2 und \mathbf{g}_3 in der Querschnittsfläche liegen. Wir wählen die Hauptachsen der Flächenträgheitsmomente als Richtungen der Vektoren \mathbf{g}_2 und \mathbf{g}_3 . Dadurch erhalten wir später eine diagonalisierte Matrix der Materialparameter. Im Fall der Referenzkonfiguration gilt $\frac{\partial}{\partial s_0} \mathbf{u}_0 = \mathbf{g}_1$, d.h. die Balkenquerschnitte stehen senkrecht zur Schwerachse. Es treten in der Referenzkonfiguration keinerlei Verzerrungen bzw. Spannungen auf.

Die aktuelle Konfiguration des Balkens wird durch

$$\bar{\Phi}(s_0) = (\bar{\mathbf{u}}(s_0), \bar{\mathbf{\Lambda}}(s_0)) \quad s_0 \in [0, L_0]$$

beschrieben. Obwohl die Länge L des Balkens in der aktuellen Konfiguration im Allgemeinen ungleich L_0 ist, werden die Größen $\bar{\mathbf{u}}$ und $\bar{\mathbf{\Lambda}}$ weiter bzgl. s_0 parametrisiert. Es besteht also im Gegensatz zur Referenzkonfiguration keine Parametrisierung nach der Bogenlänge.

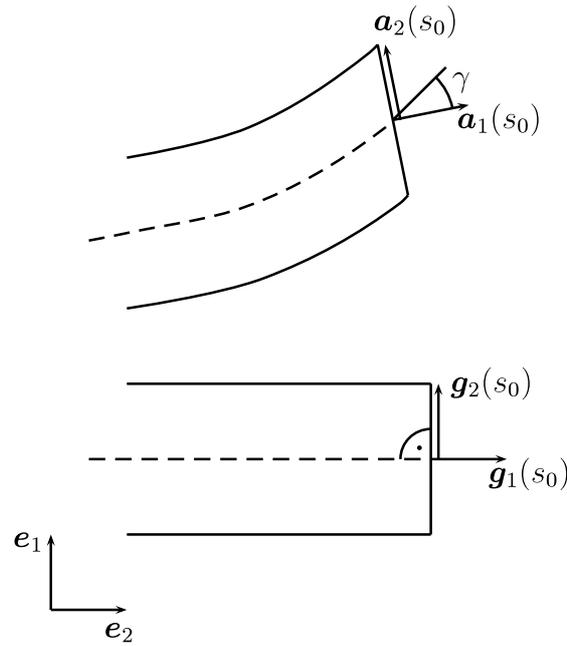


Abbildung 3.1: Kinematische Hypothese der Balkenquerschnitte nach Timoshenko, Referenzkonfiguration (unten) und aktuelle Konfiguration (oben)

Analog zur Referenzkonfiguration stellt $\bar{\mathbf{u}}(s_0)$ den Verlauf der Balkenschwerachse und die Rotationsmatrix $\bar{\mathbf{\Lambda}}(s_0)$ die Verdrehung der Balkenquerschnitte dar. Wir definieren außerdem das lokale Koordinatensystem \mathbf{a}_i durch $\mathbf{a}_i = \bar{\mathbf{\Lambda}}\mathbf{e}_i$ ($i = 1, 2, 3$). Der Vektor \mathbf{a}_1 steht dabei wieder senkrecht auf der Balkenquerschnittsfläche.

Die kinematische Hypothese nach Timoshenko besagt, dass die Balkenquerschnittsflächen auch in der aktuellen, deformierten Konfiguration eben bleiben. Dies ist in Abbildung 3.1 für den zweidimensionalen Fall dargestellt. Es treten somit keine Spannungen und Verzerrungen innerhalb der Querschnittsfläche auf. Im Gegensatz zur Hypothese nach Euler-Bernoulli stehen die Querschnittsflächen jedoch nicht länger senkrecht auf der Balkenschwerachse. Somit gilt im Allgemeinen

$$\frac{\partial}{\partial s_0}\bar{\mathbf{u}} \neq \mathbf{a}_1.$$

Abbildung 3.2 zeigt exemplarisch die Lage der globalen und lokalen Koordinatensysteme \mathbf{g}_i und \mathbf{a}_i .

Da der Balken in der Referenzkonfiguration unbelastet ist und somit z.B. Spannungen und Verzerrungen nur von der Differenz aus aktueller Konfiguration und Referenzkonfiguration abhängen, definieren wir die Verschiebungen \mathbf{u} und Verdrehungen $\mathbf{\Lambda}$ gesondert durch

$$\mathbf{u}_0 + \mathbf{u} = \bar{\mathbf{u}}, \quad \mathbf{\Lambda}\mathbf{\Lambda}_0 = \bar{\mathbf{\Lambda}}.$$

Hierbei ist zu beachten, dass bei Rotationen die Hintereinanderausführung nicht additiv, sondern multiplikativ dargestellt wird. Der Reihenfolge kommt wegen der fehlenden Kommutativität der Matrizenmultiplikation eine gesonderte Bedeutung zu. Zusätzliche Rotationen um ortsfeste Achsen werden dabei immer von links anmultipliziert, sodass $\mathbf{\Lambda}\mathbf{\Lambda}_0$ zuerst eine Rotation mit $\mathbf{\Lambda}_0$ vom globalen Koordinatensystem in die Referenzkonfiguration und danach eine Rotation mit $\mathbf{\Lambda}$ in die aktuelle Konfiguration darstellt.

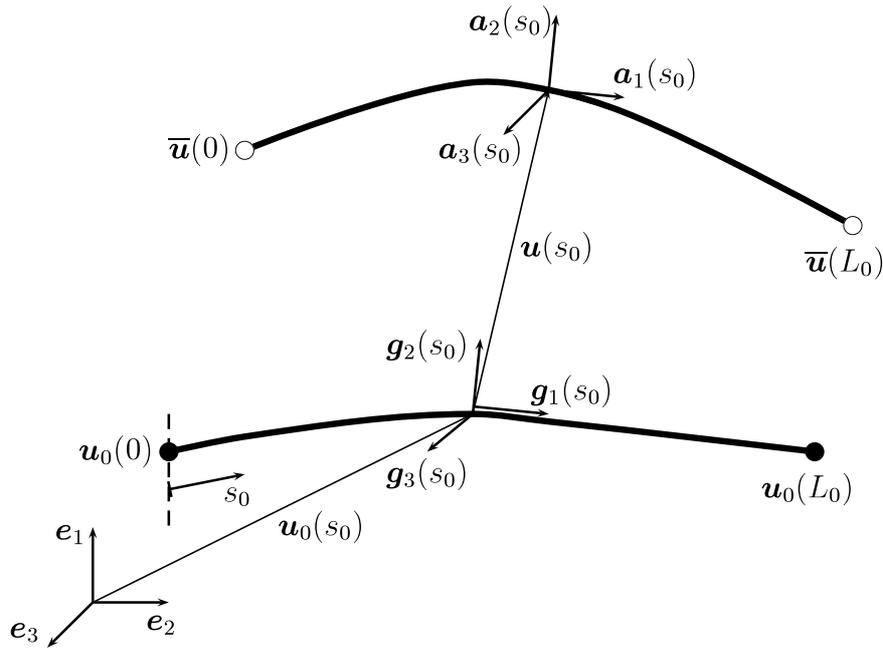


Abbildung 3.2: Kinematik des Balkens

Wir haben nun verschiedene Möglichkeiten die Vektoren des lokalen Koordinatensystems \mathbf{a}_i darzustellen.

$$\mathbf{a}_i = \bar{\Lambda} \mathbf{e}_i = \Lambda \Lambda_0 \mathbf{e}_i = \Lambda \mathbf{g}_i \quad i = 1, 2, 3$$

Analog zu den vorhergehenden Fällen definieren wir außerdem

$$\Phi(s_0) = (\mathbf{u}(s_0), \Lambda(s_0)) \quad s_0 \in [0, L_0].$$

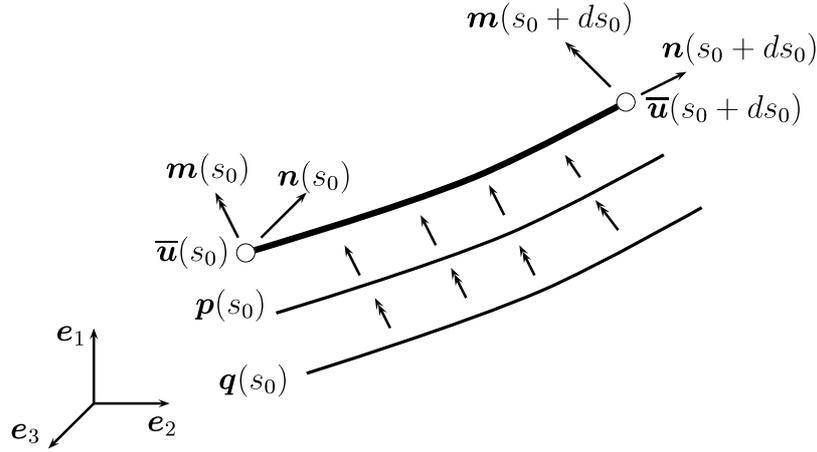
3.2 Gleichgewicht

Die Gleichgewichtsbedingungen werden am verformten Balken aufgestellt. Hierfür betrachten wir ein Balkenelement von $\mathbf{u}(s_0)$ nach $\bar{\mathbf{u}}(s_0 + ds_0)$ der differentiellen Länge $ds \neq ds_0$. Wegen $ds_0 > 0$ und da Balkenabschnitte endlicher Länge nicht durch endliche Belastung vollständig verschwinden können, folgt auch $ds > 0$.

Die Schnittkraft $\mathbf{n}(s_0)$ und das Schnittmoment $\mathbf{m}(s_0)$ sind Funktionen in s_0 . Sie werden auch als Spannungsresultierende bezeichnet, da sie aus der allgemeinen Kontinuumsmechanik mittels Aufintegration der Spannungen über der Querschnittsfläche entstehen.

Analog definieren wir die, an der Schwerlinie angreifende Streckenlast $\mathbf{p}(s_0)$ und das Streckenmoment $\mathbf{q}(s_0)$ aus den Belastungen, die in der Kontinuumsmechanik auftreten. In Abbildung 3.3 ist dies nochmals dargestellt. Dabei ist zu beachten, dass diese hier als Linienlasten in der Referenzkonfiguration Φ_0 definiert sind. Dies entspricht z.B. der Schwerkraft. Je nach Belastungsart, z.B. Wasserdruck oder Kontakt, ist diese jedoch zusätzlich von der aktuellen Konfiguration $\bar{\Phi}$ abhängig.

Wir nehmen weiter an, dass $\bar{\mathbf{u}}, \mathbf{n}, \mathbf{m} \in C^1[0, L_0]$ stetig differenzierbar sind.

Abbildung 3.3: Gleichgewicht am differentiellen Balkenelement der Länge ds

Aus der Kräfte-Gleichgewichts-Bedingung folgt

$$-\mathbf{n}(s_0) + \mathbf{n}(s_0 + ds_0) + \mathbf{p}(s_0) \frac{ds_0}{ds} ds = 0.$$

Da $ds_0 > 0$, können wir durch ds_0 teilen und erhalten

$$\frac{\mathbf{n}(s_0 + ds_0) - \mathbf{n}(s_0)}{ds_0} + \mathbf{p}(s_0) = 0.$$

Mit dem Grenzübergang $ds_0 \rightarrow 0^+$ folgt schließlich

$$\mathbf{n}'(s_0) + \mathbf{p}(s_0) = 0, \quad (3.1)$$

wobei $(\cdot)' = \frac{\partial}{\partial s_0}(\cdot)$.

Die Momenten-Gleichgewichts-Bedingung am Punkt $\bar{\mathbf{u}}(s_0)$ ergibt analog

$$\begin{aligned} & -\mathbf{m}(s_0) + \mathbf{m}(s_0 + ds_0) + \mathbf{n}(s_0 + ds_0) \times (\bar{\mathbf{u}}(s_0 + ds_0) - \bar{\mathbf{u}}(s_0)) \\ & + \mathbf{q}(s_0) \frac{ds_0}{ds} ds + \left(\mathbf{p}(s_0) \frac{ds_0}{ds} ds \right) \times \left(\frac{1}{2} (\bar{\mathbf{u}}(s_0 + ds_0) - \bar{\mathbf{u}}(s_0)) \right) = 0. \end{aligned}$$

Wir teilen wieder durch ds_0 und haben

$$\begin{aligned} \frac{\mathbf{m}(s_0 + ds_0) - \mathbf{m}(s_0)}{ds_0} + \mathbf{n}(s_0 + ds_0) \times \frac{\bar{\mathbf{u}}(s_0 + ds_0) - \bar{\mathbf{u}}(s_0)}{ds_0} \\ + \mathbf{q}(s_0) + \mathbf{p}(s_0) \times \frac{\bar{\mathbf{u}}(s_0 + ds_0) - \bar{\mathbf{u}}(s_0)}{2} = 0, \end{aligned}$$

mit $ds_0 \rightarrow 0^+$ folgt jetzt

$$\mathbf{m}'(s_0) + \mathbf{n}(s_0) \times \bar{\mathbf{u}}'(s_0) + \mathbf{q}(s_0) = 0. \quad (3.2)$$

Wenn die Gleichungen (3.1) und (3.2) für jedes $s_0 \in [0, L_0]$ erfüllt sind, so sagen wir, dass das Gleichgewicht in seiner starken Form gilt. Wir sprechen von einem statischen Gleichgewichtszustand.

3.3 Kinematik

Die Kinematik stellt die Beziehungen zwischen den äußeren Verschiebungen \mathbf{u} , $\mathbf{\Lambda}$ und den inneren Verzerrungen $\boldsymbol{\varepsilon}$, $\boldsymbol{\kappa}$ her. Dabei sind die Vektoren $\boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{pmatrix}$, $\boldsymbol{\kappa} = \begin{pmatrix} \kappa_1 \\ \kappa_2 \\ \kappa_3 \end{pmatrix} \in \mathbb{R}^3$ bzgl. dem globalen Koordinatensystem \mathbf{e}_i angegeben. Um die Anteile der Normal-, Schub-, Torsions- und Biegeverzerrung zu erhalten, müssen wir $\boldsymbol{\varepsilon}$ und $\boldsymbol{\kappa}$ zuerst bzgl. der lokalen Koordinaten \mathbf{a}_i berechnen.

$$\boldsymbol{\varepsilon}^{lokal} = \overline{\mathbf{\Lambda}}^T \boldsymbol{\varepsilon}, \quad \boldsymbol{\kappa}^{lokal} = \overline{\mathbf{\Lambda}}^T \boldsymbol{\kappa}$$

Die Normalverzerrung ist ε_1^{lokal} , die Schubverzerrungen sind ε_2^{lokal} und ε_3^{lokal} . Mit κ_1^{lokal} folgt die Torsionsverzerrung und κ_2^{lokal} sowie κ_3^{lokal} geben die Biegeverzerrungen an.

Satz 3.1 (Endliche Verzerrungen):

Für die endlichen Verzerrungen $\boldsymbol{\varepsilon}$ und $\boldsymbol{\kappa}$ gilt

$$\boldsymbol{\varepsilon} = \overline{\mathbf{u}}' - \mathbf{a}_1, \quad \boldsymbol{\kappa} = \boldsymbol{\omega}. \quad (3.3)$$

Hierbei ist $\boldsymbol{\omega}$ der axiale Vektor von $\widehat{\boldsymbol{\Omega}} := \left(\frac{\partial}{\partial s} \mathbf{\Lambda}\right) \mathbf{\Lambda}^T$.

Der Beweis ist in Anhang A.3 zu finden. Dabei werden auch Beziehungen für $\delta \tilde{\boldsymbol{\varepsilon}}$ und $\delta \tilde{\boldsymbol{\kappa}}$ genutzt, die erst beim Prinzip der virtuellen Verschiebungen in Kapitel 3.6 definiert werden.

3.4 Stoffgesetz

Wir beschränken uns hier auf linear elastisches Material. Dies entspricht dem Hook'schen Gesetz, dass Dehnung und Spannung sich zueinander proportional verhalten. Außerdem weisen die Materialparameter keinerlei Richtungsabhängigkeit aus.

Solches Material kann durch lediglich zwei Parameter beschrieben werden, einerseits dem Elastizitätsmodul E , andererseits dem Schubmodul G . Daneben ist auch eine Beschreibung über das Elastizitätsmodul E und die Querkontraktionszahl ν oder mittels der beiden Lamé-Konstanten λ und μ möglich. Die Umrechnung der Parameter kann durch folgende Beziehungen erfolgen.

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

Da wir nur Balken, d.h. linienförmig modellierte Körper betrachten, geht auch die Querschnittsgeometrie in die Parameter des Stoffgesetzes mit ein. Die Querschnittsfläche bezeichnen wir mit A , die Flächenträgheitsmomente in Richtung \mathbf{g}_2 und \mathbf{g}_3 mit I_2 bzw. I_3 . Da die lokalen Koordinaten \mathbf{g}_2 und \mathbf{g}_3 mit den Hauptachsen übereinstimmen, verschwindet das Deviationsmoment. Das polare Trägheitsmoment ist I_t . Zusätzlich ist noch zu beachten, dass die Schubspannungen nicht konstant über der Querschnittsfläche verteilt sind. Um dies zu berücksichtigen wird ein Schubkorrekturfaktor $\varkappa < 1$ und eine effektive Schubquerschnittsfläche A_s definiert. Es gilt $A_s = \varkappa A$. Für Kreisquerschnitte ist beispielsweise $\varkappa = \frac{9}{10}$, bei Rechteckquerschnitten gilt $\varkappa = \frac{5}{6}$. Nach der Hypothese von Timoshenko, vgl. Kapitel 3.1, bleiben die Balkenquerschnitte eben und behalten ihre Gestalt, sodass alle Querschnittsparameter auch im aktuellen Koordinatensystem \mathbf{a}_i gelten.

Jetzt können wir die Materialgesetze in den lokalen Koordinaten \mathbf{a}_i angeben. Für die Normalbelastung in Richtung \mathbf{a}_1 gilt $n_1 = \varepsilon_1^{lokal} EA$. Für Schubbelastungen in Richtung \mathbf{a}_2 und \mathbf{a}_3

ist analog $n_i = \epsilon_i^{lokal} GA_s = \epsilon_i^{lokal} GA_{\mathcal{K}}$ ($i = 1, 2$). Für eine Torsionsbelastung in Richtung \mathbf{a}_1 ist $m_1 = \kappa_1^{lokal} GI_t$ und schließlich gilt für eine Biegebelastung in Richtung \mathbf{a}_2 bzw. \mathbf{a}_3 die Gleichung $m_i = \kappa_i^{lokal} EI_i$ ($i = 2, 3$).

In Matrixschreibweise ergibt dies unter Berücksichtigung, dass $\boldsymbol{\varepsilon}$ und $\boldsymbol{\kappa}$ in globalen Koordinaten \mathbf{e}_i angegeben sind,

$$\mathbf{n} = \bar{\boldsymbol{\Lambda}} \mathbf{C}_n \bar{\boldsymbol{\Lambda}}^T \boldsymbol{\varepsilon} = \bar{\boldsymbol{\Lambda}} \begin{pmatrix} EA & 0 & 0 \\ 0 & GA_s & 0 \\ 0 & 0 & GA_s \end{pmatrix} \bar{\boldsymbol{\Lambda}}^T \boldsymbol{\varepsilon}$$

$$\mathbf{m} = \bar{\boldsymbol{\Lambda}} \mathbf{C}_m \bar{\boldsymbol{\Lambda}}^T \boldsymbol{\kappa} = \bar{\boldsymbol{\Lambda}} \begin{pmatrix} GI_t & 0 & 0 \\ 0 & EI_2 & 0 \\ 0 & 0 & EI_3 \end{pmatrix} \bar{\boldsymbol{\Lambda}}^T \boldsymbol{\kappa}.$$

3.5 Explizite Form der Balkengleichungen

Jetzt haben wir alle Beziehungen für das zu lösende Randwertproblem gefunden. Gesucht sind die beiden Funktionen $\mathbf{u} : [0, L_0] \rightarrow \mathbb{R}^3$ und $\boldsymbol{\Lambda} : [0, L_0] \rightarrow SO_3$, für die folgende Beziehungen gelten. Die Funktionen $\mathbf{u}_0, \mathbf{u}, \boldsymbol{\Lambda}_0$ und $\boldsymbol{\Lambda}$ sollen zweimal nach s_0 differenzierbar sein. Für $\boldsymbol{\Lambda}_0$ und $\boldsymbol{\Lambda}$ ist zu berücksichtigen, dass SO_3 eine Mannigfaltigkeit ist.

Konfiguration:	$\bar{\mathbf{u}} = \mathbf{u}_0 + \mathbf{u},$	$\bar{\boldsymbol{\Lambda}} = \boldsymbol{\Lambda} \boldsymbol{\Lambda}_0$
Gleichgewicht:	$\mathbf{0} = \mathbf{n}' + \mathbf{p},$	$\mathbf{0} = \mathbf{m}' + \mathbf{n} \times \bar{\mathbf{u}}' + \mathbf{q}$
Stoffgesetz:	$\mathbf{n} = \bar{\boldsymbol{\Lambda}} \mathbf{C}_n \bar{\boldsymbol{\Lambda}}^T \boldsymbol{\varepsilon},$	$\mathbf{m} = \bar{\boldsymbol{\Lambda}} \mathbf{C}_m \bar{\boldsymbol{\Lambda}}^T \boldsymbol{\kappa}$
Kinematik:	$\boldsymbol{\varepsilon} = \bar{\mathbf{u}}' - \mathbf{a}_1,$	$\boldsymbol{\kappa} = \boldsymbol{\omega} \quad \text{mit} \quad \widehat{\boldsymbol{\Omega}} = \boldsymbol{\Lambda}' \boldsymbol{\Lambda}^T$
Referenzgeometrie:	$\mathbf{u}_0 = \begin{pmatrix} u_1(s_0) \\ u_2(s_0) \\ u_3(s_0) \end{pmatrix}$	
Randbedingungen:		
Dirichlet: z.B.	$\mathbf{u}(0) = \mathbf{0},$	$\mathbf{u}(L_0) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix},$
	$\boldsymbol{\Lambda}(0) = \mathbf{I}_3,$	$\boldsymbol{\Lambda}(L_0) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$
Neumann: z.B.	$\mathbf{n}(0) = \mathbf{0},$	$\mathbf{n}(L_0) = \begin{pmatrix} 0 \\ 0 \\ F \end{pmatrix},$
	$\mathbf{m}(0) = \mathbf{0},$	$\mathbf{m}(L_0) = \begin{pmatrix} 0 \\ M \\ 0 \end{pmatrix}$
gemischt: z.B.	$\mathbf{u}(0) = \mathbf{0},$	$\mathbf{n}(L_0) = \begin{pmatrix} 0 \\ 0 \\ F \end{pmatrix},$
	$\boldsymbol{\Lambda}(0) = \mathbf{I}_3,$	$\mathbf{m}(L_0) = \begin{pmatrix} 0 \\ M \\ 0 \end{pmatrix}$

Die Frage nach Existenz und Eindeutigkeit betrachten wir an dieser Stelle nicht näher. Insbesondere ist z.B. bei Randbedingungen nach Neumann, wegen möglicher Starrkörperverschiebungen, keine Eindeutigkeit gegeben. Diese explizite Form bietet sich nicht zur Berechnung an. Stattdessen formulieren wir im nächsten Kapitel die Gleichgewichtsbedingungen in ihrer schwachen, variationellen Form.

3.6 Schwache Form des Gleichgewichts – Prinzip der virtuellen Verschiebungen

In diesem Abschnitt leiten wir das Prinzip der virtuellen Verschiebungen aus der schwachen Form des Gleichgewichts her.

Seien $\delta \mathbf{u}(s_0)$ und $\delta \mathbf{w}(s_0)$ zwei beliebige, stetig differenzierbare Testfunktionen. Diese werden auch als virtuelle Verschiebungen bezeichnet. Wir nehmen zur Vereinfachung an, dass Dirichlet-Randbedingungen gelten. Also ist $\delta \mathbf{u}(0) = \delta \mathbf{u}(L_0) = \delta \mathbf{w}(0) = \delta \mathbf{w}(L_0) = 0$. Auch der in Kapitel 2.3 definierte Ableitungsoperator wird mit δ bezeichnet. Dieser hat mit den Testfunktionen aus mathematischer Sicht jedoch nichts gemein. Wenn die Gleichgewichtsbedingungen (3.1) und (3.2) erfüllt sind, ist das folgende Integral ebenfalls gleich 0.

$$0 = \int_0^{L_0} \delta \mathbf{u}^T \underbrace{(\mathbf{n}' + \mathbf{p})}_{=0} + \delta \mathbf{w}^T \underbrace{(\mathbf{m}' + \mathbf{n} \times \bar{\mathbf{u}}' + \mathbf{q})}_{=0} ds_0$$

Durch partielle Integration und mit Lemma A.5 können wir das Integral umformen.

$$\begin{aligned} 0 &= \int_0^{L_0} \delta \mathbf{u}^T \underbrace{(\mathbf{n}' + \mathbf{p})}_{=0} + \delta \mathbf{w}^T \underbrace{(\mathbf{m}' + \mathbf{n} \times \bar{\mathbf{u}}' + \mathbf{q})}_{=0} ds_0 \\ &= \int_0^{L_0} \delta \mathbf{u}^T \mathbf{n}' + \delta \mathbf{w}^T (\mathbf{n} \times \bar{\mathbf{u}}') + \delta \mathbf{w}^T \mathbf{m}' ds_0 + \int_0^{L_0} \delta \mathbf{u}^T \mathbf{p} + \delta \mathbf{w}^T \mathbf{q} ds_0 \\ &= \int_0^{L_0} -\delta \mathbf{u}'^T \mathbf{n} - \delta \mathbf{w}^T (\bar{\mathbf{u}}' \times \mathbf{n}) - \delta \mathbf{w}'^T \mathbf{m} ds_0 + \underbrace{[\delta \mathbf{u}'^T \mathbf{n} + \delta \mathbf{w}'^T \mathbf{m}]_0^{L_0}}_{=0} + \int_0^{L_0} \delta \mathbf{u}^T \mathbf{p} + \delta \mathbf{w}^T \mathbf{q} ds_0 \\ &= - \int_0^{L_0} \delta \mathbf{u}'^T \mathbf{n} + (\delta \mathbf{w} \times \bar{\mathbf{u}}')^T \mathbf{n} + \delta \mathbf{w}'^T \mathbf{m} ds_0 + \int_0^{L_0} \delta \mathbf{u}^T \mathbf{p} + \delta \mathbf{w}^T \mathbf{q} ds_0 \\ &= - \underbrace{\int_0^{L_0} \delta \tilde{\boldsymbol{\varepsilon}}^T \mathbf{n} + \delta \tilde{\boldsymbol{\kappa}}^T \mathbf{m} ds_0}_{=: \delta \Pi_{int}} + \underbrace{\int_0^{L_0} \delta \mathbf{u}^T \mathbf{p} + \delta \mathbf{w}^T \mathbf{q} ds_0}_{=: \delta \Pi_{ext}} \end{aligned}$$

Dabei definieren wir die beiden Funktionen $\delta \tilde{\boldsymbol{\varepsilon}}$ und $\delta \tilde{\boldsymbol{\kappa}}$. Sie werden auch als virtuelle Verzerrungen bezeichnet. Die Tilde soll verdeutlichen, dass es sich dabei nicht um die Ableitungen der Verzerrungen $\boldsymbol{\varepsilon}$ bzw. $\boldsymbol{\kappa}$ handelt.

$$\delta \tilde{\boldsymbol{\varepsilon}} := \delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}', \quad \delta \tilde{\boldsymbol{\kappa}} := \delta \mathbf{w}' \quad (3.4)$$

Die oben hergeleitete Gleichung $\delta \Pi_{int} = \delta \Pi_{ext}$ wird als Prinzip der virtuellen Verschiebungen bezeichnet.

$$\delta \Pi_{int} = \delta \Pi_{ext} \quad \Longleftrightarrow \quad \int_0^{L_0} \delta \tilde{\boldsymbol{\varepsilon}}^T \mathbf{n} + \delta \tilde{\boldsymbol{\kappa}}^T \mathbf{m} ds_0 = \int_0^{L_0} \delta \mathbf{u}^T \mathbf{p} + \delta \mathbf{w}^T \mathbf{q} ds_0 \quad (3.5)$$

Wenn sich ein System im statischen Gleichgewicht befindet, dann entspricht die, durch innere virtuelle Verzerrungen und Schnittkräfte gegebene, innere Arbeit $\delta \Pi_{int}$ der, durch virtuelle Verschiebungen und äußere Lasten aufgebrauchte, äußeren Arbeit $\delta \Pi_{ext}$.

Wenn das Prinzip der virtuellen Verschiebungen für alle Testfunktionen $\delta \mathbf{u}$, $\delta \mathbf{w}$ aus dem Sobolev-Raum H_0^1 erfüllt ist, gelten auch die Gleichgewichtsbedingungen in ihrer starken Form. Eine genaue Definition der Sobolev-Räume ist bei Braess [Bra03], Kapitel II, gegeben.

3.7 Prinzip des minimalen Energiepotentials

Das Prinzip des minimalen Energiepotentials sagt aus, dass in einer Gleichgewichtslage die im System enthaltene Energie minimiert wird. Die Energie setzt sich hier aus der Formänderungsarbeit sowie dem Potential der äußeren Lasten zusammen. Wir verweisen wiederum auf [Bra03], Kapitel IV. Folglich muss zusätzliche Energie aufgewandt werden, um das System leicht auszulasten. Das Energiepotential Π in der Konfiguration $\bar{\Phi}$ berechnet sich durch

$$\Pi(\bar{\Phi}) := \underbrace{\frac{1}{2} \int_0^{L_0} \boldsymbol{\varepsilon}^T \mathbf{n} + \boldsymbol{\kappa}^T \mathbf{m} \, ds_0}_{=\Pi_{int}} - \Pi_{ext}, \quad (3.6)$$

wobei L_0 , $\boldsymbol{\varepsilon}$, $\boldsymbol{\kappa}$, \mathbf{n} und \mathbf{m} wie oben definiert sind.

Für das innere Potential gilt

$$\Pi_{int} = \frac{1}{2} \int_0^{L_0} \boldsymbol{\varepsilon}^T \mathbf{n} + \boldsymbol{\kappa}^T \mathbf{m} \, ds_0 = \frac{1}{2} \int_0^{L_0} \boldsymbol{\varepsilon}^T \bar{\Lambda} C_n \bar{\Lambda}^T \boldsymbol{\varepsilon} + \boldsymbol{\kappa}^T \bar{\Lambda} C_m \bar{\Lambda}^T \boldsymbol{\kappa} \, ds_0.$$

Zur Berechnung des Minimums des Potentials soll die Bedingung des Verschwindens der ersten Ableitung bzw. der ersten Variation zu Hilfe genommen werden. Als Ableitungsbegriff wird hierfür die kovariante Ableitung, eine Verallgemeinerung der Richtungsableitung, benutzt, die in Kapitel 2.3 auch für Mannigfaltigkeiten definiert wurde.

Satz 3.2 (Erste Potentialvariation):

Die erste Variation bzw. die erste kovariante Ableitung des inneren Potentials in (3.6) in die Richtung $\delta\Phi := (\delta\mathbf{u}, \delta\Lambda)$ hat die Form

$$\delta\Pi_{int}(\bar{\Phi}, \delta\Phi) = \int_0^{L_0} \delta\tilde{\boldsymbol{\varepsilon}}^T \mathbf{n} + \delta\tilde{\boldsymbol{\kappa}}^T \mathbf{m} \, ds_0. \quad (3.7)$$

Dabei sind $\delta\tilde{\boldsymbol{\varepsilon}}$ und $\delta\tilde{\boldsymbol{\kappa}}$ die in (3.4) definierten virtuellen Verzerrungen. Der Beweis ist in Anhang A.4 zu finden.

Wenn wir die erste Variation des inneren Potentials $\delta\Pi_{int}$ mit der inneren Arbeit des Prinzips der virtuellen Verschiebungen in Gleichung (3.5) vergleichen, erkennen wir, dass beide Ausdrücke gleich sind. Die erste Variation des äußeren Potentials $\delta\Pi_{ext}$ setzen wir deshalb gleich der äußeren Arbeit des Prinzips der virtuellen Verschiebungen.

Um das Minimum des Energiepotentials zu berechnen, muss die erste Ableitung nullgesetzt werden. Dies entspricht dem Prinzip der virtuellen Verschiebungen.

Wir haben somit folgendes nichtlineares, kontinuierliches Problem zu lösen.

$$\text{Finde } \bar{\Phi} \text{ mit} \quad \delta\Pi(\bar{\Phi}, \delta\Phi) \stackrel{!}{=} 0 \quad \forall \delta\Phi \quad (3.8)$$

Dies ist im Allgemeinen jedoch nicht explizit möglich, deshalb kommt das Newton-Verfahren zur Anwendung, wie es z.B. bei Deuffhard [DB02] in Kapitel 4.2 dargestellt ist. Um das Newton-Verfahren anwenden zu können, wird die zweite Variation des Potentials benötigt. Entgegen der Definition bei Meyberg [MV03], Kapitel 13, bilden wir hierfür nicht die zweite Ableitung nach dem Variationsparameter α , sondern variieren allgemeiner die Ausgangskonfiguration $\bar{\Phi}$ ein weiteres Mal in eine neue Richtung $\Delta\Phi := (\Delta\mathbf{u}, \Delta\Lambda)$ und mit neuem Variationsparameter β . Näheres hierzu kann in Kapitel 2.3 nachgeschlagen werden.

Da $\Delta\mathbf{u}$ und $\Delta\mathbf{w}$ wieder als infinitesimal klein angenommen werden, gelten alle kinematischen Beziehungen, die wir für $\delta\mathbf{u}$ und $\delta\mathbf{w}$ nachgewiesen haben, auch hier.

Satz 3.3 (Zweite Potentialvariation):

Die zweite Variation des in Gleichung (3.6) definierten Potentials bzgl. der Richtungen $\delta\Phi := (\delta\mathbf{u}, \delta\Lambda)$ sowie $\Delta\Phi := (\Delta\mathbf{u}, \Delta\Lambda)$ hat die Form

$$\begin{aligned} \Delta\delta\Pi(\bar{\Phi}, \delta\Phi, \Delta\Phi) &= \Delta\delta\Pi_{int} = \int_0^{L_0} \delta\tilde{\varepsilon}^T \bar{\Lambda} C_n \bar{\Lambda}^T \Delta\tilde{\varepsilon} + \delta\tilde{\kappa}^T \bar{\Lambda} C_m \bar{\Lambda}^T \Delta\tilde{\kappa} ds_0 \\ &+ \int_0^{L_0} (\Delta\mathbf{w} \times (\delta\mathbf{w} \times \bar{\mathbf{u}}') + \delta\mathbf{u}' \times \Delta\mathbf{w} + \Delta\mathbf{u}' \times \delta\mathbf{w})^T \mathbf{n} + (\delta\mathbf{w}' \times \Delta\mathbf{w})^T \mathbf{m} ds_0. \end{aligned} \quad (3.9)$$

Dabei wird das erste Integral oft als materieller Teil und das zweite Integral als geometrischer Teil bezeichnet. In vielen Anwendungen kann der geometrische Teil weggelassen werden ohne die Konvergenz des Newton-Verfahrens zu verlieren, allerdings verschlechtert sich dabei die Konvergenzgeschwindigkeit.

Der Beweis ist im Anhang A.5 zu finden.

Bemerkung 3.4

In den Arbeiten von Ibrahimbegovic [Ibr95] und Simo [Sim86] ist der geometrische Teil der zweiten Variation in anderer Form angegeben. Dabei unterscheiden sich auch die beiden Arbeiten auch untereinander. Ursache ist wohl ein unterschiedlich definierter Begriff der zweiten kovarianten Ableitung, bzw. eine spezielle Metrik in der Menge der Rotationen SO_3 . In [Sim92] wird näher auf diese Aspekte der kovarianten Ableitung eingegangen. Später vergleichen wir die Ergebnisse der drei unterschiedlichen geometrischen Teile der zweiten Variation, sowie der Berechnung nur mit materiellem Teil.

Das folgende Lemma dient als Grundlage dafür, dass das später in jedem Newton-Schritt zu lösende Gleichungssystem linear ist.

Lemma 3.5 (Linear- und Bilinearform der Variationen):

Es gilt:

- Die erste Variation $\delta\Pi(\bar{\Phi}, \delta\Phi)$ ist linear in $\delta\Phi$. Sie kann als Linearform bzgl. $\begin{pmatrix} \delta\mathbf{u} \\ \delta\mathbf{w} \end{pmatrix}$ und der Ableitungen $\begin{pmatrix} \delta\mathbf{u}' \\ \delta\mathbf{w}' \end{pmatrix}$ dargestellt werden.
- Die zweite Variation $\Delta\delta\Pi(\bar{\Phi}, \delta\Phi, \Delta\Phi)$ ist linear in $\delta\Phi$ und $\Delta\Phi$ und somit eine Bilinearform bzgl. $\begin{pmatrix} \delta\mathbf{u} \\ \delta\mathbf{w} \end{pmatrix}$ und $\begin{pmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{w} \end{pmatrix}$.

Beweis. Die erste Variation können wir wie folgt umformen.

$$\begin{aligned} \delta\Pi(\bar{\Phi}, \delta\Phi) &= \int_0^{L_0} \delta\tilde{\varepsilon}^T \mathbf{n} + \delta\tilde{\kappa}^T \mathbf{m} - \delta\mathbf{u}^T \mathbf{p} - \delta\mathbf{w}^T \mathbf{q} ds_0 \\ &= \int_0^{L_0} -\delta\mathbf{u}^T \mathbf{p} + \delta\mathbf{u}'^T \mathbf{n} - \delta\mathbf{w}^T \left(\mathbf{q} + \widehat{\mathbf{U}}' \mathbf{n} \right) - \delta\mathbf{w}'^T \mathbf{m} ds_0 \\ &= \int_0^{L_0} \begin{pmatrix} \delta\mathbf{u} \\ \delta\mathbf{u}' \\ \delta\mathbf{w} \\ \delta\mathbf{w}' \end{pmatrix}^T \begin{pmatrix} -\mathbf{p} \\ \mathbf{n} \\ -\mathbf{q} - \widehat{\mathbf{U}}' \mathbf{n} \\ \mathbf{m} \end{pmatrix} ds_0 \end{aligned}$$

Die Linearität in $\begin{pmatrix} \delta\mathbf{u} \\ \delta\mathbf{w} \end{pmatrix}$ ist wegen der Linearität der Ableitungen nun ersichtlich. Genauso folgt die Linearität bzgl. $\delta\Phi = \begin{pmatrix} \delta\mathbf{u} \\ \delta\widehat{\mathbf{W}}\Lambda \end{pmatrix}$.

Durch analoges Vorgehen können wir auch die zweite Variation umschreiben.

$$\begin{aligned}
& \Delta \delta \Pi (\bar{\Phi}, \delta \Phi, \Delta \Phi) \\
&= \int_0^{L_0} \delta \tilde{\varepsilon}^T \bar{\Lambda} C_n \bar{\Lambda}^T \Delta \tilde{\varepsilon} + \delta \tilde{\kappa}^T \bar{\Lambda} C_m \bar{\Lambda}^T \Delta \tilde{\kappa} \\
&\quad + (\Delta \mathbf{w} \times (\delta \mathbf{w} \times \bar{\mathbf{u}}') + \delta \mathbf{u}' \times \Delta \mathbf{w} + \Delta \mathbf{u}' \times \delta \mathbf{w})^T \mathbf{n} + (\delta \mathbf{w}' \times \Delta \mathbf{w})^T \mathbf{m} \, ds_0 \\
&= \int_0^{L_0} \delta \mathbf{u}'^T (\bar{\Lambda} C_n \bar{\Lambda}^T) \Delta \mathbf{u}' + \delta \mathbf{u}'^T (\bar{\Lambda} C_n \bar{\Lambda}^T \widehat{\mathbf{U}}') \Delta \mathbf{w} + \delta \mathbf{w}^T (\widehat{\mathbf{U}}'^T \bar{\Lambda} C_n \bar{\Lambda}^T) \Delta \mathbf{u}' \\
&\quad + \delta \mathbf{w}'^T (\widehat{\mathbf{U}}'^T \bar{\Lambda} C_n \bar{\Lambda}^T \widehat{\mathbf{U}}') \Delta \mathbf{w} + \delta \mathbf{w}'^T (\bar{\Lambda} C_m \bar{\Lambda}^T) \Delta \mathbf{w}' + \delta \mathbf{w}^T (\widehat{\mathbf{U}}' \widehat{\mathbf{N}}) \Delta \mathbf{w} \\
&\quad + \delta \mathbf{u}'^T (\widehat{\mathbf{N}}^T) \Delta \mathbf{w} + \delta \mathbf{w}^T (\widehat{\mathbf{N}}) \Delta \mathbf{u}' + \delta \mathbf{w}'^T (\widehat{\mathbf{M}}) \Delta \mathbf{w} \, ds_0 \\
&= \int_0^{L_0} \begin{pmatrix} \delta \mathbf{u} \\ \delta \mathbf{u}' \\ \delta \mathbf{w} \\ \delta \mathbf{w}' \end{pmatrix}^T \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \bar{\Lambda} C_n \bar{\Lambda}^T & \widehat{\mathbf{U}}'^T \bar{\Lambda} C_n \bar{\Lambda}^T + \widehat{\mathbf{N}} & 0 \\ 0 & \bar{\Lambda} C_n \bar{\Lambda}^T \widehat{\mathbf{U}}' + \widehat{\mathbf{N}}^T & \widehat{\mathbf{U}}'^T \bar{\Lambda} C_n \bar{\Lambda}^T \widehat{\mathbf{U}}' + \widehat{\mathbf{U}}' \widehat{\mathbf{N}} & \widehat{\mathbf{M}}^T \\ 0 & 0 & 0 & \bar{\Lambda} C_m \bar{\Lambda}^T \end{pmatrix} \begin{pmatrix} \Delta \mathbf{u} \\ \Delta \mathbf{u}' \\ \Delta \mathbf{w} \\ \Delta \mathbf{w}' \end{pmatrix} ds_0
\end{aligned}$$

Somit erhalten wir eine Bilinearform in $\begin{pmatrix} \delta \mathbf{u} \\ \delta \mathbf{w} \end{pmatrix}$ und $\begin{pmatrix} \Delta \mathbf{u} \\ \Delta \mathbf{w} \end{pmatrix}$ bzw. $\begin{pmatrix} \delta \mathbf{u} \\ \delta \widehat{\mathbf{W}}_\Lambda \end{pmatrix}$ und $\begin{pmatrix} \Delta \mathbf{u} \\ \Delta \widehat{\mathbf{W}}_\Lambda \end{pmatrix}$ □

Bemerkung 3.6

Die Linearität von $\delta \mathbf{\Pi}$ und die Bilinearität von $\Delta \delta \mathbf{\Pi}$ sind selbstverständlich, wenn ein Vektorraum zugrunde liegt. Da Λ jedoch in SO_3 liegt, kann die Linearität im Allgemeinen nur für infinitesimal kleine Variationen gelten. Bei der späteren numerischen Berechnung ist deshalb immer zu beachten, dass $\delta \Phi$ und $\Delta \Phi$ dies auch zumindest annähernd erfüllen.

Kapitel 4

Finite Elemente Methode

In diesem Kapitel betrachten wir Methoden zur approximativen Lösung des mechanischen Problems aus Kapitel 3. Um das Problem (3.8) zu diskretisieren, nutzen wir die Methode der Finiten Elemente. Außerdem muss es linearisiert werden. Hierfür wenden wir die Newton-Iteration an. Dabei sind die Nebenbedingungen durch Kontakt der Balken untereinander zu berücksichtigen.

4.1 Allgemeines

Zur Berechnung der Nullstelle in Gleichung (3.8) wenden wir die Newton-Iteration an. Für die Berechnung eines Iterationsschrittes, benötigen wir die erste, als auch die zweite Variation des Energiepotentials Π .

Im vorhergehenden Kapitel wurde gezeigt, dass die Variationen $\delta\Pi$ und $\Delta\delta\Pi$ linear in $\begin{pmatrix} \delta\mathbf{u} \\ \delta\mathbf{w} \end{pmatrix}$ bzw. bilinear in $\begin{pmatrix} \delta\mathbf{u} \\ \delta\mathbf{w} \end{pmatrix}$ und $\begin{pmatrix} \Delta\mathbf{u} \\ \Delta\mathbf{w} \end{pmatrix}$ sind. Unser Ziel ist es, beide Variationen als einfaches Matrix-Vektor-Produkt darzustellen. Hierbei stört jedoch die Integration bzgl. s_0 . Im Zuge dieses Kapitels werden wir sehen, dass dieses Problem durch die Diskretisierung mit der Finiten Elemente Methode umgangen werden kann. Dazu werden die Linearisierung durch die Newton-Iteration und die Diskretisierung mittels der Finiten Elemente Methode parallel hergeleitet. Ein rein sequentieller Aufbau des folgenden Kapitels ist daher nicht möglich. So werden z.B. einige Bemerkungen in Abschnitt 4.2 erst durch Abschnitt 4.3 ihren vollen Stellenwert erhalten.

4.2 Finite Elemente Diskretisierung

Um das kontinuierliche Problem (3.8) in eine rechnerisch lösbare Form zu bringen, muss es diskretisiert werden. Dies bedeutet dass die unbekannt Funktionen \mathbf{u} und $\mathbf{\Lambda}$ durch eine endliche Zahl an Parametern beschrieben werden. Analog verfahren wir mit den Testfunktionen $\delta\mathbf{u}$ und $\delta\mathbf{w}$ sowie den inkrementellen Deformationen $\Delta\mathbf{u}$ und $\Delta\mathbf{w}$. Aus mathematischer Sicht wählen wir aus dem Sobolev-Raum $H_0^1[0, L_0]$ einen endlichdimensionalen Unterraum, in welchem unsere Test- und Lösungsfunktionen liegen müssen. Diese Methode zur Diskretisierung wird Galerkin-Verfahren genannt.

Die unbekannt Funktionen, beispielsweise \mathbf{u} , können als endliche Linearkombination aus Ansatzfunktionen A und Knotenwerten \mathbf{u}^e dargestellt werden. Die Ansatzfunktionen sind frei wählbar, sie sollten jedoch einige Eigenschaften aufweisen, um später den Aufwand für die Berechnung gering zu halten.

- Die Ansatzfunktionen sollten leicht zu differenzieren und integrieren sein. Wenn die Integration durch eine Quadraturformel berechnet wird, sollte diese mit möglichst wenigen Integrationspunkten eine brauchbare Approximation liefern können.
- Die Ansatzfunktionen sollten einen kleinen Support besitzen. Dadurch wird das spätere lineare Gleichungssystem möglichst dünnbesetzt und effizienter lösbar.
- Die Matrix des linearen Gleichungssystems sollte möglichst gut konditioniert sein.

Eine Möglichkeit zur Definition der Ansatzfunktionen A sind stückweise polynomiale Funktionen. Sie werden auch Hutfunktionen genannt, vgl. Abbildung 4.1. Das Intervall $[0, L_0]$ wird in kleine Abschnitte, die Finiten Elemente L_e , zerlegt. Jede Ansatzfunktion ist auf dem gesamten Intervall, mit Ausnahme zweier aufeinander folgende Elemente, identisch null. Die Ansatzfunktion steigt im linken Element linear vom Wert 0 am linken Ende bis zum Wert 1 am rechten Ende an. Im rechten Element fällt sie wieder linear auf den Wert 0 ab.

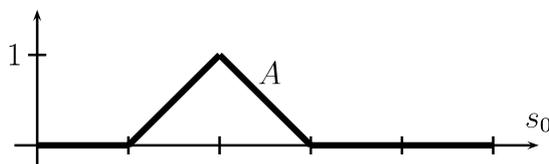


Abbildung 4.1: lineare Hutfunktion

Wir können die unbekannte Funktion \mathbf{u} nun durch eine Linearkombination aus Ansatzfunktionen und Knotenwerten approximieren. Somit ist nur noch eine endliche Zahl an unbekannt Parametern zu bestimmen. Diese können im Allgemeinen durch ein Gleichungssystem ermittelt werden.

Bei Ansatzfunktionen, die durch Polynome höherer Ordnung definiert sind, erstreckt sich der Support oft über mehr als drei Knoten. Die Vorteile, die durch eine genauere Approximation der zu Grunde liegenden Funktion \mathbf{u} entstehen, müssen dabei mit dem entsprechend höheren Rechenaufwand verglichen werden. In dieser Arbeit betrachten wir Ansatzfunktionen, die aus Polynomen bis maximal kubischer Ordnung oder aus rationalen Funktionen bestehen.

Wir definieren ein Referenzelement über dem Intervall $\xi \in [-1, 1]$. Die Abbildung Γ bildet jedes Element auf dieses Referenzelement ab. Wir differenzieren bzw. integrieren nur noch bzgl. der lokalen Variablen ξ . Da die Ableitungen bzw. die Integrationen im mechanischen Modell jedoch immer bzgl. s_0 ausgeführt werden, muss nach der Kettenregel zusätzlich noch mit der Ableitung von Γ multipliziert bzw. dividiert werden. Die Ableitung wird als Jacobimatrix $J = \frac{\partial s_0}{\partial \xi}$ bezeichnet. Um eine möglichst gut konditioniertes Gleichungssystem zu erhalten, sollte die Jacobimatrix J konstant sein. Deshalb werden die Knoten im Allgemeinen äquidistant aufgetragen.

Viele Berechnungen finden nur noch innerhalb der Elemente statt, mit den Knotenwerten als Variablen. Am Ende wird ein globales lineares Gleichungssystem aufgestellt und gelöst, wodurch die unbekannt Knotenwerte ermittelt werden können.

Um die Zahl der Funktionsauswertungen und den benötigten Speicherplatz minimal zu halten, approximieren wir alle Integrale durch Gauß-Quadraturen. Viele Funktionen, z.B. die Funktion der Rotationsmatrizen $\mathbf{\Lambda}(s_0)$, müssen dadurch nur noch punktweise an den Gauß'schen Integrationspunkten *gip* ermittelt und gespeichert werden.

4.2.1 Die Ansatzfunktionen \mathbf{A} , \mathbf{A}_{ref}

Sei nen (number of element nodes) die Zahl der linear unabhängigen Ansatzfunktionen. Die Ansatzfunktionen sind auf dem Referenzintervall $[-1, 1]$ definiert. Wir definieren Ansatzfunktionen \mathbf{A} zur Approximation der Deformationen (Verschiebungen und Verdrehungen). Ansatzfunktionen zur Approximation der Referenzkonfiguration (Geometrie) werden mit \mathbf{A}_{ref} bezeichnet.

Wir approximieren folgende Funktionen, jeweils eingeschränkt auf ein Element L_e

$$\mathbf{u} \Big|_{L_e} \approx \sum_{i=1}^{nen} A^{(i)} \mathbf{u}^{(i)} = \mathbf{A} \mathbf{u}^e \quad (4.1)$$

$$\delta \mathbf{u} \Big|_{L_e} \approx \sum_{i=1}^{nen} A^{(i)} \delta \mathbf{u}^{(i)} = \mathbf{A} \delta \mathbf{u}^e \quad (4.2)$$

$$\Delta \mathbf{u} \Big|_{L_e} \approx \sum_{i=1}^{nen} A^{(i)} \Delta \mathbf{u}^{(i)} = \mathbf{A} \Delta \mathbf{u}^e. \quad (4.3)$$

Da SO_3 kein Vektorraum ist, kann $\mathbf{\Lambda}$ nicht auf gleiche Weise durch eine Linearkombination approximiert werden. Genauso wenig kann der Rotationsvektor \mathbf{w} dargestellt werden, da die Linearität nur bei infinitesimalen Rotationen gilt.

Nach Definition ist $\delta \mathbf{w}$ eine infinitesimal kleine Rotation und somit durch eine Linearkombination approximierbar. Daneben ist $\Delta \mathbf{w}$ die in einem Newton-Schritt hinzukommende inkrementelle Rotation. $\Delta \mathbf{w}$ ist nicht infinitesimal klein, jedoch können die Werte, die $\Delta \mathbf{w}$ annimmt, durch schrittweises Aufbringen der Belastung beliebig klein bleiben. Die Annahme, dass $\Delta \mathbf{w}$ klein ist und deshalb durch eine Linearkombination zwischen den Knotenwerten interpoliert werden kann, ist eine sehr weitreichende Einschränkung der hier vorgestellten Berechnungsmethode. In den folgenden Kapiteln werden wir deshalb den Einfluss der Schrittgröße beim Aufbringen der Belastung näher untersuchen. Es ergeben sich folgende Approximationen.

$$\delta \mathbf{w} \Big|_{L_e} \approx \sum_{i=1}^{nen} A^{(i)} \delta \mathbf{w}^{(i)} = \mathbf{A} \delta \mathbf{w}^e \quad (4.4)$$

$$\Delta \mathbf{w} \Big|_{L_e} \approx \sum_{i=1}^{nen} A^{(i)} \Delta \mathbf{w}^{(i)} = \mathbf{A} \Delta \mathbf{w}^e \quad (4.5)$$

Die diskretisierte Referenzkonfiguration bezeichnen wir mit $\Phi_0^e := (\mathbf{u}_0^e, \mathbf{\Lambda}_0^e)$. Analog dazu sind Φ^e und $\bar{\Phi}^e$ definiert.

Wir definieren nun die Matrizen $\mathbf{A}, \partial \mathbf{A} \in \mathbb{R}^{3 \times 3nen}$, um die Linearkombinationen mittels einer Matrix-Vektor-Multiplikation berechnen zu können. Da innerhalb eines Elementes ausschließlich nach der Variablen ξ differenziert wird, können wir die Bezeichnung der Ableitung $\partial := \frac{\partial}{\partial \xi}$ entsprechend vereinfachen.

$$\mathbf{A} := \begin{pmatrix} A^{(1)}(\xi) & 0 & 0 & A^{(2)}(\xi) & 0 & 0 & A^{(nen)}(\xi) & 0 & 0 \\ 0 & A^{(1)}(\xi) & 0 & 0 & A^{(2)}(\xi) & 0 & \dots & 0 & A^{(nen)}(\xi) \\ 0 & 0 & A^{(1)}(\xi) & 0 & 0 & A^{(2)}(\xi) & 0 & 0 & A^{(nen)}(\xi) \end{pmatrix}$$

$$\partial \mathbf{A} := \begin{pmatrix} \partial A^{(1)}(\xi) & 0 & 0 & \partial A^{(nen)}(\xi) & 0 & 0 \\ 0 & \partial A^{(1)}(\xi) & 0 & \dots & 0 & \partial A^{(nen)}(\xi) & 0 \\ 0 & 0 & \partial A^{(1)}(\xi) & 0 & 0 & 0 & \partial A^{(nen)}(\xi) \end{pmatrix}$$

Die Referenzkonfiguration \mathbf{u}_0 kann wahlweise über eine Linearkombination dargestellt

werden. Dann gilt

$$\mathbf{u}_0 \Big|_{L_e} \approx \sum_{i=1}^{nen} A_{ref}^{(i)} \mathbf{u}_0^{(i)} = \mathbf{A}_{ref} \mathbf{u}_0^e.$$

Da die Knotenwerte \mathbf{u}_0^e schon bekannt sind, ist auch eine allgemeinere, nichtlineare Darstellung möglich.

$$\mathbf{u}_0 \Big|_{L_e} \approx \mathbf{A}_{ref} \left(\mathbf{u}_0^{(1)}, \dots, \mathbf{u}_0^{(nen)} \right)$$

Dabei ist \mathbf{A}_{ref} eine beliebige, auch nichtlineare, stetige Funktion. Beispielsweise kann es ein Kreisbogen in Abhängigkeit der beiden Endpunkte $\mathbf{u}_0^{(1)}$, $\mathbf{u}_0^{(2)}$ und des Kreismittelpunkts $\mathbf{u}_0^{(3)}$ sein.

4.2.2 Berechnung der Jacobimatrix J

Der Wert der Jacobimatrix $J := \frac{\partial s_0}{\partial \xi}$ wird an allen Gauß'schen Integrationspunkten benötigt. Sie ist aufgrund der eindimensionalen Struktur des Balkens lediglich ein Skalar. Wir nehmen an, dass unsere Finite Elemente Diskretisierung so gewählt ist, dass die Jacobimatrix positiv ist.

$$J = \frac{\partial s_0}{\partial \xi} > 0$$

Somit ist die Jacobimatrix J invertierbar. In der Referenzkonfiguration ist \mathbf{u}_0 nach der Bogenlänge s_0 parametrisiert, deshalb gilt

$$\left\| \frac{\partial}{\partial s_0} \mathbf{u}_0 \right\| = 1.$$

Wir können die Jacobimatrix J durch folgende Gleichung berechnen.

$$J \Big|_{\xi=gip} = \frac{\partial s_0}{\partial \xi} \Big|_{\xi=gip} = \underbrace{\frac{\partial s_0}{\partial \xi} \Big|_{\xi=gip}}_{>0} \left\| \frac{\partial}{\partial s_0} \mathbf{u}_0 \right\| = \left\| \frac{\partial}{\partial \xi} \mathbf{u}_0 \right\| \Big|_{\xi=gip}$$

4.2.3 Update der Verschiebung \mathbf{u} und der Rotationsmatrix $\mathbf{\Lambda}$

Im Zuge der Newton-Iteration müssen die Verschiebungen und Verdrehungen aktualisiert werden. Falls Verwechslungen auftreten können, bezeichnen wir durch den tief gestellten Index k den aktuellen Schritt der Newton-Iteration.

Verschiebung \mathbf{u}

Bei der Aktualisierung von \mathbf{u} brauchen die Ergebnisse lediglich aufaddiert zu werden. Da die Ansatzfunktionen \mathbf{A} der Verschiebung \mathbf{u} und der inkrementellen Verschiebung $\Delta \mathbf{u}$ in Gleichung (4.1) und (4.3) identisch sind, gilt

$$\mathbf{u}_{k+1}^e = \mathbf{u}_k^e + \Delta \mathbf{u}^e. \quad (4.6)$$

Rotation Λ

Die Lösung des linearen Gleichungssystems der Newton-Iteration liefert den Vektor $\Delta \mathbf{w}_k^e$. Um später die tangentielle Steifigkeitsmatrix aufstellen zu können, muss die Rotationsmatrix an einzelnen Stellen, den Gauß'schen Integrationspunkten, berechnet werden. Im k -ten Iterationsschritt wird Λ berechnet durch

$$\Lambda_{k+1} \Big|_{\xi=gip} = \Delta \Lambda \Lambda_k \Big|_{\xi=gip},$$

wobei gilt

$$\Delta \Lambda \Big|_{\xi=gip} = \exp \left(\Delta \widehat{\mathbf{W}} \right) \Big|_{\xi=gip}.$$

Um die schiefsymmetrische Matrix $\Delta \widehat{\mathbf{W}} \Big|_{\xi=gip}$ berechnen zu können, interpolieren wir $\Delta \mathbf{w}$ durch Gleichung (4.5)

$$\Delta \mathbf{w} \Big|_{\xi=gip} = \mathbf{A} \Big|_{\xi=gip} \Delta \mathbf{w}^e.$$

Wir interpolieren somit den Wert der inkrementellen Verdrehungen $\Delta \mathbf{w} \in \mathbb{R}^3$ und berechnen die eigentliche Rotationsmatrix Λ nur an den Gauß-Punkten. Die Rotationsmatrix wird nicht an den Knotenpunkten berechnet.

Bemerkung 4.1

Dabei ist zu beachten, dass das zur Anwendung kommende Finite Elemente Programm FEAP die Verschiebungen und Verdrehungen nach jedem Iterationsschritt aufsummiert. Somit muss obige Gleichung (4.6) für die Verschiebungen nicht berechnet werden. Bei den Verdrehungen führt dies jedoch zu dem Problem, dass die Programmausgabe $\tilde{\mathbf{w}}_{k+1}^e = \tilde{\mathbf{w}}_k^e + \Delta \mathbf{w}^e$ und nicht direkt $\Delta \mathbf{w}^e$ ausgibt. Hierbei ist $\tilde{\mathbf{w}}_k^e := \sum_{i=1}^k \Delta \mathbf{w}_i^e$ als Summe der inkrementellen Verdrehungen definiert. Im Allgemeinen gilt jedoch

$$\exp(\mathbf{A} + \mathbf{B}) \neq \exp \mathbf{A} \exp \mathbf{B} \quad \mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}.$$

Daraus folgt für die Verdrehungen

$$\exp \left(\sum_{i=1}^{k+1} \Delta \widehat{\mathbf{W}}_i \right) = \exp \widehat{\mathbf{W}}_{k+1} \neq \exp \widehat{\mathbf{W}}_{k+1} = \Lambda_{k+1} = \prod_{i=1}^{k+1} \Delta \Lambda_i = \prod_{i=1}^{k+1} \exp \left(\Delta \widehat{\mathbf{W}}_i \right).$$

Somit können wir Λ_{k+1} nicht aus $\tilde{\mathbf{w}}_{k+1}$ berechnen, sondern müssen das Produkt auf der rechten Seite auswerten. Nur im Falle identischer Rotationsachsen aller $\Delta \Lambda_i$, wie z.B bei ebenen Problemen, gilt in obiger Formel Gleichheit. In Kapitel 2.2 wird näher auf die Besonderheiten der Rotationsgruppe SO_3 eingegangen.

Aus obiger Bemerkung folgt, dass in jedem Iterationsschritt $\tilde{\mathbf{w}}_k^e$ benötigt wird, um zusammen mit der Programmausgabe $\tilde{\mathbf{w}}_{k+1}^e$ den Vektor $\Delta \mathbf{w}^e$ berechnen zu können. Dies wird mit Hilfe eines besonderen Speicherbereichs, den *history variables* ermöglicht. Allerdings ergeben sich deshalb durch die Subtraktion $\Delta \mathbf{w}^e = \tilde{\mathbf{w}}_{k+1}^e - \tilde{\mathbf{w}}_k^e$ numerische Instabilitäten, da die zusätzliche Verdrehung im k -ten Iterationsschritt bei Konvergenz sehr klein ist.

Bemerkung 4.2

Neben $\tilde{\mathbf{w}}_k^e$ muss auch die aktuelle Rotationsmatrix $\mathbf{\Lambda}_k|_{\xi=gip}$ in den *history variables* gespeichert werden. Um Speicherplatz zu sparen, speichern wir nicht die 9 Einträge der Matrix $\mathbf{\Lambda}$, sondern beschreiben die Matrix $\mathbf{\Lambda}$ durch eine Quaternion $\mathbf{q}_{\mathbf{\Lambda}}$. Dadurch müssen nur noch 4 Einträge abgespeichert werden. Wichtig ist dabei, dass sämtliche benötigten Umformungen numerisch stabil sind und das Produkt der inkrementellen Rotationsmatrizen dabei sehr effizient ermittelt werden kann. Genaueres hierzu ist im Kapitel 2.4 über Quaternionen erläutert.

4.2.4 Alternative Ansätze zum Update der Rotationsmatrix $\mathbf{\Lambda}$

Ein zentraler Punkt dieser Arbeit ist der Vergleich verschiedener Methoden zur Berechnung der Rotationen. Allen Algorithmen gemeinsam ist das Ziel, aus den inkrementellen Rotationen um die globalen Koordinatenachsen \mathbf{e}_i , die an den Knotenpunkten gegeben sind, die Rotationsmatrix $\mathbf{\Lambda}_{k+1}$ an den Gauß'schen Integrationspunkten zu ermitteln. Es ist also in jedem Iterationsschritt ein inkrementelles Update, eine Interpolation und die Transformation des Rotationsvektors zur Rotationsmatrix zu berechnen.

Zuerst müssen wir annehmen, dass die inkrementellen Rotationen um die Koordinatenachsen klein sind, sonst stimmen sie nicht mit dem Rotationsvektor $\Delta\mathbf{w}_i$ überein.

In dieser Arbeit wird zuerst die Interpolation durchgeführt, d.h. $\Delta\mathbf{w}_i$ wird am Gauß-Punkt $\xi = gip$ berechnet. Danach wird die inkrementelle Quaternion $\mathbf{q}_{\Delta\mathbf{\Lambda}_i}$ ermittelt. Nun multiplizieren wir diese mit der gespeicherten Quaternion $\mathbf{q}_{\mathbf{\Lambda}_k}$ und erhalten $\mathbf{q}_{\mathbf{\Lambda}_{k+1}}$, woraus wir leicht $\mathbf{\Lambda}_{k+1}$ berechnen können. $\mathbf{q}_{\mathbf{\Lambda}_{k+1}}$ wird dann für das Update im nächsten Iterationsschritt abgespeichert. Dieser Algorithmus ist beispielsweise bei Ibrahimbegovic [Ibr95] beschrieben. Die Quaternion $\mathbf{q}_{\mathbf{\Lambda}_{k+1}}$ kann anstatt in jedem Iterationsschritt auch nur nach jedem Lastschritt, vgl. Kapitel 4.3.1, abgespeichert werden.

Natürlich kann die Berechnung auch ohne Quaternionen durchgeführt werden, dann ist jedoch mehr Speicherplatz erforderlich und die Rotationsmatrix $\Delta\mathbf{\Lambda}_i$ muss direkt aus dem Rotationsvektor $\Delta\mathbf{w}_i$ ermittelt werden. Falls $\Delta\mathbf{w}_i$ sehr klein ist, können numerische Instabilitäten auftreten, die gesondert umgangen werden müssen.

Als Grundlage für eine alternative Herangehensweise bietet sich [IFK95] an. Dabei wird zuerst der inkrementelle Rotationsvektor $\Delta\mathbf{w}_i$ am Gauß-Punkt interpoliert. Dann wird jedoch aus dem inkrementellen Rotationsvektor $\Delta\mathbf{w}_i$ und dem gespeicherten Rotationsvektor \mathbf{w}_k zuerst der neue Rotationsvektor \mathbf{w}_{k+1} berechnet. Diese Operation ist jedoch keine einfache Addition, sondern ein Operator $T \in \mathbb{R}^{3 \times 3}$ wird hierfür benötigt, siehe [IFK95]. Danach kann die Rotationsmatrix $\mathbf{\Lambda}$ leicht berechnet werden, wahlweise mit oder ohne Quaternionen. Der Vorteil liegt hier in dem nochmals reduzierten Speicherbedarf, allerdings steht für den Operator T nur eine Näherungsformel zur Verfügung, die zudem bei kleinen $\Delta\mathbf{w}_i$ zu numerischer Instabilität neigt und gesondert behandelt werden muss. Außerdem können bei Rotationsvektoren \mathbf{w}_k großer Magnitude Singularitäten auftreten.

In der kürzlich erschienenen Arbeit von Ghosh und Roy [GR08], wird eine Interpolation direkt in der gekrümmten Menge der Einheitsquaternionen beschrieben. Somit werden zu-

erst die Knotenwerte $\Delta \mathbf{w}$ in Quaternionen umgerechnet und dann erst am Gauß-Punkt interpoliert. Danach wird die Rotationsmatrix $\mathbf{\Lambda}$ wie oben bestimmt. Die hierfür benötigte Theorie stammt aus dem Bereich der Computergrafik. Allerdings stehen für höhere Approximationsordnungen noch keine Formeln zur Verfügung, Ghosh und Roy stellen nur eine Näherungsformel für quadratische polynomiale Ansätze vor.

Somit erscheint der hier gewählte Algorithmus zur Berechnung der Rotationsmatrix $\mathbf{\Lambda}_k$ als der effektivste und stabilste Weg, sodass wir die anderen Methoden nicht weiter verfolgen und auf eine Implementierung verzichten.

4.2.5 Update der Verzerrungen $\boldsymbol{\varepsilon}$ und $\boldsymbol{\kappa}$

Analog zu den Verdrehungen müssen auch die Verzerrungen in jedem Iterationsschritt für alle Integrationspunkte neu berechnet werden.

Normal- und Schubverzerrung $\boldsymbol{\varepsilon}$

Für die Normal- und Schubverzerrung $\boldsymbol{\varepsilon}$ gilt

$$\boldsymbol{\varepsilon} = \frac{\partial}{\partial s_0} \bar{\mathbf{u}} - \mathbf{a}_1.$$

Dies können wir durch

$$\boldsymbol{\varepsilon} \Big|_{\xi=gip} = \frac{1}{J} \partial \bar{\mathbf{u}}_{k+1} - \bar{\mathbf{\Lambda}}_{k+1} \mathbf{e}_1 \Big|_{\xi=gip} = \frac{1}{J} \partial \mathbf{A}_{ref}^T \mathbf{u}_0^e + \frac{1}{J} \partial \mathbf{A}^T \mathbf{u}_{k+1}^e - \bar{\mathbf{\Lambda}}_{k+1} \mathbf{e}_1 \Big|_{\xi=gip}$$

berechnen. Allerdings können dabei numerische Instabilitäten auftreten, falls die Verdrehungen groß, jedoch die Verzerrungen nur klein sind. Dann sind die Ableitung $\frac{\partial}{\partial s_0} \bar{\mathbf{u}}$ und der erste Vektor \mathbf{a}_1 des aktuellen Koordinatensystems fast identisch.

Deshalb formen wir die obige Gleichung unter Ausnutzung der Identität $\frac{\partial}{\partial s_0} \mathbf{u}_0 = \mathbf{g}_1$ etwas um.

$$\boldsymbol{\varepsilon} \Big|_{\xi=gip} = \frac{1}{J} \partial \mathbf{A}^T \mathbf{u}_{k+1}^e - (\mathbf{\Lambda}_{k+1} - \mathbf{I}_3) \mathbf{\Lambda}_0 \mathbf{e}_1 \Big|_{\xi=gip}$$

Dies verhindert die numerischen Instabilitäten nicht vollständig, da nun $\mathbf{\Lambda}_{k+1} - \mathbf{I}_3 \approx \mathbf{0}$. In der Anwendung sind jedoch Konvergenzverbesserungen zu beobachten, insbesondere vor dem ersten Lastschritt.

Biegeverzerrung $\boldsymbol{\kappa}$

Aus dem vorherigen Kapitel wissen wir, dass für die Biegeverzerrungen $\boldsymbol{\kappa} = \boldsymbol{\omega}_{k+1}$ gilt. $\boldsymbol{\omega}_{k+1}$ ist dabei der axiale Vektor von $\widehat{\boldsymbol{\Omega}}_{k+1} = \frac{\partial}{\partial s_0} \mathbf{\Lambda}_{k+1} \mathbf{\Lambda}_{k+1}^T$. Die Rotationsmatrix $\mathbf{\Lambda}_{k+1}$ ist schon bekannt, $\frac{\partial}{\partial s_0} \mathbf{\Lambda}_{k+1}$ muss allerdings noch berechnet werden. Es gilt

$$\frac{\partial}{\partial s_0} \mathbf{\Lambda}_{k+1} = \frac{\partial}{\partial s_0} \exp \widehat{\mathbf{W}}_{k+1} = \exp \widehat{\mathbf{W}}_{k+1} \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_{k+1} = \mathbf{\Lambda}_{k+1} \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_{k+1}.$$

Allerdings ist $\widehat{\mathbf{W}}_{k+1}$ bzw. der axiale Vektor \mathbf{w}_{k+1} nicht verfügbar, da im Allgemeinen gilt

$$\mathbf{w}_{k+1} \neq \mathbf{w}_k + \Delta \mathbf{w}.$$

Für die Ableitung können wir jedoch unter Ausnutzung von $\frac{\partial}{\partial s_0} \mathbf{\Lambda} = \frac{\partial}{\partial s_0} \widehat{\mathbf{W}} \mathbf{\Lambda}$ Folgendes zeigen, vgl. hierzu auch [Ibr95], Gleichung (3.20).

$$\begin{aligned} \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_{k+1} &= \mathbf{\Lambda}_{k+1}^T \mathbf{\Lambda}_{k+1} \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_{k+1} \\ &= \mathbf{\Lambda}_{k+1}^T \frac{\partial}{\partial s_0} \mathbf{\Lambda}_{k+1} \\ &= \mathbf{\Lambda}_k^T \Delta \mathbf{\Lambda}^T \frac{\partial}{\partial s_0} (\Delta \mathbf{\Lambda} \mathbf{\Lambda}_k) \\ &= \mathbf{\Lambda}_k^T \Delta \mathbf{\Lambda}^T \frac{\partial}{\partial s_0} (\Delta \mathbf{\Lambda}) \mathbf{\Lambda}_k + \mathbf{\Lambda}_k^T \Delta \mathbf{\Lambda}^T \Delta \mathbf{\Lambda} \frac{\partial}{\partial s_0} \mathbf{\Lambda}_k \\ &= \mathbf{\Lambda}_k^T \Delta \mathbf{\Lambda}^T \Delta \mathbf{\Lambda} \frac{\partial}{\partial s_0} (\Delta \widehat{\mathbf{W}}) \mathbf{\Lambda}_k + \mathbf{\Lambda}_k^T \mathbf{\Lambda}_k \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_k \\ &= \mathbf{\Lambda}_k^T \frac{\partial}{\partial s_0} (\Delta \widehat{\mathbf{W}}) \mathbf{\Lambda}_k + \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_k \end{aligned}$$

Daraus ist ersichtlich, dass sich die Ableitung von $\widehat{\mathbf{W}}_{k+1}$ rekursiv berechnen lässt. Wenn wir dies in die obige Gleichung für $\widehat{\mathbf{\Omega}}_{k+1}$ einsetzen, folgt

$$\begin{aligned} \widehat{\mathbf{\Omega}}_{k+1} &= \frac{\partial}{\partial s_0} \mathbf{\Lambda}_{k+1} \mathbf{\Lambda}_{k+1}^T \\ &= \mathbf{\Lambda}_{k+1} \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_{k+1} \mathbf{\Lambda}_{k+1}^T \\ &= \mathbf{\Lambda}_{k+1} \left(\mathbf{\Lambda}_k^T \frac{\partial}{\partial s_0} (\Delta \widehat{\mathbf{W}}) \mathbf{\Lambda}_k + \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_k \right) \mathbf{\Lambda}_{k+1}^T \\ &= \Delta \mathbf{\Lambda} \mathbf{\Lambda}_k \mathbf{\Lambda}_k^T \frac{\partial}{\partial s_0} (\Delta \widehat{\mathbf{W}}) \mathbf{\Lambda}_k \mathbf{\Lambda}_k^T \Delta \mathbf{\Lambda}^T + \Delta \mathbf{\Lambda} \mathbf{\Lambda}_k \frac{\partial}{\partial s_0} \widehat{\mathbf{W}}_k \mathbf{\Lambda}_k^T \Delta \mathbf{\Lambda}^T \\ &= \Delta \mathbf{\Lambda} \frac{\partial}{\partial s_0} (\Delta \widehat{\mathbf{W}}) \Delta \mathbf{\Lambda}^T + \Delta \mathbf{\Lambda} \frac{\partial}{\partial s_0} \mathbf{\Lambda}_k \mathbf{\Lambda}_k^T \Delta \mathbf{\Lambda}^T \\ &= \Delta \mathbf{\Lambda} \frac{\partial}{\partial s_0} (\Delta \widehat{\mathbf{W}}) \Delta \mathbf{\Lambda}_k^T + \Delta \mathbf{\Lambda} \widehat{\mathbf{\Omega}}_k \Delta \mathbf{\Lambda}^T \\ &= \Delta \mathbf{\Lambda} \left(\frac{\partial}{\partial s_0} (\Delta \widehat{\mathbf{W}}) + \widehat{\mathbf{\Omega}}_k \right) \Delta \mathbf{\Lambda}^T. \end{aligned}$$

Somit haben wir eine leicht zu berechnende Rekursionsformel für $\widehat{\mathbf{\Omega}}_{k+1}$ gefunden. Sinnvollerweise wenden wir die Rekursionsformel jedoch nicht auf die antisymmetrische Matrix $\widehat{\mathbf{\Omega}}_{k+1}$ an. Mit Hilfe von Gleichung (A.7) aus Lemma A.6 können wir die Rekursionsformel direkt auf den axialen Vektor $\boldsymbol{\omega}_k$ anwenden.

$$\boldsymbol{\omega}_{k+1} = \Delta \mathbf{\Lambda} \left(\frac{\partial}{\partial s_0} (\Delta \mathbf{w}) + \boldsymbol{\omega}_k \right)$$

Bemerkung 4.3

Dieses Update benötigt im Gegensatz zu den Implementierungen von Ibrahimbegovic [IF94, Ibr95, IFK95, Ibr97, IT02] keinen zusätzlichen Operator $T \in \mathbb{R}^{3 \times 3}$. Dieser Operator ist gleichwohl zusätzlich in den Elementroutinen implementiert. Welcher der beiden Algorithmen zum Update von κ gewählt wird, hat jedoch keinen erkennbaren Einfluss auf die Ergebnisse.

4.2.6 Berechnung der Spannungsergebnisse \mathbf{n} und \mathbf{m}

Mit Hilfe der Rotationsmatrix Λ und den Verzerrungen ε und κ können nun die Spannungsergebnisse \mathbf{n} und \mathbf{m} berechnet werden. Zuerst berechnen wir die Matrizen der Materialparameter \mathbf{C}_n , \mathbf{C}_m , die bzgl. des lokalen Koordinatensystems \mathbf{a}_i gegeben sind, auf das globale Koordinatensystem \mathbf{e}_i um.

$$\mathbf{D}_n = \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \qquad \mathbf{D}_m = \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T$$

Die Spannungsergebnisse lassen sich jetzt leicht bestimmen.

$$\mathbf{n} = \mathbf{D}_n \varepsilon \qquad \mathbf{m} = \mathbf{D}_m \kappa$$

Um die in den Variationen des Potentials (3.7) und (3.9) auftretenden Vektorkreuzprodukte, leicht berechnen zu können, definieren wir die folgenden schiefsymmetrischen Matrizen. Der Beweis ist im Anhang A, Gleichung (A.1) gegeben.

$$\widehat{\mathbf{N}} := \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix} \qquad \widehat{\mathbf{M}} := \begin{pmatrix} 0 & -m_3 & m_2 \\ m_3 & 0 & -m_1 \\ -m_2 & m_1 & 0 \end{pmatrix}$$

4.2.7 Berechnung der tangentialen Steifigkeitsmatrix $TANG$

Sei $G \subset [-1, 1]$ die Menge der Gauß'schen Integrationspunkte. Die Zahl der Punkte richtet sich nach der Anzahl und Art der Ansatzfunktionen, z.B. polynomial oder nicht. Dies wird in den jeweiligen Abschnitten in Kapitel 5 näher erläutert. Um die tangentiale Steifigkeitsmatrix, d.h. das in jedem Newton-Schritt zu lösende lineare Gleichungssystem, aufstellen zu können, definieren wir zunächst die schiefsymmetrische Matrix $\widehat{\partial \mathbf{U}} := \begin{pmatrix} 0 & -\partial \bar{u}_3 & \partial \bar{u}_2 \\ \partial \bar{u}_3 & 0 & -\partial \bar{u}_1 \\ -\partial \bar{u}_2 & \partial \bar{u}_1 & 0 \end{pmatrix}$ der Ableitung der aktuellen Verschiebung $\partial \bar{\mathbf{u}}$. Dadurch können wir die folgenden Matrizen in jedem Integrationspunkt $gip \in G$ berechnen.

$$\begin{aligned}
KN_{mat} \Big|_{\xi=gip} &:= \partial \mathbf{A}^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \partial \mathbf{A} \Big|_{\xi=gip} \\
KNM_{mat} \Big|_{\xi=gip} &:= \mathbf{A}^T \partial \widehat{\mathbf{U}}^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \partial \mathbf{A} \Big|_{\xi=gip} \\
KMN_{mat} \Big|_{\xi=gip} &:= \partial \mathbf{A}^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \partial \widehat{\mathbf{U}} \mathbf{A} \Big|_{\xi=gip} \\
KM_{mat} \Big|_{\xi=gip} &:= \mathbf{A}^T \partial \widehat{\mathbf{U}}^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \partial \widehat{\mathbf{U}} \mathbf{A} + \partial \mathbf{A}^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \partial \mathbf{A} \Big|_{\xi=gip} \\
KN_{geo} \Big|_{\xi=gip} &:= 0 \\
KNM_{geo} \Big|_{\xi=gip} &:= \mathbf{A}^T \widehat{\mathbf{N}} \partial \mathbf{A} \Big|_{\xi=gip} \\
KMN_{geo} \Big|_{\xi=gip} &:= \partial \mathbf{A}^T \widehat{\mathbf{N}}^T \mathbf{A} \Big|_{\xi=gip} \\
KM_{geo} \Big|_{\xi=gip} &:= \mathbf{A}^T \partial \widehat{\mathbf{U}} \widehat{\mathbf{N}} \mathbf{A} + \partial \mathbf{A}^T \widehat{\mathbf{M}}^T \mathbf{A} \Big|_{\xi=gip}
\end{aligned}$$

Nun wird die tangentielle Steifigkeitsmatrix aufintegriert bzw. über alle Gaußpunkte aufsummiert.

$$\mathbf{TANG}_{mat} := \sum_{gip \in G} \frac{1}{J|_{\xi=gip}} \begin{pmatrix} KN_{mat}|_{\xi=gip} & KNM_{mat}|_{\xi=gip} \\ KMN_{mat}|_{\xi=gip} & KM_{mat}|_{\xi=gip} \end{pmatrix} \quad (4.7)$$

$$\mathbf{TANG}_{geo} := \sum_{gip \in G} \frac{1}{J|_{\xi=gip}} \begin{pmatrix} KN_{geo}|_{\xi=gip} & KNM_{geo}|_{\xi=gip} \\ KMN_{geo}|_{\xi=gip} & KM_{geo}|_{\xi=gip} \end{pmatrix} \quad (4.8)$$

Die Anteile der geometrischen tangentialen Steifigkeitsmatrix sind klein gegenüber jenen der materiellen Matrix. Außerdem können sie je nach zu Grunde liegender Definition der zweiten Gâteaux-Variation von den oben beschriebenen Werten verschieden sein, vgl. hierzu Kapitel 2.3. Oft wird deshalb nur der materielle Teil der tangentialen Steifigkeitsmatrix berechnet.

Beispielsweise in der Definition von KN_{mat} treten es zwei Ableitungen nach ξ auf. Um, wie in der zweiten Variation $\Delta\delta\Pi$ definiert, nach s_0 abzuleiten, müssen wir, entsprechend der Kettenregel, durch J^2 teilen. Bei der Integration bzw. der Gauß-Quadratur folgt analog eine Multiplikation mit J . Somit multiplizieren wir die tangentielle Steifigkeitsmatrix mit $J^{-2} J = \frac{1}{J}$.

Nun kann die zweite Variation des Potentials durch folgende diskrete Gleichung approximiert werden

$$\Delta\delta\Pi \approx \begin{pmatrix} \delta \mathbf{u}^e \\ \delta \mathbf{w}^e \end{pmatrix}^T \underbrace{(\mathbf{TANG}_{mat} + \mathbf{TANG}_{geo})}_{=: \mathbf{TANG}} \begin{pmatrix} \Delta \mathbf{u}^e \\ \Delta \mathbf{w}^e \end{pmatrix}.$$

Wir haben somit die zweite Variation durch eine Bilinearform approximiert. In Kapitel 3.7 haben wir schon die Bilinearität von $\Delta\delta\Pi$ in $(\frac{\delta \mathbf{u}}{\delta \mathbf{w}})$ und $(\frac{\Delta \mathbf{u}}{\Delta \mathbf{w}})$ gezeigt. Die zu Grunde liegende Struktur der Bilinearität der zweiten Variation konnten wir somit erhalten und in ein Matrix-Vektor-Produkt überführen.

4.2.8 Berechnung des Residuums RES

Das Residuum RES stellt die rechte Seite des Gleichungssystems dar. Wir betrachten zwei Möglichkeiten das Residuum zu berechnen.

Berechnung mittels erster Potentialvariation

Das Residuum entspricht der ersten Variation des Potentials $\delta\Pi$. Es ist wie die tangentiale Steifigkeitsmatrix ein Integral und wird deshalb auch als Summe über den Gaußpunkten berechnet. Die Zahl der Gaußpunkte könnte dabei gegenüber der tangentialen Steifigkeitsmatrix reduziert werden, allerdings müssten dafür Schnittkräfte, Verdrehungen, usw. an neuen Punkten $\xi = gip$ berechnet werden. Deshalb werden exakt die gleichen Gauß'schen Integrationspunkte wie bei der tangentialen Steifigkeitsmatrix genutzt.

Wie bei der tangentialen Steifigkeitsmatrix $TANG$ muss auch hier, bei der Ableitung bzw. Integration nach ξ , wieder die Kettenregel berücksichtigt werden. Da jedoch genau eine Ableitung und eine Integration auftritt, heben sich die Jacobimatrizen gegenseitig auf.

Sei $G \subset [-1, 1]$ wieder die Menge der Gauß-Punkte. Wir definieren zuerst

$$RES_{int} := \sum_{gip \in G} \left(\begin{array}{c} \partial A^T \mathbf{n} \Big|_{\xi=gip} \\ A^T \widehat{\partial U}^T \mathbf{n} + \partial A^T \mathbf{m} \Big|_{\xi=gip} \end{array} \right),$$

sodass die erste Variation des Potentials wie folgt approximiert werden kann

$$\delta\Pi_{int} \approx \begin{pmatrix} \delta\mathbf{u}^e \\ \delta\mathbf{w}^e \end{pmatrix}^T RES_{int}.$$

Falls äußere Streckenlasten \mathbf{p} oder \mathbf{q} auftreten, muss noch ein energetisch äquivalenter Knotenlastvektor $\begin{pmatrix} \mathbf{p}^e \\ \mathbf{q}^e \end{pmatrix}$ berechnet werden, um $\delta\Pi_{ext}$ approximieren zu können. Im Zuge dieser Arbeit betrachten wir jedoch nur Lasten an den Knotenpunkten. Bei diesen ist der Knotenlastvektor direkt gegeben.

$$\begin{aligned} \delta\Pi_{ext} &\approx \begin{pmatrix} \delta\mathbf{u}^e \\ \delta\mathbf{w}^e \end{pmatrix}^T \begin{pmatrix} \mathbf{p}^e \\ \mathbf{q}^e \end{pmatrix} \\ \delta\Pi = \delta\Pi_{int} - \delta\Pi_{ext} &\approx \begin{pmatrix} \delta\mathbf{u}^e \\ \delta\mathbf{w}^e \end{pmatrix}^T \underbrace{\left(RES_{int} - \begin{pmatrix} \mathbf{p}^e \\ \mathbf{q}^e \end{pmatrix} \right)}_{=: RES} \end{aligned}$$

Berechnung mittels tangentialer Steifigkeitsmatrix

Eine weitere Möglichkeit zur Berechnung des Residuums ist die Multiplikation der im k -ten Newton-Schritt aufgestellten tangentialen Steifigkeitsmatrix $TANG$ mit dem Verschiebungsvektor $\begin{pmatrix} \mathbf{u}^e \\ \tilde{\mathbf{w}}^e \end{pmatrix}_{k-1}$ nach dem $(k-1)$ -ten Schritt.

$$RES_{int k} := -TANG_k \begin{pmatrix} \mathbf{u}_{k-1}^e \\ \tilde{\mathbf{w}}_{k-1}^e \end{pmatrix}$$

Diese Approximation der ersten Variation $\delta\Pi$ durch \mathbf{RES}_{int} ist sinnvoll, da wir diese mittels Taylorentwicklung an der Stelle $\bar{\Phi} + \Delta\Phi$ in der Nähe der Gleichgewichtskonfiguration $\bar{\Phi}$ wie folgt darstellen können.

$$\underbrace{\delta\Pi(\bar{\Phi}, \delta\Phi)}_{=0} \approx \delta\Pi(\bar{\Phi} + \Delta\Phi, \delta\Phi) + \Delta\delta\Pi(\bar{\Phi} + \Delta\Phi, \delta\Phi, \Delta\Phi) \quad \forall \delta\Phi$$

Somit erleichtert sich die Berechnung, da wir keine Schnittkräfte auswerten müssen. Dieses Verfahren kann lediglich bei kleinen Deformationen angewendet werden, da sonst die Taylorentwicklung nur sehr ungenaue Ergebnisse liefert. Bei großen Verdrehungen konvergiert das Newton-Verfahren im Allgemeinen nicht. Außerdem nehmen wir an dieser Stelle an, dass $\tilde{\mathbf{w}} \approx \mathbf{w}$ gilt. Wir berechnen ausschließlich die materielle tangentialsteifigkeitsmatrix \mathbf{TANG}_{mat} . Sonst müssten die Schnittkräfte für die geometrische tangentialsteifigkeitsmatrix \mathbf{TANG}_{geo} ausgewertet werden, was die Vereinfachungen zunichte macht. Ein Vergleich der beiden Methoden folgt in Kapitel 7 der Elementtests.

4.2.9 Zusammenbau der globalen Steifigkeitsmatrix

In den vorherigen Abschnitten wurde die Berechnung des Residuums \mathbf{RES} und der tangentialen Steifigkeitsmatrix \mathbf{TANG} auf Elementebene betrachtet. Diese müssen nun wieder in den globalen Zusammenhang gebracht werden. Dazu werden in jedem Element für alle lokalen Knoten die entsprechenden globalen Knoten abgespeichert. Die Einträge des Residuums \mathbf{RES} eines einzelnen Elements werden nun an den entsprechenden Stellen des globalen Vektors hinzuaddiert. Analog verfahren wir mit der tangentialen Steifigkeitsmatrix \mathbf{TANG} .

Das verwendete Programm FEAP führt diese Additionen automatisch mit Hilfe der Eingabedatei durch. Deshalb gehen wir auf diesen Bereich an dieser Stelle nicht weiter ein. Da keine Verwechslungen zu befürchten sind, verwenden wir die Bezeichnungen \mathbf{TANG} und \mathbf{RES} im Folgenden sowohl für die lokalen Matrizen und Vektoren im Element, als auch für deren globalen Entsprechungen.

4.3 Newton-Iteration

Das zuvor durch die Finite Elemente Methode diskretisierte Problem (3.8) wird nun mittels dem Newton Verfahren linearisiert.

Unter der Annahme von Differenzierbarkeit ist das Verschwinden der Ableitung eine notwendige Bedingung zur Lösung von Gleichung (3.8). Wir suchen somit eine Konfiguration $\bar{\Phi}$, für die gilt

$$\delta\Pi(\bar{\Phi}, \delta\Phi) = 0 \quad \forall \delta\Phi.$$

Zur Berechnung von Nullstelle nichtlinearer Gleichungen bietet sich das Newton-Verfahren an. Bei eindimensionalen Problemen lautet die Iterationsformel

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad k \in \mathbb{N}^+ \quad x_1 \text{ geeignet.}$$

Unter entsprechenden Zusatzvoraussetzungen konvergiert x_k für $k \rightarrow \infty$ gegen eine Nullstelle von f . Die Konvergenzgeschwindigkeit ist dabei quadratisch. In [DB02], Kapitel 4.2 sind die Zusatzvoraussetzungen näher erläutert und ein Beweis zur Konvergenz des Verfahrens gegeben.

Um das Newton-Verfahren auf ein mehrdimensionales Problem anwenden zu können, müssen wir es leicht umformulieren. Sei $\Delta x_k := x_{k+1} - x_k$, dann ist obige Iterationsformel äquivalent zu

$$f'(x_k) \Delta x_k = -f(x_k).$$

Angewandt auf unser Problem, die Nullstelle der ersten Variation des Potentials zu finden, ergibt sich folgende Iterationsvorschrift

$$\text{Finde } \Delta \Phi_k \text{ mit } \Delta \delta \Pi (\bar{\Phi}_k, \delta \Phi, \Delta \Phi_k) = -\delta \Pi (\bar{\Phi}_k, \delta \Phi) \quad k \in \mathbb{N}^+ \quad \forall \delta \Phi. \quad (4.9)$$

Das Funktional $\Delta \delta \Pi$ ist eine Bilinearform und $\delta \Pi$ eine Linearform.

Nach der Diskretisierung mit der Finite Elemente Methode erhalten wir folgende Gleichung

$$\text{Finde } \begin{pmatrix} \Delta \mathbf{u}_k^e \\ \Delta \mathbf{w}_k^e \end{pmatrix} \text{ mit } \begin{pmatrix} \delta \mathbf{u}^e \\ \delta \mathbf{w}^e \end{pmatrix}^T \mathbf{TANG}_k \begin{pmatrix} \Delta \mathbf{u}_k^e \\ \Delta \mathbf{w}_k^e \end{pmatrix} = - \begin{pmatrix} \delta \mathbf{u}^e \\ \delta \mathbf{w}^e \end{pmatrix}^T \mathbf{RES}_k \quad \forall \begin{pmatrix} \delta \mathbf{u}^e \\ \delta \mathbf{w}^e \end{pmatrix}.$$

Wenn diese Gleichung für alle Testfunktionen $(\delta \mathbf{u}^e, \delta \mathbf{w}^e)$ gelten soll, muss folgendes Gleichungssystem erfüllt sein

$$\mathbf{TANG}_k \begin{pmatrix} \Delta \mathbf{u}_k^e \\ \Delta \mathbf{w}_k^e \end{pmatrix} = -\mathbf{RES}_k \quad k \in \mathbb{N}^+ \quad \begin{pmatrix} \mathbf{u}_1^e \\ \mathbf{w}_1^e \end{pmatrix} \text{ geeignet.} \quad (4.10)$$

Gleichung (4.10) stellt nun das in jedem Iterationsschritt zu lösende lineare Gleichungssystem dar.

Die Lösung $\begin{pmatrix} \Delta \mathbf{u}_k^e \\ \Delta \mathbf{w}_k^e \end{pmatrix}$ wird nach jedem Schritt zur aktuellen Konfiguration $\bar{\Phi}_k^e$ hinzuaddiert. Insbesondere für den Rotationsanteil $\Delta \mathbf{w}_k^e$ muss dabei die besondere Struktur der Mannigfaltigkeit SO_3 berücksichtigt werden. Dies ist in Kapitel 4.2.3 genauer erläutert.

Den ersten Iterationsschritt bezeichnen wir ausnahmsweise mit $k = 1$, da der tief gestellte Index 0 schon die Referenzkonfiguration bezeichnet. Außerdem schreiben wir den Index k des Iterationsschrittes nur, wenn er essentielle Bedeutung hat, z.B. in einer Rekursionsformel. Im Zusammenhang mit Δ wird er fast durchgehend ausgelassen.

Die Iteration wird abgebrochen, wenn ein zuvor definiertes Abbruchkriterium erreicht ist. Bei FEAP muss die Energienorm des Residuums kleiner als ein Schwellwert, z.B. 10^{-16} , sein. Sei nnp die Zahl der Knotenpunkte. Die Norm berechnet sich dann durch

$$\sum_{i=1}^{nnp} \mathbf{RES}^{(i)} \Delta \mathbf{u}^{(i)} \approx \Delta \Pi (\bar{\Phi}, \Delta \Phi).$$

Außerdem ist in der Eingabedatei eine maximale Anzahl an Iterationen vorgegeben, meist 100.

Es stellt sich noch die Frage nach einem geeigneten Startwert für das Newton-Verfahren in (4.10). Wir wählen hier $\begin{pmatrix} \mathbf{u}_1^e \\ \mathbf{w}_1^e \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, da keine zusätzliche Information über die Lösung zur Verfügung steht, die einen anderen Startwert begründen könnte.

Danach kann im nachfolgenden Iterationsschritt die neue tangentialsteifigkeitsmatrix \mathbf{TANG}_{k+1} und der neue Residuumsvektor \mathbf{RES}_{k+1} bzgl. der neuen Konfiguration $\bar{\Phi}_{k+1}^e$ bestimmt werden.

Der oben erwähnte Konvergenzbeweis in [DB02] gilt hier jedoch nicht, da die zu Grunde liegende Definitionsmenge $\mathbb{R}^3 \times SO_3$ kein Vektorraum, sondern nur eine Mannigfaltigkeit ist. Außerdem ist die Diskretisierung mit der Finite Element Methode nur approximativ, sodass auch hier ein Fehler auftritt. In Kapitel 7 der Elementtests betrachten wir die dabei auftretenden Konvergenzprobleme und die verschlechterte Konvergenzgeschwindigkeit genauer.

4.3.1 Schrittweise Lastauftragung

Um die Konvergenz zu verbessern oder überhaupt zu erreichen, kann die Belastung in kleinen Lastschritten aufgetragen werden. Da unsere Annahme, dass $(\frac{\Delta \mathbf{u}^e}{\Delta \mathbf{w}^e})$ klein ist, vor allem in den ersten Schritten der Newton-Iteration verletzt wird, kann bei dreidimensionalen Problemen durch die Wahl kleiner Lastschritte auch die Exaktheit der Lösung verbessert werden. Ein weiterer Grund folgt mit dem Kontaktalgorithmus in Kapitel 6. Wenn die Deformationen zwischen zwei Iterationsschritten in der Größenordnung der Radien der Balken oder größer sind, kann unter Umständen ein vorhandener Kontakt nicht erkannt werden.

Somit tragen wir, entsprechend den Gegebenheiten, die Belastung in mehreren Lastschritten auf. In FEAP ist ein entsprechendes Makro vorhanden. Dabei steigt die Rechenzeit jedoch unter Umständen stark an. In Kapitel 7 werden entsprechende Tests durchgeführt, um den Einfluss der Größe der Lastschritte zu bestimmen.

4.3.2 Eine Alternative zum Newton-Verfahren

Um Rechenzeit einzusparen besteht die Möglichkeit, die Belastung schrittweise aufzutragen und dabei die maximale Zahl an zulässigen Newton-Schritten so stark zu minimieren, dass keine Konvergenz eintritt. Im Extremfall kann die Newton-Iteration ganz weggelassen werden, d.h. in jedem Lastschritt wird nur der erste Iterations-Schritt berechnet. Wir ermitteln somit in jedem Lastschritt das Gleichgewicht wie bei linearen Theorien im vorhergehenden Zustand. Durch die Wahl hinreichend kleiner Lastschritte wird ein nahezu exaktes Ergebnis erreicht.

Dieses Verfahren eignet sich, um schnell einen Überblick über die Größenordnung der Lösung zu erhalten oder wenn das Newton-Verfahren nicht konvergiert. Die Güte der Ergebnisse und die benötigte Rechenzeit wird im Kapitel 7 der Elementtests verglichen.

Kapitel 5

Elementroutinen

In diesem Kapitel sind Einzelheiten zu den implementierten Elementroutinen angegeben. Insbesondere werden dabei die verschiedenen Ansatzfunktionen hergeleitet und definiert.

5.1 2-Knoten Balkenelemente

Die einfachsten und damit am schnellsten zu berechnenden Balkenelemente besitzen nur zwei Knoten. Wir wählen die beiden Ansatzfunktionen $A^{(1)}$ und $A^{(2)}$, sodass gilt

$$\begin{aligned} A^{(1)}(-1) &= 1 & A^{(1)}(1) &= 0 \\ A^{(2)}(-1) &= 0 & A^{(2)}(1) &= 1. \end{aligned}$$

Somit entspricht \mathbf{u}_i^e dem Funktionswert $\mathbf{u}(s_0 = i)$ an der Stelle des i -ten Knotens ($i = 1, 2$). Dies ist bei späteren Elementen nicht mehr für alle Knoten gegeben. Außerdem sollen $A^{(1)}$ und $A^{(2)}$ linear sein, damit lineare Funktionen exakt dargestellt werden können. Die Ansatzfunktionen lauten dann

$$A^{(1)}(\xi) = \frac{1 - \xi}{2} \quad A^{(2)}(\xi) = \frac{1 + \xi}{2} \quad \xi \in [-1, 1]$$

Definieren wir $\xi^{(1)} := -1$ und $\xi^{(2)} := 1$, so folgt die Lagrange-Bedingung $A^{(i)}(\xi^{(j)}) = \delta_{ij}$ mit dem Kroneckersymbol δ_{ij} . Da die Lagrange-Bedingung erfüllt ist, gilt auch $A^{(1)}(\xi) + A^{(2)}(\xi) = 1 \quad \forall \xi \in [-1, 1]$.

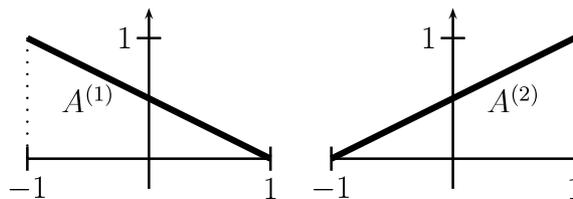


Abbildung 5.1: Ansatzfunktionen der 2-Knoten Elemente *elmt33* und *elmt82*

Die Matrizen \mathbf{A} , $\partial\mathbf{A} \in \mathbb{R}^{3 \times 6}$ haben somit folgende Form

$$\mathbf{A} = \begin{pmatrix} \frac{1-\xi}{2} & 0 & 0 & \frac{1+\xi}{2} & 0 & 0 \\ 0 & \frac{1-\xi}{2} & 0 & 0 & \frac{1+\xi}{2} & 0 \\ 0 & 0 & \frac{1-\xi}{2} & 0 & 0 & \frac{1+\xi}{2} \end{pmatrix}$$

$$\partial\mathbf{A} = \begin{pmatrix} -\frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & -\frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & -\frac{1}{2} & 0 & 0 & \frac{1}{2} \end{pmatrix}.$$

Wir wählen zur Approximation der Referenzkonfiguration die gleichen Ansatzfunktionen

$$A_{ref}^{(i)}(\xi) = A^{(i)}(\xi) \quad i = 1, 2 \quad \xi \in [-1, 1].$$

Ein wichtiger Aspekt ist, dass die Ableitungen der Ansatzfunktionen ∂A und ∂A_{ref} nicht von ξ abhängen. Sie müssen deshalb nicht in jedem Gauß'schen Integrationspunkt neu ausgewertet werden.

5.1.1 Kleine Deformationen - *elmt33*

Dieses Element wurde von Herrn Dr. Alexander Konyukhov entwickelt und implementiert. Die Referenzkonfiguration kann gerade Balken von beliebiger Länge in Richtung der globalen Koordinate \mathbf{e}_1 beschreiben. Wir nehmen an, dass nur kleine Deformationen auftreten. Dadurch wird der Anwendungsbereich stark eingeschränkt, jedoch ergeben sich auch zahlreiche Vereinfachungen, die insbesondere Rechenzeit einsparen.

Die einzelnen Vereinfachungen sind:

- Materialparameter:
Die Materialparameter \mathbf{C}_n und \mathbf{C}_m sind bzgl. dem lokalen Koordinatensystem \mathbf{a}_i gegeben und müssen mit Hilfe von $\bar{\mathbf{\Lambda}}$ in das globale Koordinatensystem \mathbf{e}_i überführt werden. Hier lassen wir diesen Schritt aus, da $\mathbf{a}_i \approx \mathbf{e}_i$. Deshalb gilt

$$\mathbf{D}_n \approx \mathbf{C}_n, \quad \mathbf{D}_m \approx \mathbf{C}_m.$$

Den Fehler beim Gleichsetzen der obigen Werte sehen wir als vernachlässigbar an.

- Tangentiale Steifigkeitsmatrix:
Wir berechnen nur den materiellen Teil der tangentialen Steifigkeitsmatrix \mathbf{TANG}_{mat} und berücksichtigen den geometrischen Teil \mathbf{TANG}_{geo} nicht. Dadurch verschlechtert sich die Konvergenzgeschwindigkeit etwas. Bei kleinen Deformationen ist dies jedoch kaum bemerkbar. Die Spannungsresultierenden \mathbf{n} und \mathbf{m} brauchen so nicht berechnet werden.
- Residuum:
Das Residuum \mathbf{RES} berechnen wir über die tangentielle Steifigkeitsmatrix.

$$\mathbf{RES}_{k+1} = -\mathbf{TANG}_{k+1} \begin{pmatrix} \Delta \mathbf{u}_k^e \\ \Delta \mathbf{w}_k^e \end{pmatrix}$$

Da hier die Spannungsresultierenden \mathbf{n} , und \mathbf{m} , wie bei der tangentialen Steifigkeitsmatrix **TANG**, nicht berechnet werden müssen, kann ein großer Teil des Algorithmus übersprungen werden. Dies ist beispielsweise das Update der Rotationsmatrix $\mathbf{\Lambda}$ und der Biegeverzerrungen $\boldsymbol{\kappa}$.

Die optimale Zahl der Gauß'schen Integrationspunkte ist noch zu bestimmen. Dazu betrachten wir zuerst die tangential Steifigkeitsmatrix **TANG**. Aus der Definition von KM_{mat} erkennen wir, dass KM_{mat} ein quadratisches Polynom in ξ ist. Alle weiteren Anteile von **TANG** sind auch von quadratischer oder niedrigerer Ordnung. Somit ist **TANG** ein quadratisches Polynom in ξ . Die Gauß-Legendre-Quadratur mit $ngip$ Stützstellen ist exakt für Polynome vom Grad $2 \cdot ngip - 1$. Wir wählen daher $ngip = 2$, um den Rechenaufwand zu minimieren.

5.1.2 Endliche Deformationen - *elmt82*

Hier beschränken wir uns nicht auf kleine Deformationen. Sämtliche Vereinfachungen, die durch die Annahme kleiner Deformationen möglich waren, fallen deshalb weg.

Die Elementroutine besitzt neben den Materialparametern zwei weitere Parameter, um die Berechnungsweise der tangentialen Steifigkeitsmatrix **TANG** und des Residuums **RES** auswählen zu können. Eine genaue Beschreibung der Parameter ist in Anhang B gegeben. Es sind beliebige, auch gekrümmte, Referenzkonfigurationen möglich.

Die Zahl der Gauß'schen Integrationspunkte muss neu bestimmt werden. Da jedoch die stark nichtpolynomiale Rotationsmatrix $\bar{\mathbf{\Lambda}}$ nun explizit in der Berechnung von **TANG** und **RES** auftreten kann, wird auch mit vielen Integrationspunkten die Exaktheit der Quadratur nicht länger sichergestellt. Wir setzen wie bei *elmt33* $ngip = 2$. Die Zahl der Integrationspunkte $ngip$ kann bei Bedarf geändert werden, z.B. bei reduzierter Integration. Allerdings muss die Elementroutine hierfür neu kompiliert werden.

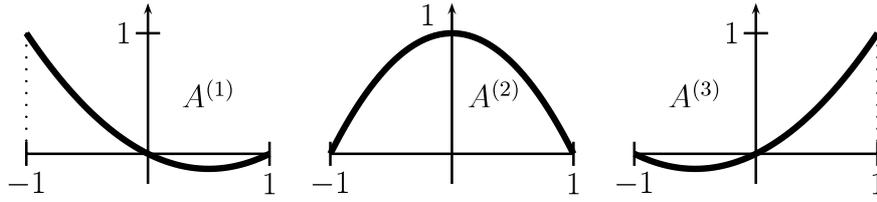
5.2 Das 3-Knoten Balkenelement - *elmt83*

Das *elmt83* vereinigt mehrere 3-Knoten Elemente mit unterschiedlichen Ansatzfunktionen. Es stehen zusätzliche Parameter zur Verfügung, um die entsprechenden Ansatzfunktionen auszuwählen. Näheres zur Verwendung dieser Parameter ist in Anhang B zu finden.

5.2.1 Lagrange Ansatz

Zur Konstruktion der Ansatzfunktionen A zur Interpolation der Verschiebungen nutzen wir die Lagrange-Bedingung, welche auch bei den linearen Elementen erfüllt ist. Dazu definieren wir

$$\xi^{(1)} := -1, \quad \xi^{(2)} := 0, \quad \xi^{(3)} := 1.$$

Abbildung 5.2: Lagrange Ansatzfunktionen des 3 Knoten Elements *elmt83*

Die Ansatzfunktionen sind nun quadratische Polynome, mit $A^{(i)}(\xi^{(j)}) = \delta_{ij}$. Die dadurch definierten Funktionen sind

$$A^{(1)}(\xi) = \frac{\xi^2 - \xi}{2}, \quad A^{(2)}(\xi) = 1 - \xi^2, \quad A^{(3)}(\xi) = \frac{\xi^2 + \xi}{2} \quad \xi \in [-1, 1].$$

Die Matrizen \mathbf{A} , $\partial\mathbf{A} \in \mathbb{R}^{3 \times 9}$ haben somit folgende Form

$$\mathbf{A} = \begin{pmatrix} \frac{\xi^2 - \xi}{2} & 0 & 0 & 1 - \xi^2 & 0 & 0 & \frac{\xi^2 + \xi}{2} & 0 & 0 \\ 0 & \frac{\xi^2 - \xi}{2} & 0 & 0 & 1 - \xi^2 & 0 & 0 & \frac{\xi^2 + \xi}{2} & 0 \\ 0 & 0 & \frac{\xi^2 - \xi}{2} & 0 & 0 & 1 - \xi^2 & 0 & 0 & \frac{\xi^2 + \xi}{2} \end{pmatrix}$$

$$\partial\mathbf{A} = \begin{pmatrix} \xi - \frac{1}{2} & 0 & 0 & -2\xi & 0 & 0 & \xi + \frac{1}{2} & 0 & 0 \\ 0 & \xi - \frac{1}{2} & 0 & 0 & -2\xi & 0 & 0 & \xi + \frac{1}{2} & 0 \\ 0 & 0 & \xi - \frac{1}{2} & 0 & 0 & -2\xi & 0 & 0 & \xi + \frac{1}{2} \end{pmatrix}.$$

Zur Approximation der Referenzkonfiguration können die gleichen quadratischen Ansatzfunktionen $A_{ref}^{(i)}$ gewählt werden.

$$A_{ref}^{(i)}(\xi) = A^{(i)}(\xi) \quad i = 1, 2, 3 \quad \xi \in [-1, 1]$$

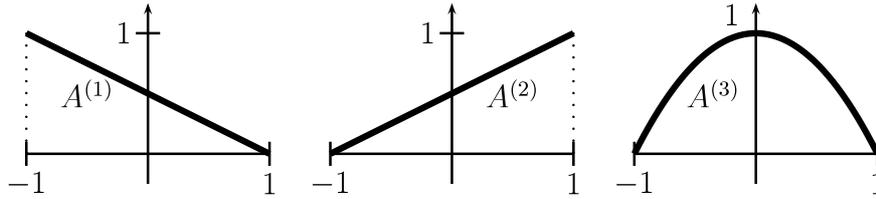
5.2.2 Hierarchischer Ansatz

Hier nutzen wir nicht die Lagrange-Bedingung zur Konstruktion der Ansatzfunktionen. Stattdessen verfolgen wir das Konzept der hierarchischen Ansatzfunktionen, wie es z.B. bei Solin [SSD03] beschrieben ist. Dabei ist die Zahl der Ansatzfunktionen größer als die Zahl der realen Knoten. Es werden virtuelle Knoten eingeführt, um die Zahl der Gleichungen und der Unbekannten wieder anzugleichen.

Wir konstruieren zunächst ein Element mit zwei Knoten und linearen Ansatzfunktionen nach der Lagrange-Bedingung, wie bei *elmt82*. Dieses beschreibt die Lage der Endpunkte unseres quadratischen Elements und interpoliert linear zwischen diesen Beiden. Für $A^{(1)}$ und $A^{(2)}$ gilt

$$A^{(1)}(\xi) = \frac{1 - \xi}{2}, \quad A^{(2)}(\xi) = \frac{1 + \xi}{2} \quad \xi \in [-1, 1].$$

$A^{(1)}$ und $A^{(2)}$ bezeichnen wir als externe Ansatzfunktionen, da sie zusammen mit den zugehörigen Knotenwerten die Lage des Elements nach außen hin beschreiben. Ihr Support erstreckt sich, auf den ganzen Balken betrachtet, über mehr als ein Element.

Abbildung 5.3: Hierarchische Ansatzfunktionen des 3-Knoten Elements *elmt83*

Außerdem definieren wir eine quadratische Funktion, die so genannte *bubble function*, die z.B. Krümmungen innerhalb des Elements wiedergibt. Da sie keinen Einfluss auf die Lage der Knoten haben soll, muss sie an den Knoten den Wert 0 annehmen.

$$A^{(3)}(\xi) = 1 - \xi^2 \quad \xi \in [-1, 1]$$

$A^{(3)}$ heißt auch interne Ansatzfunktion, da sie zusammen mit \mathbf{u}^e nur den Verlauf der Schwerachse im Innern des Elements selbst beschreibt und ihr Support nur in einem einzigen Element liegt. Die Ansatzfunktionen der Referenzkonfiguration $A_{ref}^{(i)}$ ($i = 1, 2, 3$) können ebenfalls nach dem hierarchischen Ansatz gewählt werden. Wir teilen die Matrizen \mathbf{A} , $\partial\mathbf{A} \in \mathbb{R}^{3 \times 9}$ in einen externen und einen internen Teil auf, um die unterschiedliche Bedeutung des externen und internen Anteils zu verdeutlichen.

$$\mathbf{A} = (\mathbf{A}^{ext} \quad \mathbf{A}^{int}) = \begin{pmatrix} \frac{1-\xi}{2} & 0 & 0 & \frac{1+\xi}{2} & 0 & 0 & 1-\xi^2 & 0 & 0 \\ 0 & \frac{1-\xi}{2} & 0 & 0 & \frac{1+\xi}{2} & 0 & 0 & 1-\xi^2 & 0 \\ 0 & 0 & \frac{1-\xi}{2} & 0 & 0 & \frac{1+\xi}{2} & 0 & 0 & 1-\xi^2 \end{pmatrix}$$

$$\partial\mathbf{A} = (\partial\mathbf{A}^{ext} \quad \partial\mathbf{A}^{int}) = \begin{pmatrix} -\frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & -2\xi & 0 & 0 \\ 0 & -\frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & -2\xi & 0 \\ 0 & 0 & -\frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 & 0 & -2\xi \end{pmatrix}$$

Die zusätzlichen Freiheitsgrade im virtuellen Knoten repräsentieren nun nicht mehr die Lage oder den Wert der Verdrehung des Knotens, sondern geben nur noch auf abstrakte Weise die Form des Elements wieder. Beide Elemente sind mathematisch gleichwertig, da die hierarchischen Ansatzfunktionen genauso wie die Ansatzfunktionen nach Lagrange quadratische Polynome exakt darstellen. Dies prüfen wir bei späteren Elementtests nach. Das Konzept der hierarchischen Ansatzfunktionen dient als Grundlage für weitere Elemente. Dabei bleibt der externe Teil der Ansatzfunktionen \mathbf{A}^{ext} gleich, nur der interne Teil \mathbf{A}^{int} ändert sich.

5.2.3 Rationale Bézier-Kurven

Durch die Verwendung von rationalen Bézier-Kurven werden auch Balkengeometrien exakt beschrieben, die mittels polynomialer Kurven nur approximativ dargestellt werden können, wie z.B. Kreisabschnitte. Die Theorie zu rationalen Bézier-Kurven wurde bereits in Abschnitt 2.5 erläutert. Hier beschränken wir uns auf die Implementierung der Ansatzfunktionen bei schon bekannten Gewichten w_i .

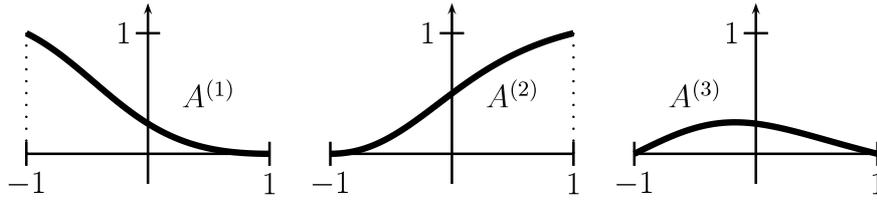


Abbildung 5.4: Rationale Ansatzfunktionen des 3-Knoten Elements *elmt83* mit den Gewichten $w_0 = 1, w_1 = 1, w_2 = 2$

Für das 3-Knoten Element *elmt83* benötigen wir Bernstein-Polynome zweiter Ordnung $B_{i,2}$. Diese sind bzgl. des Parameters $t \in [0, 1]$ parametrisiert. Da das Referenzelement durch $\xi \in [-1, 1]$ parametrisiert ist, müssen die Polynome entsprechend angepasst werden. Es gilt

$$t(\xi) = \frac{1}{2}(\xi + 1) \qquad \frac{\partial t}{\partial \xi} = \frac{1}{2}.$$

Die Polynome bzgl. ξ werden mit \tilde{B} bezeichnet, um Verwechslungen mit den ursprünglichen Bernstein-Polynomen zu vermeiden. Außerdem sind die Gewichte w_i schon hinzumultipliziert.

$$\tilde{B}_{0,2}(\xi) = w_0(1 - \xi)^2, \quad \tilde{B}_{1,2}(\xi) = w_1(1 - \xi)(1 + \xi), \quad \tilde{B}_{2,2}(\xi) = w_2(1 + \xi)^2$$

Die Ableitungen sind dann

$$\partial \tilde{B}_{0,2}(\xi) = -2w_0(1 - \xi), \quad \partial \tilde{B}_{1,2}(\xi) = -2w_1\xi, \quad \partial \tilde{B}_{2,2}(\xi) = 2w_2(1 + \xi).$$

Wir berechnen außerdem die Summe S aller Polynome und die Summe ∂S der Ableitungen

$$S := \sum_{i=0}^2 \tilde{B}_{i,2}(\xi), \qquad \partial S := \sum_{i=0}^2 \partial \tilde{B}_{i,2}(\xi).$$

Die Ansatzfunktionen zur Verschiebungsapproximation A und zur Geometrieapproximation A_{ref} sowie deren Ableitungen ∂A und ∂A_{ref} lauten nun

$$\begin{aligned} A^{(1)} = A_{ref}^{(1)} &= \frac{\tilde{B}_{0,2}(\xi)}{S}, & \partial A^{(1)} = \partial A_{ref}^{(1)} &= \frac{\partial \tilde{B}_{0,2}(\xi)S - \tilde{B}_{0,2}(\xi)\partial S}{S^2}, \\ A^{(2)} = A_{ref}^{(2)} &= \frac{\tilde{B}_{2,2}(\xi)}{S}, & \partial A^{(2)} = \partial A_{ref}^{(2)} &= \frac{\partial \tilde{B}_{2,2}(\xi)S - \tilde{B}_{2,2}(\xi)\partial S}{S^2}, \\ A^{(3)} = A_{ref}^{(3)} &= \frac{\tilde{B}_{1,2}(\xi)}{S}, & \partial A^{(3)} = \partial A_{ref}^{(3)} &= \frac{\partial \tilde{B}_{1,2}(\xi)S - \tilde{B}_{1,2}(\xi)\partial S}{S^2}. \end{aligned}$$

Dabei ist zu beachten, dass die Nummerierung der Ansatzfunktionen geändert wurde, um zwischen externen und internen Ansatzfunktionen unterscheiden zu können. Die Matrizen $\mathbf{A}, \partial \mathbf{A} \in \mathbb{R}^{3 \times 9}$ werden analog zu den vorherigen Elementen definiert.

Einen schwerwiegenden Kritikpunkt dieses Ansatzes stellt die Wahl der Gewichte w_i dar. Zur Approximation von unterschiedlichen Referenzgeometrien müssen diese jedes Mal neu berechnet werden, was den Aufwand erhöht. Bei der Verschiebungsapproximation entsteht zudem das Problem, dass die Lösung bekannt sein muss, um die idealen Gewichte bestimmen zu können.

5.2.4 Kreisgeometrie

Die Referenzgeometrie muss, im Gegensatz zu den Deformationen, nicht durch eine Linearkombination aus Ansatzfunktionen und Knotenwerten dargestellt werden, d.h. $\mathbf{u}_0 \neq \mathbf{A}_{ref} \mathbf{u}_0^e$ ist möglich. Somit sind auch allgemeinere Geometrien, wie z.B. Kreise, darstellbar.

Ein ebener Kreisbogen im dreidimensionalen Raum kann durch drei Punkte beschrieben werden. Benötigt werden der Anfangspunkt des Bogens, der Endpunkt und der Mittelpunkt der zugehörigen Kugel. Diese drei Punkte bzw. Knoten dürfen jedoch nicht auf einer Geraden liegen. Wir definieren den Anfangspunkt durch $\mathbf{u}_0^{(1)}$, den Endpunkt durch $\mathbf{u}_0^{(2)}$ und den Mittelpunkt durch $\mathbf{u}_0^{(3)}$. In Abbildung 5.5 sind die Strecken \mathbf{g}_1 , \mathbf{g}_2 und \mathbf{g}_{12}

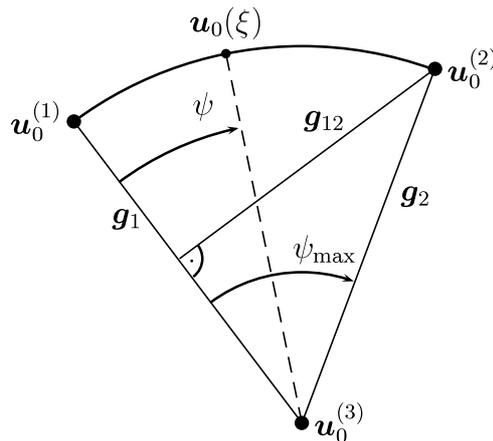


Abbildung 5.5: Kreisgeometrie

definiert. Dabei ist \mathbf{g}_{12} nach Erhard-Schmidt orthogonalisiert.

$$\mathbf{g}_1 := \mathbf{u}_0^{(1)} - \mathbf{u}_0^{(3)} \quad \mathbf{g}_2 := \mathbf{u}_0^{(2)} - \mathbf{u}_0^{(3)} \quad \mathbf{g}_{12} := \mathbf{g}_2 - \frac{\mathbf{g}_1^T \mathbf{g}_2}{\|\mathbf{g}_1\| \|\mathbf{g}_2\|} \mathbf{g}_1$$

Der Bogen \mathbf{u}_0 wird zuerst nach dem Winkel $\psi \in [0, \psi_{\max}]$ parametrisiert. Hierzu muss der maximale Winkel $\psi_{\max} = \arccos\left(\frac{\mathbf{g}_1^T \mathbf{g}_2}{\|\mathbf{g}_1\| \|\mathbf{g}_2\|}\right)$ ermittelt werden. Die Auswertung des Arcuscosinus ist für Winkel $\psi_{\max} \neq 0$ bzw. $\psi_{\max} \neq \pi$ numerisch stabil.

Das Referenzelement ist nicht bzgl. $\psi \in [0, \psi_{\max}]$, sondern bzgl. $\xi \in [-1, 1]$ definiert. Dabei besteht folgender Zusammenhang

$$\psi = \frac{\psi_{\max}}{2}(\xi + 1), \quad \frac{\partial \psi}{\partial \xi} = \frac{\psi_{\max}}{2}.$$

Somit können \mathbf{u}_0 und $\partial\mathbf{u}_0$ wie folgt berechnet werden

$$\mathbf{u}_0 = \mathbf{u}_0^{(3)} + \cos(\psi) \mathbf{g}_1 + \sin(\psi) \frac{\|\mathbf{g}_1\|}{\|\mathbf{g}_{12}\|} \mathbf{g}_{12}$$

$$\partial\mathbf{u}_0 = \frac{\psi_{\max}}{2} \left(-\sin(\psi) \mathbf{g}_1 + \cos(\psi) \frac{\|\mathbf{g}_1\|}{\|\mathbf{g}_{12}\|} \mathbf{g}_{12} \right).$$

5.3 Das 4-Knoten Balkenelement - *elmt84*

Das Element *elmt84* verfügt wie *elmt83* über verschiedene Ansatzfunktionen. Diese werden in FEAP im Block der Materialparameter bestimmt. Näheres zur Verwendung der Parameter ist in Anhang B angegeben.

5.3.1 Lagrange Ansatz

Die polynomialen Ansatzfunktionen besitzen bei 4-Knoten Elementen kubische Ordnung. Sie werden, wie schon bei *elmt82* und *elmt83*, über die Lagrange-Bedingung $A^{(i)}(\xi^{(j)}) = \delta_{ij}$ hergeleitet. Wir definieren

$$\xi^{(1)} := -1, \quad \xi^{(2)} := -\frac{1}{3}, \quad \xi^{(3)} := \frac{1}{3}, \quad \xi^{(4)} := 1.$$

Die Ansatzfunktionen $A^{(i)} = A_{ref}^{(i)}$ lauten nun

$$A^{(1)} = -\frac{1}{16} (1 + 3\xi) (1 - 3\xi) (1 - \xi), \quad A^{(2)} = \frac{9}{16} (1 + \xi) (1 - 3\xi) (1 - \xi),$$

$$A^{(3)} = \frac{9}{16} (1 + \xi) (1 + 3\xi) (1 - \xi), \quad A^{(4)} = -\frac{1}{16} (1 + \xi) (1 + 3\xi) (1 - 3\xi).$$

Die Ableitungen $\partial A^{(i)} = \partial A_{ref}^{(i)}$ sind

$$\partial A^{(1)} = -\frac{27}{16}\xi^2 + \frac{1}{8}\xi + \frac{1}{16}, \quad \partial A^{(2)} = \frac{81}{16}\xi^2 - \frac{1}{8}\xi - \frac{27}{16},$$

$$\partial A^{(3)} = -\frac{81}{16}\xi^2 - \frac{1}{8}\xi + \frac{27}{16}, \quad \partial A^{(4)} = \frac{27}{16}\xi^2 + \frac{1}{8}\xi - \frac{1}{16}.$$

Die Matrizen \mathbf{A} , $\mathbf{A}_{ref} \in \mathbb{R}^{3 \times 12}$ werden wie bei den vorhergehenden Elementen definiert.

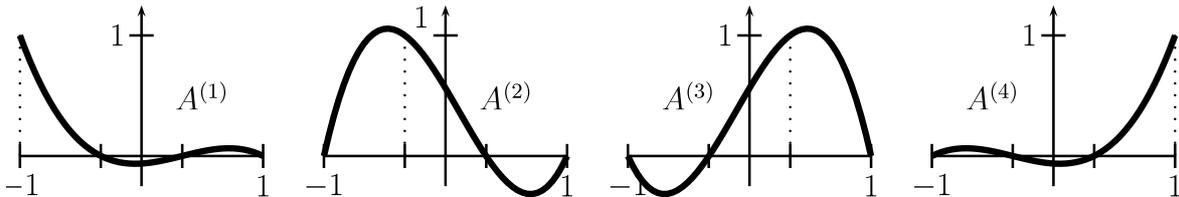


Abbildung 5.6: Lagrange Ansatzfunktionen des 4-Knoten Elements *elmt84*

5.3.2 Hierarchischer Ansatz

Die Vorgehensweise beim hierarchischen Ansatz entspricht dem bei *elmt83*. Nun stehen zwei virtuelle Knoten zur Verfügung, sodass auch eine kubische *bubble function* dargestellt werden kann. Die externen Ansatzfunktionen bleiben gleich

$$A^{(1)}(\xi) = \frac{1 - \xi}{2}, \quad A^{(2)}(\xi) = \frac{1 + \xi}{2} \quad \xi \in [-1, 1].$$

Zur internen Ansatzfunktion $A^{(3)}$ kommt noch das kubische Polynom $A^{(4)}$ hinzu.

$$A^{(3)}(\xi) = 1 - \xi^2, \quad A^{(4)}(\xi) = \xi - \xi^3 \quad \xi \in [-1, 1]$$

Für $\xi = -1$ bzw. $\xi = 1$ sind $A^{(3)}(\xi)$ und $A^{(4)}(\xi)$ gleich 0. Für die Ableitungen gilt

$$\partial A^{(1)}(\xi) = -\frac{1}{2}, \quad \partial A^{(2)}(\xi) = \frac{1}{2}, \quad \partial A^{(3)}(\xi) = -2\xi, \quad \partial A^{(4)}(\xi) = 1 - 3\xi^2 \quad \xi \in [-1, 1].$$

Die gleichen Ansatzfunktionen können wiederum zur Geometrie-Approximation verwendet werden, also $A_{ref}^{(i)} = A^{(i)}$.

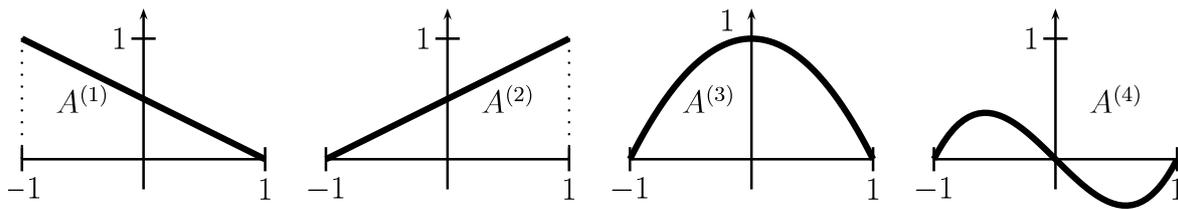


Abbildung 5.7: Hierarchische Ansatzfunktionen des 4-Knoten Elements *elmt84*

5.3.3 Rationale Bézier-Kurven

Wie schon für das 3-Knoten Element definieren wir auch hier rationale Bézier-Kurven zur Geometrie- und Verschiebungsapproximation.

Wir benötigen nun Bernstein-Polynome dritter Ordnung $B_{i,3}$. Diese sind bzgl. des Parameters $t \in [0, 1]$ parametrisiert. Da das Referenzelement durch $\xi \in [-1, 1]$ parametrisiert ist, müssen die Polynome entsprechend angepasst werden. Es gilt

$$t(\xi) = \frac{1}{2}(\xi + 1), \quad \frac{\partial t}{\partial \xi} = \frac{1}{2}.$$

Die Polynome bzgl. ξ werden wieder mit \tilde{B} bezeichnet. Auch die Gewichte w_i sind schon hinzumultipliziert.

$$\begin{aligned} \tilde{B}_{0,3}(\xi) &= w_0(1 - \xi)^3, & \tilde{B}_{1,3}(\xi) &= w_1(1 - \xi)^2(1 + \xi), \\ \tilde{B}_{2,3}(\xi) &= w_2(1 - \xi)(1 + \xi)^2, & \tilde{B}_{3,3}(\xi) &= w_3(1 + \xi)^3 \end{aligned}$$

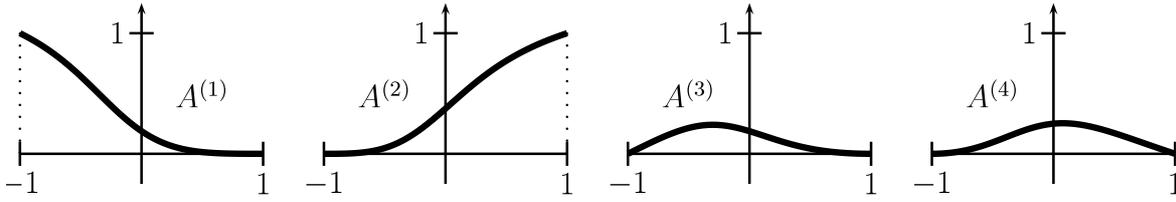


Abbildung 5.8: Rationale Ansatzfunktionen des 4-Knoten Elements *elmt84* mit den Gewichten $w_0 = 1, w_1 = 1, w_2 = \frac{4}{3}, w_3 = 2$

Die Ableitungen sind dann

$$\begin{aligned} \partial \tilde{B}_{0,3}(\xi) &= -3w_0(1 - \xi)^2, & \partial \tilde{B}_{1,3}(\xi) &= -3w_1(1 - \xi)(1 + 3\xi), \\ \partial \tilde{B}_{2,3}(\xi) &= 3w_2(1 + \xi)(1 - 3\xi), & \partial \tilde{B}_{3,3}(\xi) &= 3w_3(1 + \xi)^2. \end{aligned}$$

Wir berechnen außerdem die Summen S und ∂S

$$S := \sum_{i=0}^3 \tilde{B}_{i,3}(\xi), \quad \partial S := \sum_{i=0}^3 \partial \tilde{B}_{i,3}(\xi).$$

Die Ansatzfunktionen zur Verschiebungsapproximation A und zur Geometrieapproximation A_{ref} sowie deren Ableitungen $\partial A, \partial A_{ref}$ lauten nun

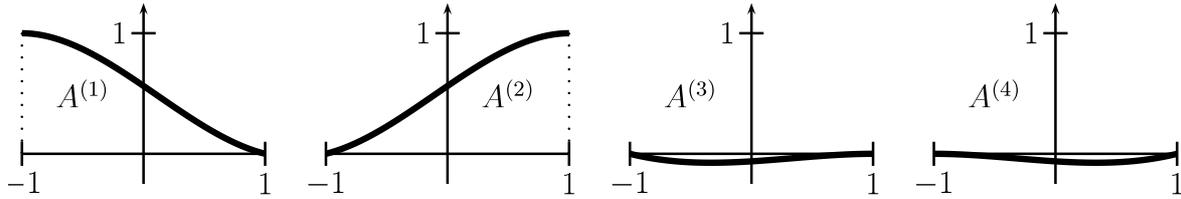
$$\begin{aligned} A^{(1)} = A_{ref}^{(1)} &= \frac{\tilde{B}_{0,3}(\xi)}{S}, & \partial A^{(1)} = \partial A_{ref}^{(1)} &= \frac{\partial \tilde{B}_{0,3}(\xi)S - \tilde{B}_{0,3}(\xi)\partial S}{S^2}, \\ A^{(2)} = A_{ref}^{(2)} &= \frac{\tilde{B}_{3,3}(\xi)}{S}, & \partial A^{(2)} = \partial A_{ref}^{(2)} &= \frac{\partial \tilde{B}_{3,3}(\xi)S - \tilde{B}_{3,3}(\xi)\partial S}{S^2}, \\ A^{(3)} = A_{ref}^{(3)} &= \frac{\tilde{B}_{1,3}(\xi)}{S}, & \partial A^{(3)} = \partial A_{ref}^{(3)} &= \frac{\partial \tilde{B}_{1,3}(\xi)S - \tilde{B}_{1,3}(\xi)\partial S}{S^2}, \\ A^{(4)} = A_{ref}^{(4)} &= \frac{\tilde{B}_{2,3}(\xi)}{S}, & \partial A^{(4)} = \partial A_{ref}^{(4)} &= \frac{\partial \tilde{B}_{2,3}(\xi)S - \tilde{B}_{2,3}(\xi)\partial S}{S^2}. \end{aligned}$$

Die Nummerierung der Ansatzfunktionen wurde wieder geändert, um zwischen externen und internen Ansatzfunktionen unterscheiden zu können. Die Matrizen $\mathbf{A}, \partial \mathbf{A} \in \mathbb{R}^{3 \times 12}$ werden entsprechend definiert.

5.3.4 C^1 -Spline Ansatz

Die kubischen Polynome des 4-Knoten Elements *elmt84* können so gewählt werden, dass eine stetig differenzierbare Funktion dargestellt wird. Die Differenzierbarkeit der Finite-Elemente-Approximation beim Übergang von einem Element zum Nächsten stellt aus mechanischer Sicht eine wünschenswerte Eigenschaft dar. Ein Ansatz hierbei sind kubische Splines. In Kapitel 2.5.1 werden diese mittels Bernstein-Polynomen hergeleitet.

Für das erste und letzte Element eines Balkens müssen die Ansatzfunktionen leicht abgeändert werden. Wir definieren zwei Parameter c_1 und c_2 , die bestimmen, ob ein Element am Anfang, am Ende oder im Inneren des Balkens liegt. Für innere Elemente gilt

Abbildung 5.9: C^1 -Spline Ansatzfunktionen des 4-Knoten Elements *elmt84* mit $c_1 = c_2 = 0$

$c_1 = c_2 = 0$. Bei Anfangselementen wird $c_1 = 1$ gesetzt und bei Elementen am Ende entsprechend $c_2 = 1$. Es ist nicht möglich, dass ein Element zugleich am Anfang und am Ende eines Balkens liegt. Diese Parameter werden zusammen mit den Materialparametern eingelesen. Näheres hierzu ist in Anhang B erläutert.

Gegeben seien o.B.d.A. die Knoten $\mathbf{u}^{(i)}$ ($i = 1, \dots, n$). Zur Berechnung des Elements bzw. des Splines zwischen den Knoten $\mathbf{u}^{(k)}$ und $\mathbf{u}^{(k+1)}$ benötigen wir zusätzlich die Knotenwerte $\mathbf{u}^{(k-1)}$ sowie $\mathbf{u}^{(k+2)}$. Seien $B_{i,3}$ ($i = 0, \dots, 3$) Bernstein-Polynome dritter Ordnung. Dann ist nach Konstruktion die Kurve $\tilde{\mathbf{u}}(t)$ mit

$$\begin{aligned} \tilde{\mathbf{u}}(t) := & B_{0,3}(t)\mathbf{u}^{(k)} + B_{1,3}(t) \left(-\frac{1}{6}\mathbf{u}^{(k-1)} + \mathbf{u}^{(k)} + \frac{1}{6}\mathbf{u}^{(k+1)} \right) \\ & + B_{2,3}(t) \left(\frac{1}{6}\mathbf{u}^{(k)} + \mathbf{u}^{(k+1)} - \frac{1}{6}\mathbf{u}^{(k+2)} \right) + B_{3,3}(t)\mathbf{u}^{(k+1)} \quad t \in [0, 1] \end{aligned}$$

der gesuchte Funktionsverlauf von $\tilde{\mathbf{u}}$. Am linken Rand des Elements gilt $\tilde{\mathbf{u}}|_{t=0} = \mathbf{u}^{(k)}$ und $\frac{\partial}{\partial t}\tilde{\mathbf{u}}|_{t=0} = \frac{1}{2}(\mathbf{u}^{(k-1)} - \mathbf{u}^{(k+1)})$. Entsprechendes gilt am rechten Rand für $t = 1$. Somit ist durch Aneinanderreihung der Elemente die C^1 -Kontinuität gewährleistet.

Die Bernstein-Polynome $B_{i,3}$ sind bzgl. $t \in [0, 1]$ definiert. Nach der Umparametrisierung bzgl. $\xi \in [-1, 1]$ werden sie mit $\tilde{B}_{i,3}$ bezeichnet. Außerdem sind die Sonderfälle des ersten und letzten Elements zu berücksichtigen. Es gilt

$$\begin{aligned} \tilde{B}_{0,3}(\xi) &= \frac{1}{8}(1 - \xi)^3, & \tilde{B}_{1,3}(\xi) &= \frac{3}{8}(1 - \xi)^2(1 + \xi), \\ \tilde{B}_{2,3}(\xi) &= \frac{3}{8}(1 - \xi)(1 + \xi)^2, & \tilde{B}_{3,3}(\xi) &= \frac{1}{8}(1 + \xi)^3. \end{aligned}$$

Die Ableitungen sind entsprechend

$$\begin{aligned} \partial\tilde{B}_{0,3}(\xi) &= -\frac{3}{8}(1 - \xi)^2, & \partial\tilde{B}_{1,3}(\xi) &= -\frac{3}{8}(1 - \xi)(1 + 3\xi), \\ \partial\tilde{B}_{2,3}(\xi) &= \frac{3}{8}(1 + \xi)(1 - 3\xi), & \partial\tilde{B}_{3,3}(\xi) &= \frac{3}{8}(1 + \xi)^2. \end{aligned}$$

Mit den Parametern c_1, c_2 folgen die Ansatzfunktionen $A^{(i)}, \partial A^{(i)}, A_{ref}^{(i)}$ und $\partial A_{ref}^{(i)}$, zur Geometrie- und Verschiebungsapproximation, die nach externen und internen Funktionen geordnet sind.

$$\begin{aligned}
A^{(1)} &= A_{ref}^{(1)} = \tilde{B}_{0,3}(\xi) + \left(1 - \frac{c_1}{2}\right) \tilde{B}_{1,3}(\xi) + \frac{1+3c_2}{6} \tilde{B}_{2,3}(\xi), \\
\partial A^{(1)} &= \partial A_{ref}^{(1)} = \partial \tilde{B}_{0,3}(\xi) + \left(1 - \frac{c_1}{2}\right) \partial \tilde{B}_{1,3}(\xi) + \frac{1+3c_2}{6} \partial \tilde{B}_{2,3}(\xi), \\
A^{(2)} &= A_{ref}^{(2)} = \tilde{B}_{3,3}(\xi) + \frac{1+3c_1}{6} \tilde{B}_{1,3}(\xi) + \left(1 - \frac{c_2}{2}\right) \tilde{B}_{2,3}(\xi), \\
\partial A^{(2)} &= \partial A_{ref}^{(2)} = \partial \tilde{B}_{3,3}(\xi) + \frac{1+3c_1}{6} \partial \tilde{B}_{1,3}(\xi) + \left(1 - \frac{c_2}{2}\right) \partial \tilde{B}_{2,3}(\xi), \\
A^{(3)} &= A_{ref}^{(3)} = \frac{c_1-1}{6} \tilde{B}_{1,3}(\xi) - \frac{c_2}{6} \tilde{B}_{2,3}(\xi), \\
\partial A^{(3)} &= \partial A_{ref}^{(3)} = \frac{c_1-1}{6} \partial \tilde{B}_{1,3}(\xi) - \frac{c_2}{6} \partial \tilde{B}_{2,3}(\xi), \\
A^{(4)} &= A_{ref}^{(4)} = \frac{-c_1}{6} \tilde{B}_{1,3}(\xi) + \frac{c_2-1}{6} \tilde{B}_{2,3}(\xi), \\
\partial A^{(4)} &= \partial A_{ref}^{(4)} = \frac{-c_1}{6} \partial \tilde{B}_{1,3}(\xi) + \frac{c_2-1}{6} \partial \tilde{B}_{2,3}(\xi)
\end{aligned}$$

Die Matrizen $\mathbf{A}, \partial \mathbf{A}, \mathbf{A}_{ref}, \partial \mathbf{A}_{ref} \in \mathbb{R}^{3 \times 12}$ sind wie üblich definiert.

5.4 Die Elemente *elmt86*, *elmt87* und *elmt88*

Die Elemente *elmt86*, *elmt87* und *elmt88* entsprechen in allen Details den Balkenelementen *elmt82*, *elmt83* bzw. *elmt84*. Um jedoch Berechnungen mit den Kontaktelementen *elmt90*, *elmt91* und *elmt92* in FEAP durchführen zu können, muss die Zahl der Knoten bei allen Elementen übereinstimmen.

Deshalb sind bei *elmt86* formal zwei zusätzliche Knoten definiert, die jedoch innerhalb der Elementroutine nicht gebraucht werden. Diese beiden Knoten werden durch zusätzliche Randbedingungen in allen sechs Freiheitsgraden festgehalten, sodass bis auf einen leicht höheren Speicherbedarf keinerlei Änderung des Gesamtsystems auftritt. Damit ist die Kompatibilität des Balkenelements *elmt86* mit dem Kontaktelement *elmt90* gewährleistet. Bei *elmt87* und *elmt88* werden entsprechend drei bzw. vier Knoten hinzugefügt, um Berechnungen zusammen mit den Kontaktelementen *elmt91* und *elmt92* durchführen zu können.

5.5 Die Kontaktelemente *elmt90*, *elmt91* und *elmt92*

Aus Gründen der Übersichtlichkeit sind die meisten Schritte der Implementierung schon in Kapitel 6.5 dargestellt. An dieser Stelle soll nur kurz auf die Definition der Ansatzfunktionen A und A_{ref} eingegangen werden.

Wie in Kapitel 6.5.1 erläutert ist, müssen die Ansatzfunktionen beider Balkenelemente auch im Kontaktelement definiert werden. Insbesondere die Art des Ansatzes, z.B. Lagrange, hierarchisch oder C1-Spline, muss übereinstimmen. Da die rationalen Ansatzfunktionen bei den Elementtests ohne Kontakt keinerlei Vorteile gegenüber den polynomialen Ansatzfunktionen erzielen, sind diese bei den drei Kontaktelementen nicht implementiert. Das Gleiche gilt bei *elmt92* für den Geometriefaktor χ_k des C1-Spline Ansatzes. Für diesen gilt nun auch bei großen Krümmungen $\chi_k = 1 \quad \forall k$. Da die zweite Ableitung der Ansatzfunktionen für die Berechnung des minimalen Abstandes benötigt wird, müssen diese ein zweites Mal abgeleitet werden. In Anhang B sind die Definitionen der Materialparameter aufgeführt. Dabei wurde auch schon die Möglichkeit der Implementierung von Kontakt mit Reibung berücksichtigt.

Kapitel 6

Kontakt

Die in diesem Kapitel vorgestellte Formulierung von Kontakt stützt sich, neben dem Buch von Wriggers [Wri06] auf die Vorlesung “Contact Mechanics” von Dr. Konyukhov. Die beiden Artikel [KS05, KSht] behandeln Kontakt in der Kontinuumsmechanik, sowie bei Balken.

Durch Kontakt der Balken untereinander gehen zusätzliche Nebenbedingungen in das mechanische Problem ein, was weitere Nichtlinearitäten hervorruft. Bei der Variationsformulierung muss nicht länger das globale Minimum der potentiellen Energie Π ermittelt werden, sondern ein Minimum unter Nebenbedingungen. Hierfür existieren drei grundlegende Möglichkeiten der Implementierung: a) Penalisierung, b) Lagrange-Multiplikatoren-Regel und c) erweiterte Lagrange-Multiplikatoren-Regel.

Die Lagrange-Multiplikatoren-Regel erfüllt die, durch Kontakt hervorgerufene, Nebenbedingung exakt, allerdings entstehen dadurch weitere Unbekannte, die Lagrange-Multiplikatoren. Somit wird das zu lösende Gleichungssystem größer, was hier unerwünscht ist. Diese Methode wird deshalb im Zuge dieser Arbeit nicht weiter betrachtet.

Stattdessen wenden wir eine Penalisierung mittels Strafterm an. Wir betrachten den Fall, dass sich zwei Balken nicht nur berühren, sondern gegenseitig durchdringen, d.h. die Nebenbedingung wird verletzt. Dann wird an beiden Balken an der Stelle der größten Durchdringung eine Kraft aufgetragen, die jeweils weg vom anderen Balken weist. Dadurch werden beide Balken auseinander gedrückt, sodass die Nebenbedingung erfüllt ist. Die Größe der Kraft richtet sich dabei nach der Größe der Durchdringung. Es werden keine zusätzlichen Unbekannten benötigt, um das Problem lösen zu können. Die Nebenbedingung wird bei der Penalisierung mittels Strafterm jedoch nicht exakt, sondern nur approximativ erfüllt.

Eine weitere Möglichkeit der Implementierung bieten die erweiterten Lagrange-Multiplikatoren, die beide Ansätze verbinden. Darauf wird an dieser Stelle jedoch nicht weiter eingegangen.

6.1 Generelles Vorgehen

Durch die Penalisierung und das Auftragen zusätzlicher Kräfte ändert sich sowohl die Steifigkeitsmatrix **TANG**, als auch das Residuum **RES** im Bereich des Kontakts. Wir implementieren neben den Balkenelementen ein zusätzliches Kontaktelement, das sich über beide am Kontakt beteiligten Balkenelemente erstreckt. Innerhalb der Elementroutine wird die Differenz der Steifigkeitsmatrix und des Residuums berechnet und beim Zusammenbau der globalen Steifigkeitsmatrix bzw. des globalen Residuums auf die Balkenelemente ohne Kontakt hinzuaddiert. Ein Kontaktelement hat somit immer doppelt so viele Knoten wie das entsprechende Balkenelement. Die Probleme, die im Programm FEAP dabei entstehen, werden in Kapitel 5.4 erläutert.

6.2 Abstandsberechnung

Um Ort und Größe der Durchdringung berechnen zu können, muss zuerst der kleinste Abstand der Balkenschwerachsen zueinander ermittelt werden. Im Gegensatz zu Wriggers [Wri06], Kapitel 12, definieren wir ein zusätzliches lokales Koordinatensystem. Dieses dient als Referenz für sämtliche Kontaktgrößen.

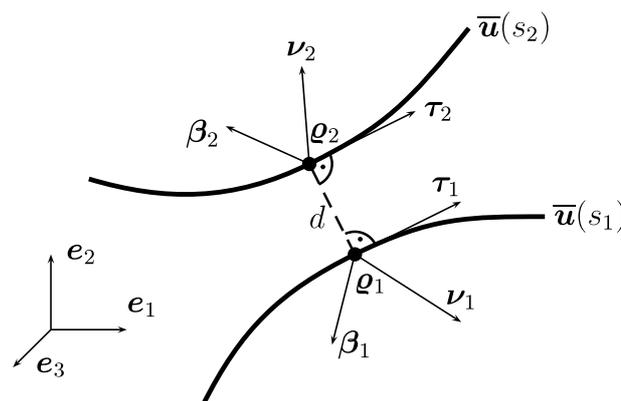


Abbildung 6.1: Berechnung des minimalen Abstands

6.2.1 Geometrie und krummlinige Koordinatensysteme

In Abbildung 6.1 sind die Geometrie und die lokalen Koordinatensysteme der beiden Balken abgebildet. Hier existiert kein “*master*”- und “*slave*”-Balken, wie bei der Kontaktformulierung zwischen Kontinua. Der Verlauf der Schwerachse des ersten Balkens ist nach der Bogenlänge s_1 parametrisiert, der zweite Balken entsprechend nach s_2 .

$$\boldsymbol{q}_1 := \bar{\boldsymbol{u}}(s_1) \qquad \boldsymbol{q}_2 := \bar{\boldsymbol{u}}(s_2)$$

Dabei ist zu beachten, dass die Kurve \boldsymbol{q}_j nicht nur in der Referenzkonfiguration, sondern auch in der aktuellen Konfiguration nach dem Bogenlängenparameter s_j parametrisiert ist. Der Index $j = 1, 2$ bezeichnet im Folgenden die beiden Balken bzw. Kurven.

Die Kurve $\boldsymbol{\rho}_j$ beschreibt den Verlauf der Balkenschwerachse $\bar{\mathbf{u}}$ in Innern eines einzelnen Elements. Wir können daher o.B.d.A. annehmen, dass die Kurve $\boldsymbol{\rho}_j$ zweimal stetig differenzierbar ist. Die Krümmung κ_j ist dann definiert durch

$$\kappa_j := \left\| \frac{\partial^2}{\partial s_j^2} \boldsymbol{\rho}_j \right\|.$$

Wir nehmen an, dass die Krümmung nicht verschwindet, d.h. $\kappa_j > 0 \quad \forall s_j$. Dann können wir die Basisvektoren $\boldsymbol{\tau}_j, \boldsymbol{\nu}_j, \boldsymbol{\beta}_j$ an der Stelle $\boldsymbol{\rho}_j$ wie folgt definieren.

$$\boldsymbol{\tau}_j = \frac{\partial}{\partial s_j} \boldsymbol{\rho}_j, \quad \boldsymbol{\nu}_j = \frac{\frac{\partial^2}{\partial s_j^2} \boldsymbol{\rho}_j}{\kappa_j}, \quad \boldsymbol{\beta}_j = \boldsymbol{\tau}_j \times \boldsymbol{\nu}_j \quad j = 1, 2$$

Aufgrund der Parametrisierung nach der Bogenlänge haben die Basisvektoren $\boldsymbol{\tau}_j, \boldsymbol{\nu}_j, \boldsymbol{\beta}_j$ Einheitslänge und stehen senkrecht zueinander. Der Vektor $\boldsymbol{\tau}$ wird als Tangentialvektor bezeichnet, $\boldsymbol{\nu}$ als Normale und $\boldsymbol{\beta}$ als Binormale.

Der Abstand d der beiden Punkte $\bar{\mathbf{u}}(s_1)$ und $\bar{\mathbf{u}}(s_2)$ ist dann allgemein

$$d = \|\bar{\mathbf{u}}(s_1) - \bar{\mathbf{u}}(s_2)\| = \|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2\|.$$

Um den Abstand d zu minimieren, definieren wir die Funktion F .

$$F(s_1, s_2) = \frac{1}{2} d^2 = \frac{1}{2} \|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2\|^2 \stackrel{!}{=} \text{Min}$$

Da die beiden Kurven $\boldsymbol{\rho}_1$ und $\boldsymbol{\rho}_2$ zweimal stetig differenzierbar sind, ist auch die Funktion F zweimal stetig differenzierbar. Somit muss die Ableitung $\mathbf{grad} F$ an der Stelle des Minimums verschwinden. Dies ist äquivalent zur Bedingung $(\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1) \perp \boldsymbol{\tau}_j$.

$$\mathbf{grad} F = \begin{pmatrix} (\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2)^T \frac{\partial}{\partial s_1} \boldsymbol{\rho}_1 \\ (\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1)^T \frac{\partial}{\partial s_2} \boldsymbol{\rho}_2 \end{pmatrix} = \begin{pmatrix} (\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2)^T \boldsymbol{\tau}_1 \\ (\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1)^T \boldsymbol{\tau}_2 \end{pmatrix} \stackrel{!}{=} \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

Wir haben nun eine geometrische Bedingung für ein minimales d gefunden. Der Vektor $\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1$ muss im Schnitt der beiden Ebenen $(\boldsymbol{\nu}_1, \boldsymbol{\beta}_1)$ und $(\boldsymbol{\nu}_2, \boldsymbol{\beta}_2)$ liegen. Sei o.B.d.A. $d > 0$, dann definieren wir den Einheitsvektor

$$\tilde{\mathbf{e}}_1 := \frac{\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1}{\|\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1\|}.$$

Der Vektor $\tilde{\mathbf{e}}_1$ und dessen Ableitung $\tilde{\mathbf{g}}_1$ lassen sich darstellen als

$$\tilde{\mathbf{e}}_1 = \cos \varphi_1 \boldsymbol{\nu}_1 + \sin \varphi_1 \boldsymbol{\beta}_1, \quad \tilde{\mathbf{g}}_1 = \frac{\partial}{\partial \varphi_1} \tilde{\mathbf{e}}_1 = -\sin \varphi_1 \boldsymbol{\nu}_1 + \cos \varphi_1 \boldsymbol{\beta}_1,$$

wobei φ_1 den Winkel zwischen $\boldsymbol{\nu}_1$ und $\tilde{\mathbf{e}}_1$ angibt. Die Einheitsvektoren $\boldsymbol{\tau}_1, \tilde{\mathbf{e}}_1$ und $\tilde{\mathbf{g}}_1$ stehen senkrecht aufeinander und bilden somit ein Orthonormalsystem. Dies ermöglicht es nun, $\boldsymbol{\rho}_2$ nicht nur in Abhängigkeit von s_2 zu beschreiben, sondern auch bzgl. der krummlinigen Koordinaten s_1, d und φ_1 . Die Parameter sind hier zusätzlich explizit angegeben.

$$\boldsymbol{\rho}_2(s_2) = \boldsymbol{\rho}_2(s_1, d, \varphi_1) = \boldsymbol{\rho}_1(s_1) + d \tilde{\mathbf{e}}_1(\varphi_1)$$

Wir können nun die Hesse-Matrix \mathbf{H} der zu minimierenden Funktion F vereinfachen.

$$\begin{aligned}\mathbf{H} &= \begin{pmatrix} \boldsymbol{\tau}_1^T \boldsymbol{\tau}_1 + (\boldsymbol{q}_1 - \boldsymbol{q}_2)^T \frac{\partial}{\partial s_1} \boldsymbol{\tau}_1 & -\boldsymbol{\tau}_1^T \boldsymbol{\tau}_2 \\ -\boldsymbol{\tau}_1^T \boldsymbol{\tau}_2 & \boldsymbol{\tau}_2^T \boldsymbol{\tau}_2 + (\boldsymbol{q}_2 - \boldsymbol{q}_1)^T \frac{\partial}{\partial s_2} \boldsymbol{\tau}_2 \end{pmatrix} \\ &= \begin{pmatrix} 1 - \kappa_1 d \tilde{\boldsymbol{e}}_1^T \boldsymbol{\nu}_1 & -\boldsymbol{\tau}_1^T \boldsymbol{\tau}_2 \\ -\boldsymbol{\tau}_1^T \boldsymbol{\tau}_2 & 1 - \kappa_2 d \tilde{\boldsymbol{e}}_2^T \boldsymbol{\nu}_2 \end{pmatrix} \\ &= \begin{pmatrix} 1 - \kappa_1 d \cos \varphi_1 & -\cos \psi \\ -\cos \psi & 1 - \kappa_2 d \cos \varphi_2 \end{pmatrix}\end{aligned}$$

Hierbei ist ψ der Winkel zwischen den beiden Tangentialvektoren, $\boldsymbol{\tau}_1^T \boldsymbol{\tau}_2 = \cos \psi$. Da die Hesse-Matrix bei einer Implementierung des Newton-Verfahrens invertiert werden muss, berechnen wir die Determinante

$$\det \mathbf{H} = (1 - \kappa_1 d \cos \varphi_1)(1 - \kappa_2 d \cos \varphi_2) - \cos^2 \psi.$$

Wir können nun für verschiedene Geometrien erkennen, ob Existenz und Eindeutigkeit des minimalen Abstandes vorliegen. Dies wird anhand eines Beispiels erläutert.

Kreise in einer Ebene

Im Fall zweier Kreise, die in der gleichen Ebene liegen, sind die Vektoren $\boldsymbol{\nu}_1, \boldsymbol{\nu}_2$ und $\tilde{\boldsymbol{e}}_1$ parallel zueinander. Dadurch folgt $\varphi_1 = \varphi_2 = 0$. Außerdem ist $\kappa_1 > 0$ und $\kappa_2 > 0$. Für die Determinante $\det \mathbf{H}$ gilt dann

$$\det \mathbf{H} = (1 - \kappa_1 d)(1 - \kappa_2 d) - \cos^2 \psi.$$

An der Stelle des minimalen Abstandes sind die Tangentialvektoren $\boldsymbol{\tau}_1$ und $\boldsymbol{\tau}_2$ parallel zueinander. Somit gilt $\cos^2 \psi = 1$. Damit folgt für die Determinante $\det \mathbf{H} = (1 - \kappa_1 d)(1 - \kappa_2 d) - 1 < 0$. Wir können nun einen maximalen bzw. minimalen Winkel $\psi_{\max}(\kappa_1, \kappa_2, d) = \arccos\left(\sqrt{(1 - \kappa_1 d)(1 - \kappa_2 d)}\right)$ angeben, bei welchem die Determinante das Vorzeichen wechselt. Im Intervall $\psi \in [0, \psi_{\max}]$ konvergiert das Newton-Verfahren.

Gerade Balken

Bei geraden Balken tritt keine Krümmung κ_1 bzw. κ_2 auf. Somit ist das obige Koordinatensystem nicht definierbar. Trotzdem können wir die Abstandsfunktion F definieren und zweimal ableiten. Dies ergibt folgende Hesse-Matrix

$$\mathbf{H} = \begin{pmatrix} \boldsymbol{\tau}_1^T \boldsymbol{\tau}_1 + (\boldsymbol{q}_1 - \boldsymbol{q}_2)^T \frac{\partial}{\partial s_1} \boldsymbol{\tau}_1 & -\boldsymbol{\tau}_1^T \boldsymbol{\tau}_2 \\ -\boldsymbol{\tau}_1^T \boldsymbol{\tau}_2 & \boldsymbol{\tau}_2^T \boldsymbol{\tau}_2 + (\boldsymbol{q}_2 - \boldsymbol{q}_1)^T \frac{\partial}{\partial s_2} \boldsymbol{\tau}_2 \end{pmatrix} = \begin{pmatrix} 1 & -\cos \psi \\ -\cos \psi & 1 \end{pmatrix}.$$

Die Determinante $\det \mathbf{H} = 1 - \cos^2 \psi$ ist positiv, falls $\cos \psi \neq \pm 1$. Eine Lösung existiert somit für $\psi \neq n\pi$ ($n \in \mathbb{Z}$). Geometrisch bedeutet dies, dass die Balken nicht parallel zueinander liegen dürfen.

6.2.2 Durchdringung und Kontaktkraft

Nachdem der Punkt des minimalen Abstandes gefunden wurde, kann der Abstand $d = \|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2\|$ berechnet werden. Seien r_1 und r_2 die Radien der beiden Balkenquerschnitte. Dann gilt für die Durchdringung p

$$p = \|\boldsymbol{\rho}_1 - \boldsymbol{\rho}_2\| - r_1 - r_2 = d - r_1 - r_2.$$

Im Fall $p < 0$ liegt Kontakt vor. Dabei nehmen wir an, dass die Balkenquerschnitte senkrecht auf der Balkenschwerachse stehen, Hypothese nach Euler-Bernoulli, was bei dem in Kapitel 3 definierten Balkenmodell nach Timoshenko nicht der Fall ist. Der Fehler ist jedoch von vernachlässigbarer Größe.

Die Größe der senkrecht auf den Balkenschwerachsen wirkenden, negativen Kontaktkraft N hängt linear von der Durchdringung p ab.

$$N = \epsilon_N p \quad (\epsilon_N \gg 1)$$

Der konstante Strafparameter ϵ_N sollte dabei so groß wie möglich gewählt werden, um die Kontaktbedingung möglichst exakt zu erfüllen. Allerdings wird bei zu großem ϵ_N insbesondere bei großen Lastschritten die Berechnung unter Umständen instabil. Neben der Kontaktkraft N welche senkrecht zu den beiden Balkenschwerachsen wirkt, treten bei positiver Reibung auch Kontaktkräfte T tangential zu den Balken auf. Wir vernachlässigen Reibung jedoch an dieser Stelle.

6.3 Kinematik

Um später die Variation des Potentials berechnen zu können, müssen wir zuerst die Kinematik der zulässigen Variationen ermitteln. Dazu benötigen wir die Formel von Serret-Frenet wie sie z.B. bei Schottenloher [Sch95], Anhang G, zu finden ist.

Satz 6.1 (Formel von Serret-Frenet):

Sei $\boldsymbol{\gamma}$ eine nach der Bogenlänge s_0 parametrisierte zweimal stetig differenzierbare Kurve im \mathbb{R}^3 . Für die Krümmung gelte $\kappa := \left\| \frac{\partial^2}{\partial s_0^2} \boldsymbol{\gamma} \right\| > 0$. Dann können wir folgendes Orthonormalsystem definieren

$$\boldsymbol{\tau} = \frac{\partial}{\partial s_0} \boldsymbol{\gamma}, \quad \boldsymbol{\nu} = \frac{\frac{\partial^2}{\partial s_0^2} \boldsymbol{\gamma}}{\kappa}, \quad \boldsymbol{\beta} = \boldsymbol{\tau} \times \boldsymbol{\nu}.$$

Für die Ableitung folgt dann

$$\frac{\partial}{\partial s_0} \begin{pmatrix} \boldsymbol{\tau} \\ \boldsymbol{\nu} \\ \boldsymbol{\beta} \end{pmatrix} = \begin{pmatrix} 0 & \kappa & 0 \\ -\kappa & 0 & \tau \\ 0 & -\tau & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{\tau} \\ \boldsymbol{\nu} \\ \boldsymbol{\beta} \end{pmatrix}.$$

Der Tangentialvektor $\boldsymbol{\tau}$ (fett gedruckt) und die Torsion τ (dünn gedruckt) sind dabei zu unterscheiden. Für den Beweis wird auf die oben genannte Literatur verwiesen.

Angewendet auf $\tilde{\mathbf{e}}_1$ ergibt die Formel von Serret-Frenet

$$\begin{aligned}\frac{\partial}{\partial s_1} \tilde{\mathbf{e}}_1 &= \cos \varphi_1 \frac{\partial}{\partial s_1} \boldsymbol{\nu} + \sin \varphi_1 \frac{\partial}{\partial s_1} \boldsymbol{\beta} \\ &= (-\kappa_1 \boldsymbol{\tau}_1 + \tau_1 \boldsymbol{\beta}_1) \cos \varphi_1 + (-\tau_1 \boldsymbol{\nu}_1) \sin \varphi_1 \\ &= -\kappa_1 \cos \varphi_1 \boldsymbol{\tau}_1 + \tau_1 \tilde{\mathbf{g}}_1.\end{aligned}$$

Die zulässige Variation von $\tilde{\mathbf{e}}_1$, d.h. die Ableitung von $\tilde{\mathbf{e}}_1$ bzgl. der Kurve $\boldsymbol{\rho}_1$, ist nun

$$\begin{aligned}\delta \tilde{\mathbf{e}}_1 &= \frac{\partial}{\partial s_1} \tilde{\mathbf{e}}_1 \delta s_1 + \frac{\partial}{\partial \varphi_1} \tilde{\mathbf{e}}_1 \delta \varphi_1 \\ &= (-\kappa_1 \cos \varphi_1 \boldsymbol{\tau}_1 + \tau_1 \tilde{\mathbf{g}}_1) \delta s_1 + \tilde{\mathbf{g}}_1 \delta \varphi_1 \\ &= -\delta s_1 \kappa_1 \cos \varphi_1 \boldsymbol{\tau}_1 + (\delta s_1 \tau_1 + \delta \varphi_1) \tilde{\mathbf{g}}_1 \\ &\approx \delta \varphi_1 \tilde{\mathbf{g}}_1.\end{aligned}$$

Die Approximation gilt unter der Annahme, dass Krümmung κ_1 und Torsion τ_1 klein sind und somit die entsprechenden Terme vernachlässigt werden können. Wir nutzen dieses Ergebnis, um die Variation von $\boldsymbol{\rho}_2$ zu berechnen. Hierbei ist zu beachten, dass $(\boldsymbol{\rho}_1 + \delta \boldsymbol{\rho}_1, \boldsymbol{\rho}_2 + \delta \boldsymbol{\rho}_2)$ nicht länger die Punkte mit minimalem Abstand sind. Deshalb muss auch ein Anteil der Variation in Richtung $\boldsymbol{\tau}_1$ berücksichtigt werden.

$$\delta \boldsymbol{\rho}_2 = \delta \boldsymbol{\rho}_1 + \delta s_1 \boldsymbol{\tau}_1 + \delta (d \tilde{\mathbf{e}}_1) = \delta \boldsymbol{\rho}_1 + \delta s_1 \boldsymbol{\tau}_1 + \delta d \tilde{\mathbf{e}}_1 + d \delta \varphi_1 \tilde{\mathbf{g}}_1$$

Da $\boldsymbol{\tau}_1$, $\tilde{\mathbf{e}}_1$ und $\tilde{\mathbf{g}}_1$ senkrecht aufeinander stehen, können wir nun die einzelnen Richtungsanteile der Variationen berechnen. Durch analoge Anwendung am zweiten Balken erhalten wir folgende Variationen der lokalen Koordinaten

$$\begin{aligned}\delta s_1 &= (\delta \boldsymbol{\rho}_2 - \delta \boldsymbol{\rho}_1)^T \boldsymbol{\tau}_1 & \delta s_2 &= (\delta \boldsymbol{\rho}_1 - \delta \boldsymbol{\rho}_2)^T \boldsymbol{\tau}_2 \\ \delta d &= (\delta \boldsymbol{\rho}_2 - \delta \boldsymbol{\rho}_1)^T \tilde{\mathbf{e}}_1 & &= (\delta \boldsymbol{\rho}_1 - \delta \boldsymbol{\rho}_2)^T \tilde{\mathbf{e}}_2 \\ d \delta \varphi_1 &= (\delta \boldsymbol{\rho}_2 - \delta \boldsymbol{\rho}_1)^T \tilde{\mathbf{g}}_1 & d \delta \varphi_2 &= (\delta \boldsymbol{\rho}_2 - \delta \boldsymbol{\rho}_1)^T \tilde{\mathbf{g}}_1.\end{aligned}$$

6.4 Variationsformulierung und Linearisierung

Sei \mathbf{r}_j die Kontaktkraft. Unter den Annahmen, dass keine Momente übertragen werden und dass keine Reibung existiert, gilt

$$\mathbf{r}_j = N_j \tilde{\mathbf{e}}_j + T_j \boldsymbol{\tau}_j + M_j \tilde{\mathbf{g}}_j = N_j \tilde{\mathbf{e}}_j \quad j = 1, 2.$$

Wir nehmen außerdem an, dass sich die Balken nur punktweise berühren bzw. nur einmal je Paar von Balkenelementen. Die Koordinate der Kontaktstelle bezeichnen wir mit s_j^c ($j = 1, 2$). Damit nimmt die Kontaktkraft die Form einer Dirac-Delta-Funktion D an.

$$\mathbf{r}_j(s_j) = \mathbf{r}_j(s_j^c) D(s_j - s_j^c)$$

Wegen der Gleichgewichtsbedingung ist $\mathbf{r}_1 ds_1 + \mathbf{r}_2 ds_2 = \mathbf{0}$. Für die Variation des Potentials $\delta\Pi$ folgt dann

$$\begin{aligned}\delta\Pi &= \int_0^{L_1} \delta\boldsymbol{\varrho}_1^T \mathbf{r}_1 ds_1 + \int_0^{L_2} \delta\boldsymbol{\varrho}_2^T \mathbf{r}_2 ds_2 = \int_0^{L_1} (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \mathbf{r}_1 ds_1 \\ &= (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \mathbf{r}_1 \Big|_{s_1=s_1^c} = \epsilon_N (d - r_1 - r_2) (\delta\boldsymbol{\varrho}_2 - \delta\boldsymbol{\varrho}_1)^T \tilde{\mathbf{e}}_1 \Big|_{s_1=s_1^c} = N_1 \delta d \Big|_{s_1=s_1^c}\end{aligned}$$

Durch den nur punkweisen Kontakt fällt die Integration weg. Dadurch erübrigt sich auch das Problem, dass die Balken in allen Konfigurationen nach der Weglänge parametrisiert sein müssen. Dies gilt für die Balkentheorie aus Kapitel 3 nur in der Referenzkonfiguration.

Wir linearisieren im Folgenden wie bei den Balkenelementen mittels der Newton-Iteration. Dafür benötigen wir die zweite Variation $\Delta\delta\Pi$. Die Konstanten ϵ_N, r_1, r_2 besitzen keine Variation. Außerdem nehmen wir an, dass $\Delta\boldsymbol{\varrho}_j = \Delta\bar{\mathbf{u}}_j$ ($j = 1, 2$) unabhängig von der ersten Variation δ sind, vgl. die Variationsformulierung für Balkenelemente in Kapitel 3.7. Somit gilt $\Delta\delta\boldsymbol{\varrho}_j = 0$.

$$\begin{aligned}\Delta\delta\Pi &= \epsilon_N \Delta(d - r_1 - r_1) (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{e}}_1 + \epsilon_N (d - r_1 - r_1) (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \Delta\tilde{\mathbf{e}}_1 \\ &= \epsilon_N (\Delta\boldsymbol{\varrho}_1 - \Delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{e}}_1 (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{e}}_1 \\ &+ \epsilon_N (d - r_1 - r_1) \Delta\varphi_1 (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{g}}_1 \\ &\approx \epsilon_N (\Delta\boldsymbol{\varrho}_1 - \Delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{e}}_1 (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{e}}_1 \\ &+ \epsilon_N \left(1 - \frac{r_1 + r_2}{d}\right) (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{g}}_1 (\Delta\boldsymbol{\varrho}_1 - \Delta\boldsymbol{\varrho}_2)^T \tilde{\mathbf{g}}_1 \\ &= \epsilon_N (\Delta\boldsymbol{\varrho}_1 - \Delta\boldsymbol{\varrho}_2)^T \left(\tilde{\mathbf{e}}_1 \tilde{\mathbf{e}}_1^T + \left(1 - \frac{r_1 + r_2}{d}\right) \tilde{\mathbf{g}}_1 \tilde{\mathbf{g}}_1^T \right) (\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2)\end{aligned}$$

Da bei erfüllter Kontaktbedingung gilt $1 - \frac{r_1+r_2}{d} \approx 0$, ist der zweite Summand klein gegenüber dem Ersten. Er wird somit bei der Berechnung der Tangentialmatrix \mathbf{TANG}_{kont} ausgelassen. Die erste und zweite Variation bilden wiederum eine Linear- bzw. Bilinearform in $\delta\boldsymbol{\varrho}_1 - \delta\boldsymbol{\varrho}_2$ und $\Delta\boldsymbol{\varrho}_1 - \Delta\boldsymbol{\varrho}_2$.

6.5 Implementierung

Um den Algorithmus zur Berechnung des Kontakts implementieren zu können, muss ein Kontaktelement analog zu den Balkenelementen in Kapitel 5 implementiert werden.

6.5.1 Ansatzfunktionen

Die Ansatzfunktionen $A^{(i)}$ der Kontaktelemente entsprechen denen der zugehörigen Balkenelemente, da sonst der minimale Abstand nicht ermittelt werden kann. Da sich das Kontaktelement über zwei Balkenelemente erstreckt, sind auch alle Ansatzfunktionen doppelt zu definieren.

Sei nen die Zahl der Ansatzfunktionen der beiden Balkenelemente, dann definieren wir

$$\begin{aligned} A_{kontakt}^{(i)} &\equiv A_{balken1}^{(i)} & i = 1, \dots, nen \\ A_{kontakt}^{(i+nen)} &\equiv A_{balken2}^{(i)} & i = 1, \dots, nen \end{aligned}$$

Analog werden die Ansatzfunktionen $\partial A, A_{ref}, \partial A_{ref}$ definiert. Wir benötigen nun auch die zweiten Ableitungen und speichern diese in $\partial^2 A$ und $\partial^2 A_{ref}$ ab. Die Matrix $\mathbf{A} \in \mathbb{R}^{3 \times 2nen}$ hat nun die Form

$$\mathbf{A} = \begin{pmatrix} A^{(1)} & 0 & 0 & & A^{(nen)} & 0 & 0 \\ 0 & A^{(1)} & 0 & \dots & 0 & A^{(nen)} & 0 \\ 0 & 0 & A^{(1)} & & 0 & 0 & A^{(nen)} \\ -A^{(1+nen)} & 0 & 0 & & -A^{(2nen)} & 0 & 0 \\ 0 & -A^{(1+nen)} & 0 & \dots & 0 & -A^{(2nen)} & 0 \\ 0 & 0 & -A^{(1+nen)} & & 0 & 0 & -A^{(2nen)} \end{pmatrix}.$$

Analog wird die Matrix \mathbf{A}_{ref} definiert. Dabei werden die Ansatzfunktionen des zweiten Balkens negativ angesetzt. Somit kann, zusammen mit den Knotenwerten der Referenzkonfiguration $\mathbf{u}_{0balken1}^e, \mathbf{u}_{0balken2}^e$ und der aktuellen Konfiguration $\mathbf{u}^e_{balken1}, \mathbf{u}^e_{balken2}$ der beiden Balken, der häufig benötigte Abstand $\boldsymbol{\varrho}_1 - \boldsymbol{\varrho}_2$ als einfaches Matrix-Vektor-Produkt dargestellt werden.

$$\mathbf{A}_{ref} \begin{pmatrix} \mathbf{u}_{0balken1}^e \\ \mathbf{u}_{0balken2}^e \end{pmatrix} + \mathbf{A} \begin{pmatrix} \mathbf{u}_{balken1}^e \\ \mathbf{u}_{balken2}^e \end{pmatrix} \approx \boldsymbol{\varrho}_1 - \boldsymbol{\varrho}_2$$

6.5.2 Globale Suche durch Newton-Iteration

Wir suchen den kleinsten Abstand nun nicht mehr bzgl. der Bogenlängenparameter s_j , sondern bzgl. der Elementkoordinaten ξ_j . Als Anfangswert wählen wir jeweils die Elementmitte. Die Iterationsvorschrift der Newton-Iteration lautet dann

$$\begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix}_{k+1} = \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix}_k - \mathbf{H}^{-1} \mathbf{grad} F \quad k \in \mathbb{N}^+ \quad \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix}_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

Die Newton-Iteration wird abgebrochen, wenn $\|\mathbf{H}^{-1} \mathbf{grad} F\|$ klein ist oder eine maximale Anzahl an Iterationsschritten $ncpp$ (*number of closest point projection*) erreicht wurde.

6.5.3 Lokale Suche

Nachdem die beiden Koordinaten ξ_1^c und ξ_2^c für den minimalen Abstand gefunden sind, müssen wir überprüfen, ob diese innerhalb des Kontaktelements liegen. Falls $|\xi_j^c| > 1$ für $j = 1$ oder $j = 2$, liegt der ermittelte minimale Balkenabstand außerhalb des Elements. Es gibt entweder keinen Kontakt oder die Kontaktstelle liegt im Bereich eines anderen Elements.

Um Instabilitäten beim Übergang von einem Kontaktelement zum nächsten zu vermeiden, kann die lokale Suche angepasst werden. Wähle $\mu \geq 0$. Für $|\xi_j^c| < 1 - \mu$ findet Kontakt wie oben definiert statt. Bei $|\xi_j^c| > 1 + \mu$ ist kein Kontakt vorhanden. Im Übergangsbereich zwischen $1 - \mu$ und $1 + \mu$ passen wir zuerst den Strafparameter ϵ_N linear an.

$$\epsilon_N^{neu} = \begin{cases} \epsilon_N^{alt} & \text{falls } |\xi_j^c| < 1 - \mu \\ \epsilon_N^{alt} \left(\frac{1 - |\xi_j^c|}{2\mu} + \frac{1}{2} \right) & \text{falls } 1 - \mu \leq |\xi_j^c| \leq 1 + \mu \\ \text{kein Kontakt} & \text{falls } 1 + \mu < |\xi_j^c| \end{cases}$$

Allerdings sind die Ansatzfunktionen A im Intervall $1 < |\xi_j^c| \leq 1 + \mu$ nicht definiert. Deshalb werten wir nach Berechnung des neuen Strafparameters ϵ_N^{neu} die Ansatzfunktion A in diesem Fall an der Stelle $\xi_j^c = 1$ bzw. $\xi_j^c = -1$ aus.

6.5.4 Die Steifigkeitsmatrix $TANG_{kont}$

Seien $\begin{pmatrix} \xi_1^c \\ \xi_2^c \end{pmatrix} \in [-1, 1]^2$ die Koordinaten der Kontaktstelle. Durch die besondere Definition der Matrix \mathbf{A} der Ansatzfunktionen ergibt sich folgende einfache Berechnung der Steifigkeitsmatrix $TANG$. Da der Kontakt nur punktweise stattfindet, muss hier nicht aufintegriert bzw. aufsummiert werden.

$$TANG_{kont} = \epsilon_N \mathbf{A}^T (\tilde{\mathbf{e}}_1 \tilde{\mathbf{e}}_1^T) \mathbf{A} \Big|_{\xi_1 = \xi_1^c, \xi_2 = \xi_2^c}$$

Somit folgt die diskrete Approximation für die zweite Variation

$$\Delta \delta \Pi \approx \begin{pmatrix} \delta \mathbf{u}_{balken1}^e \\ \delta \mathbf{u}_{balken2}^e \end{pmatrix}^T TANG_{kont} \begin{pmatrix} \Delta \mathbf{u}_{balken1}^e \\ \Delta \mathbf{u}_{balken2}^e \end{pmatrix}.$$

Dabei ist zu beachten, dass keine infinitesimalen Verdrehungen $\delta \mathbf{w}^e$ in die Variation des Potentials eingehen. Beim Zusammenbau der globalen Steifigkeitsmatrix müssen die Anteile zu den Verschiebungsfreiheitsgraden der entsprechenden Knoten zugeordnet werden.

6.5.5 Das Residuum RES_{kont}

Wie die Steifigkeitsmatrix $TANG_{kont}$ definieren wir das Residuum RES_{kont} . Hier treten im Gegensatz zur Balkenformulierung keine äußeren Lasten auf.

$$RES_{kont} = -\epsilon_N p \mathbf{A}^T \tilde{\mathbf{e}}_1 \Big|_{\xi_1 = \xi_1^c, \xi_2 = \xi_2^c}$$

Die diskrete Approximation der ersten Variation des Potentials ist dann entsprechend

$$-\delta \Pi \approx \begin{pmatrix} \delta \mathbf{u}_{balken1}^e \\ \delta \mathbf{u}_{balken2}^e \end{pmatrix}^T RES_{kont}.$$

In Kapitel 5.5 sind weitere Hinweise zu Implementierung der Kontaktelemente gegeben.

Kapitel 7

Elementtests

In diesem Kapitel betrachten wir einige Anwendungsprobleme, um die im vorherigen Kapitel definierten Elemente zu verifizieren und zu bewerten. Ein Hauptaspekt ist der Vergleich der Approximationsgüte zwischen linearen Elementen und Elementen höherer Ordnung.

Im ersten Abschnitt werden die Gemeinsamkeiten und Unterschiede zur linearen Balkentheorie nach Euler-Bernoulli bzw. Timoshenko anhand eines eingespannten Balkens aufgezeigt. Das vollständige Aufrollen eines Balkens unter Momentenbelastung zum Ring, bzw. das Abrollen in entgegengesetzter Richtung ist Gegenstand des darauf folgenden Abschnitts. Die numerischen Ergebnisse können dabei mit der existierenden analytischen Lösung verglichen werden. In der Arbeit von Bathe [BB79] wird ein dreidimensionaler Test vorgestellt, welcher im dritten Abschnitt nachgerechnet wird. Außerdem betrachten wir eine aktuelle Verifikation zur Spannungsberechnung von gebördelten Blechen aus der Dissertation von Fleischer [Flegt]. Der letzte Teil behandelt Kontaktprobleme zwischen Balken, Ringen und Kabeln.

In den folgenden Tabellen bezeichnet ein Stern an der Zahl der Iterationsschritte, dass die Iteration erfolgreich konvergiert. Die Grafiken der undeformierten und deformierten Balken wurden mit MATLAB erstellt. Hierbei geben die Farbverläufe nicht die Spannungen im Balken wieder, sondern sind im Allgemeinen den Werten der e_3 -Achse zugeordnet.

7.1 Gemeinsamkeiten mit der linearen Balkentheorie

Um die korrekte Abbildung der linearen Balkentheorien von Euler-Bernoulli und Timoshenko zu überprüfen, betrachten wir den analytisch exakt lösbaren Fall eines eingespannten geraden Balkens mit Einzelmoment bzw. Einzelkraft am Balkenende. Die Modelle von Euler-Bernoulli und Timoshenko sind linear und entsprechen somit der ersten Näherungslösung der Newton-Iteration des in dieser Arbeit hergeleiteten nichtlinearen Modells. Außerdem berechnen wir durch Konvergenz des Newton-Verfahrens die nichtlineare Lösung und können somit den Modellierungsfehler der linearen Modelle veranschaulichen.

7.1.1 Eingespannter gerader Balken mit Einzelmoment

Problemstellung

Gegeben ist ein eingespannter, gerader Balken mit einer Länge von 1 m und einem Kreisquerschnitt mit 10 mm Radius. Die Materialparameter entsprechen in etwa jenen von Stahl (E-Modul 200 kN/mm^2 , Querkontraktion $\nu = 0,3$). Am freien Balkenende greift ein Biegemoment von $M = 1000\text{ Nm}$ an. Der Balken liegt auf der e_1 -Achse und das Moment M dreht um die e_3 -Achse. Das Problem ist somit nur zweidimensional.

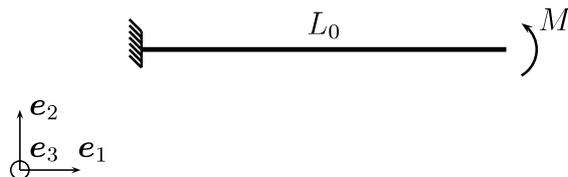


Abbildung 7.1: Aufbau des Tests “Eingespannter gerader Balken mit Einzelmoment”

Analytische Lösung, linear

Die Schnittkräfte sind alle identisch Null, außer dem Moment $M_3 = -1000\text{ kN/mm}$. Wir benötigen nur die Biegesteifigkeit $EI = 1570796\text{ kNmm}^2$. Durch die verschwindende Schubbelastung stimmen die analytischen Lösungen nach Euler-Bernoulli und Timoshenko überein. Aus [SGH02] wissen wir, dass $\frac{\partial^2}{\partial s_0^2} u_2(s_0) = -\frac{M_3}{EI}$. Zusammen mit den Randbedingungen folgt $u_2(s_0) = \frac{M}{2EI} s_0^2$ und $w_3(s_0) = \frac{M}{EI} s_0$. Am Balkenende ergibt sich somit eine Verschiebung $u_2 \approx 318,3\text{ mm}$ und eine Verdrehung $w_3 \approx 0,6366$. Der Balken beschreibt eine Parabel zweiter Ordnung.

Analytische Lösung, nichtlinear

Da keine Normalverzerrung im Balken auftritt, bleibt auch im nichtlinearen Fall die gleiche Verdrehung w_3 am Balkenende erhalten. Der Balken beschreibt aufgrund des konstanten Biegemoments einen Kreisbogen mit Radius $R \approx 1570,8\text{ mm}$. Am Balkenende ergibt sich somit eine vertikale Verschiebung $u_2 \approx 307,7\text{ mm}$ und eine horizontale Verschiebung $u_1 \approx -66,2\text{ mm}$.

Numerische Berechnung

Da die lineare analytische Lösung durch ein quadratisches Polynom beschrieben wird, müssen Elemente quadratischer oder höherer Ordnung die lineare Lösung exakt darstellen.

Wir diskretisieren somit den Balken durch ein einziges 3-Knoten Element (*elmt83*) mit quadratischen Ansatzfunktionen nach hierarchischem Ansatz. Daneben testen wir ein einzelnes 4-Knoten Element (*elmt84*), auch mit hierarchischem Ansatz. Außerdem wollen wir

Art der Berechnung				Verschiebung am Endknoten		
Elementtyp	Elemente	Lastschritte	Iterationen	u_1	u_2	$\tilde{w}_3 = w_3$
analytische Lösung nach Euler-Bernoulli				0	$3,18310e+2$	$6,36620e-1$
<i>elmt82</i>	1	1	1	0	$2,75630e-1$	$5,51259e-4$
volle Integration	100	1	1	0	$2,85381e+2$	$5,70763e-1$
<i>elmt83</i> , hierarchisch	1	1	1	0	$3,18310e+2$	$6,36620e-1$
volle Integration						
<i>elmt84</i> , hierarchisch	1	1	1	0	$3,18310e+2$	$6,36620e-1$
volle Integration						
<i>elmt82</i>	1	1	1	0	$3,18310e+2$	$6,36620e-1$
reduzierte Integration	100	1	1	0	$3,18310e+2$	$6,36620e-1$
<i>elmt83</i> , hierarchisch	1	1	1	0	$3,18310e+2$	$6,36620e-1$
reduzierte Integration						
<i>elmt84</i> , hierarchisch	1	1	1	0	$3,18310e+2$	$6,36620e-1$
reduzierte Integration						

Tabelle 7.1: Eingespannter gerader Balken mit Einzelmoment nach linearer Theorie

Art der Berechnung				Verschiebung am Endknoten		
Elementtyp	Elemente	Lastschritte	Iterationen	u_1	u_2	$\tilde{w}_3 = w_3$
analytische Lösung				$-6,61918e+1$	$3,07704e+2$	$6,36620e-1$
<i>elmt82</i>	1	1	3*	$-5,94138e-6$	$2,75630e-1$	$5,51259e-4$
volle Integration	100	1	39*	$-9,21861e+1$	$2,47821e+2$	$5,76255e-1$
<i>elmt83</i> , hierarchisch	1	1	23*	$-4,39257e+0$	$8,10805e+1$	$1,62518e-1$
volle Integration						
<i>elmt84</i> , hierarchisch	1	1	5*	$-6,58978e+1$	$3,07585e+2$	$6,36203e-1$
volle Integration						
<i>elmt82</i>	1	1	3*	$-5,02343e+1$	$3,12962e+2$	$6,36620e-1$
reduzierte Integration	100	1	3*	$-6,61902e+1$	$3,07704e+2$	$6,36620e-1$
<i>elmt83</i> , hierarchisch	1	1	3*	$-6,62278e+1$	$3,07692e+2$	$6,36620e-1$
reduzierte Integration						
<i>elmt84</i> , hierarchisch	1	1	3*	$-6,61918e+1$	$3,07704e+2$	$6,36620e-1$
reduzierte Integration						

Tabelle 7.2: Eingespannter gerader Balken mit Einzelmoment nach nichtlinearer Theorie

den Approximationsfehler durch lineare 2-Knoten Elemente (*elmt82*) betrachten. Hierfür diskretisieren wir den Balken mit einem sowie mit 100 Elementen. Wir variieren außerdem die Zahl der Gauß'schen Integrationspunkte und rechnen sowohl mit voller, als auch mit reduzierter Integration. Die Belastung wird in einem einzigen Lastschritt aufgetragen.

Analyse der numerischen Lösungen

Wie erwartet, stimmt die analytische Lösung nach Euler-Bernoulli mit der Lösung des 3-Knoten Elements sowie des 4-Knoten Elements nach dem ersten Iterationsschritt exakt überein. Die linearen Elemente sind bei voller Integration bei Weitem zu steif. Erst mit 100 Elementen stimmt die Größenordnung der Verschiebungen. Dieses Locking-Phänomen, dass Verschiebungen um Größenordnungen zu klein berechnet werden, wurde unter anderem auch bei Ibrahimbegovic [Ibr95] festgestellt. Durch Reduktion der Integrationsordnung erhalten wir auch bei linearen Elementen die exakte Lösung am Endknoten, der quadratische Verlauf des Balkens kann jedoch auch in diesem Fall nicht beschrieben werden.

Nun berechnen wir das Newton-Iterationsschema bis zur Konvergenz. Dabei wird lediglich der materielle Teil der tangentialen Steifigkeitsmatrix ausgewertet. Das Residuum wird

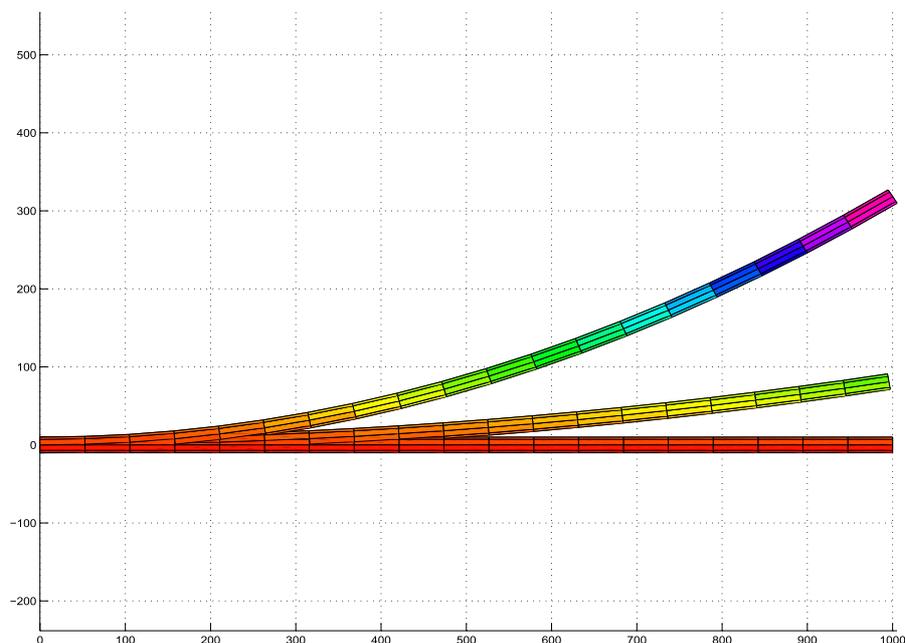


Abbildung 7.2: *elmt83* mit hierarchischem Ansatz, reduzierte Integration, 1 Element, Referenzgeometrie (unten), sowie Lösung nach 1 Schritt (oben) und 3 Schritten (Mitte)

über die erste Potentialvariation berechnet. Wie bereits erwähnt, bezeichnet ein Stern in Tabelle 7.1 die Konvergenz des Newton-Verfahrens. Dabei können wir feststellen, dass die lineare Theorie unter anderem die vorhandenen Verschiebungen in \mathbf{e}_1 -Richtung nicht abbildet. Bei voller Integration entsprechen nur die Ergebnisse des 4-Knoten Elements näherungsweise der analytischen Lösung. Bei reduzierter Integration verringert zum Einen die Zahl der Integrationsschritte erheblich auf jetzt konstant drei Schritte. Außerdem ist die Lösung von *elmt84* nun bis auf mindestens sechs Stellen exakt. Auch das Ergebnis von *elmt83* kommt nahe an die analytische Lösung heran. Bei *elmt82* tritt im nichtlinearen Fall jedoch ein großer Fehler auf.

7.1.2 Eingespannter gerader Balken mit Einzelkraft

Problemstellung

Gegeben ist ein eingespannter gerader Balken mit einer Länge von 10 cm . Ansonsten entspricht er dem vorhergehenden Test, d.h. der Kreisquerschnitt hat einen Radius $r = 10\text{ mm}$, der Elastizitätsmodul $E = 200\text{ kN/mm}^2$ und die Querkontraktion $\nu = 0,3$. Am Balkenende wirkt eine Kraft $F = 100\text{ kN}$ senkrecht zur Balkenlängsachse. Damit der Anteil der Schubverzerrungen nach der Timoshenko-Theorie größer ist und somit verifiziert werden kann, wurde die Balkenlänge entsprechend verkürzt. Der Balken liegt wieder auf der \mathbf{e}_1 -Achse. Die Kraft wirkt in Richtung der \mathbf{e}_2 -Achse. Das Problem bleibt damit zweidimensional.

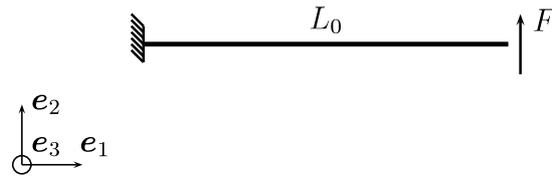


Abbildung 7.3: Aufbau des Tests “Eingespannter gerader Balken mit Einzelkraft”

Analytische Lösung, linear

Da bei diesem Beispiel auch Schubbelastung auftritt, berechnen wir die analytische Lösung nach der Theorie von Timoshenko. Zur Berechnung der Verschiebung des Kraftangriffspunktes, d.h. des Balkenendes, in Richtung der Kraft F , wenden wir den Satz von Castigliano an, vgl. [Bet04], Kapitel 6.3. Die Querkraft ist konstant $Q_2(s_0) = -F$. Das Biegemoment besitzt einen linearen Verlauf $M_3(s_0) = F(L_0 - s_0)$. Für die innere Arbeit gilt

$$\Pi_{int} = \int_0^{L_0} \frac{Q_2^2}{2GA_s} + \frac{M_3^2}{2EI} ds_0 = \frac{F^2 L_0}{2GA_s} + \frac{F^2 L_0^3}{6EI}.$$

Nach dem Satz von Castigliano folgt für die Verschiebung

$$u_2(L_0) = \frac{\partial}{\partial s_0} \Pi_{int} = \frac{FL_0}{GA_s} + \frac{FL_0^3}{3EI} \approx 21,7 \text{ mm}.$$

Die Verdrehung kann nach der Theorie von Euler-Bernoulli berechnet werden, da die Schubverzerrung keinen Einfluss auf die Verdrehung hat. Es gilt $w_3(L_0) \approx 0,3183$.

Numerische Berechnung

Wir verwenden die gleiche Diskretisierung wie beim vorherigen Balkentest unter Momentenbelastung. Nach der linearen Theorie von Timoshenko nimmt die Balkenachse die Form einer kubischen Parabel an, d.h. kubische 4-Knoten Elemente (*elmt84*) müssen die lineare analytische Lösung nach dem ersten Iterationsschritt exakt darstellen. Wir variieren weiterhin die Zahl der Integrationspunkte.

Analyse der numerischen Lösungen

Das 4-Knoten Element (*elmt84*) gibt die analytische Lösung nach der Timoshenko-Theorie exakt wieder. Bei reduzierter Integration sind auch die Ergebnisse von *elmt83* exakt. Die Verdrehung am Balkenende wird dann sogar vom 2-Knoten Element exakt angegeben.

Bei der nichtlinearen Berechnung steht nun keine analytische Lösung mehr zur Verfügung. Die Beobachtungen des ersten Tests in Kapitel 7.1.1 können jedoch bestätigt werden. Lineare Elemente sind bei vollständiger Integration zu steif. Eine Reduktion der Integrationsordnung verbessert die Ergebnisse und beschleunigt die Konvergenz, wenn auch weniger als im obigen Test. Bei einem 2-Knoten Element verschlechtert sich die Konvergenz sogar, was sich durch die nun größere Verschiebung erklärt.

Art der Berechnung				Verschiebung am Endknoten		
Elementtyp	Elemente	Lastschritte	Iterationen	u_1	u_2	$\tilde{w}_3 = w_3$
analytische Lösung nach Timoshenko				0	$2,16804e + 1$	$3,18310e - 1$
<i>elmt82</i>	1	1	1	0	$1,72911e + 0$	$2,53867e - 2$
volle Integration	100	1	1	0	$2,16555e + 1$	$3,17943e - 1$
<i>elmt83</i> , hierarchisch	1	1	1	0	$1,79792e + 1$	$3,18310e - 1$
volle Integration						
<i>elmt84</i> , hierarchisch	1	1	1	0	$2,16804e + 1$	$3,18310e - 1$
volle Integration						
<i>elmt82</i>	1	1	1	0	$1,63753e + 1$	$3,18310e - 1$
reduzierte Integration	100	1	1	0	$2,16799e + 1$	$3,18310e - 1$
<i>elmt83</i> , hierarchisch	1	1	1	0	$2,16804e + 1$	$3,18310e - 1$
reduzierte Integration						
<i>elmt84</i> , hierarchisch	1	1	1	0	$2,16804e + 1$	$3,18310e - 1$
reduzierte Integration						

Tabelle 7.3: Eingespannter gerader Balken mit Einzelkraft nach linearer Theorie

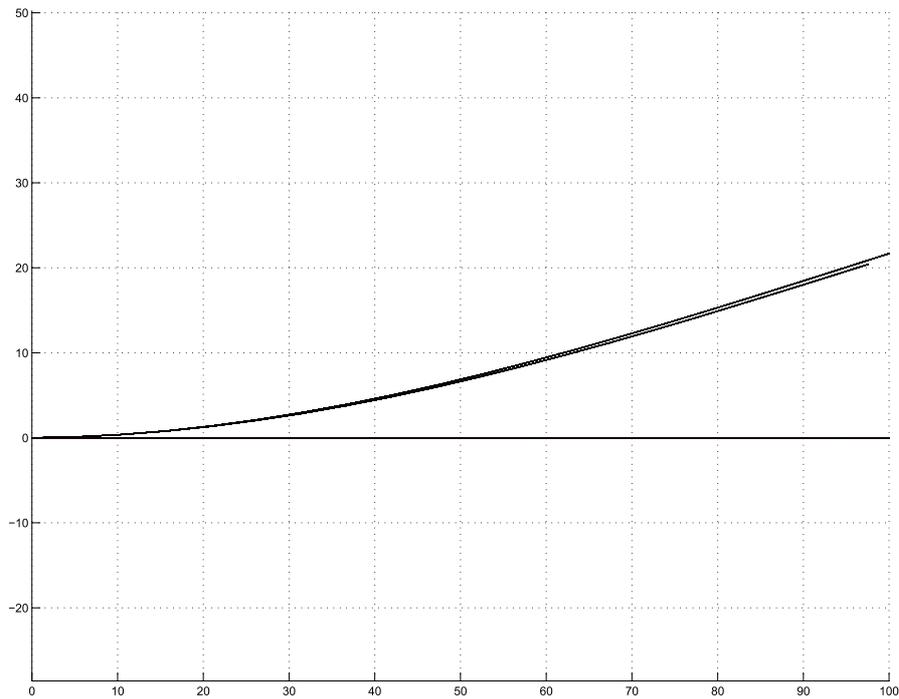


Abbildung 7.4: *elmt84* mit hierarchischem Ansatz, reduzierte Integration, 1 Element, Referenzgeometrie (unten), lineare Lösung nach einem Iterationsschritt (ganz oben) und Lösung nach 9 Iterationsschritten (oben)

Art der Berechnung				Verschiebung am Endknoten		
Elementtyp	Elemente	Lastschritte	Iterationen	u_1	u_2	$\tilde{w}_3 = w_3$
<i>elmt82</i>	1	1	4*	$-1,10417e-2$	$1,72879e+0$	$2,53826e-2$
volle Integration	100	1	9*	$-2,56216e+0$	$2,07243e+1$	$3,06786e-1$
<i>elmt83</i> , hierarchisch	1	1	11*	$-1,63250e+0$	$1,61924e+1$	$2,85511e-1$
volle Integration						
<i>elmt84</i> , hierarchisch	1	1	10*	$-2,48300e+0$	$2,04017e+1$	$3,03739e-1$
volle Integration						
<i>elmt82</i>	1	1	8*	$-1,27827e+0$	$1,60999e+1$	$3,14241e-1$
reduzierte Integration	100	1	9*	$-2,56761e+0$	$2,07457e+1$	$3,07116e-1$
<i>elmt83</i> , hierarchisch	1	1	9*	$-2,53964e+0$	$2,07356e+1$	$3,06869e-1$
reduzierte Integration						
<i>elmt84</i> , hierarchisch	1	1	9*	$-2,56778e+1$	$2,07453e+1$	$3,07114e-1$
reduzierte Integration						

Tabelle 7.4: Eingespannter gerader Balken mit Einzelkraft nach nichtlinearer Theorie

7.1.3 Eingespannter Viertelkreisbogen mit Einzelkraft

Der dritte Elementtest berücksichtigt nun auch die Approximation von gekrümmten Referenzgeometrien.

Problemstellung

Der gegebene Balken hat die Form eines Viertelkreises mit einem Radius R_0 von 100 mm . An einem Ende ist er fest eingespannt, am anderen Ende wirkt eine Kraft F von 100 kN in Richtung des Mittelpunktes des Kreisbogens. Die Materialparameter bleiben wie bei den vorhergehenden Tests. Der Balken liegt in der $\mathbf{e}_1, \mathbf{e}_2$ -Ebene und die Kraft F wirkt entgegengesetzt zur \mathbf{e}_1 -Achse. Das Problem ist wieder zweidimensional.

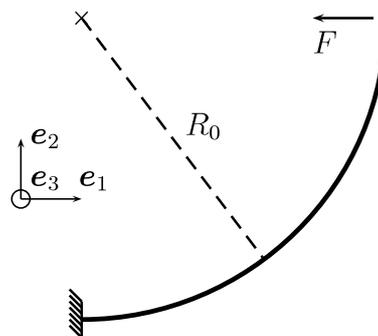


Abbildung 7.5: Aufbau des Tests ‘Eingespannter Viertelkreisbogen mit Einzelkraft’

Analytische Lösung, linear

Es ist möglich die Verschiebung des Balkenendes in \mathbf{e}_1 -Richtung nach dem linearen Modell von Timoshenko analytisch zu berechnen. Der Balken wird hierzu neben der Bogenlänge s_0 auch nach dem Winkel ψ parametrisiert. Es gilt $\psi = \frac{\pi}{2L_0}s_0 = \frac{1}{R_0}s_0$. Wir wenden

Art der Berechnung			Verschiebung am Endknoten		
Elementtyp	Referenzgeometrie	Iterationen	u_1	u_2	$\tilde{w}_3 = w_3$
lineare analytische Lösung nach Timoshenko			$-5,04861e + 1$		
<i>elmt82</i> , red. Integration	linear	1	$-2,29456e + 1$	$2,27205e + 1$	$4,50158e - 1$
<i>elmt83</i> , hierarchisch reduzierte Integration	linear	1	$-3,04482e + 1$	$3,02231e + 1$	$4,50158e - 1$
	Lagrange	1	$-4,94337e + 1$	$3,15119e + 1$	$6,36793e - 1$
	Kreis	1	$-4,97153e + 1$	$3,14041e + 1$	$6,39698e - 1$
<i>elmt84</i> , hierarchisch reduzierte Integration	rational	1	$-4,62851e + 1$	$3,05060e + 1$	$6,17534e - 1$
	linear	1	$-3,04482e + 1$	$3,02231e + 1$	$4,50158e - 1$
	Lagrange	1	$-5,06017e + 1$	$3,20122e + 1$	$6,37740e - 1$
	rational	1	$-5,05962e + 1$	$3,19300e + 1$	$6,37669e - 1$

Tabelle 7.5: Eingespannter Viertelkreisbogen mit Einzelkraft nach linearer Theorie

wieder den Satz von Castigliano an. Die Schnittkräfte in lokalen Koordinaten sind

$$N = F \cos \psi, \quad Q_2 = F \sin \psi, \quad M_3 = F R_0 \cos \psi.$$

Somit folgt für die innere Energie

$$\begin{aligned} \Pi_{int} &= \int_0^{L_0} \frac{N^2}{2EA} + \frac{Q_2^2}{2GA_s} + \frac{M_3^2}{2EI} ds_0 \\ &= \int_0^{\frac{\pi}{2}} \left(\frac{F^2 \cos^2 \psi}{2EA} + \frac{F^2 \sin^2 \psi}{2GA_s} + \frac{F^2 R_0^2 \cos^2 \psi}{2EI} \right) \underbrace{\frac{\partial s_0}{\partial \psi}}_{=R_0} d\psi \\ &= \frac{F^2 R_0}{2} \left(\frac{\pi}{4EA} + \frac{\pi}{4GA_s} + \frac{\pi R_0^2}{4EI} \right) \end{aligned}$$

und schließlich für die Verschiebung u_1 in Richtung der Kraft F

$$u_1 = \frac{\partial}{\partial F} \Pi_{int} = \frac{FR_0\pi}{4} \left(\frac{1}{EA} + \frac{1}{GA_s} + \frac{R_0^2}{EI} \right) \approx -50,5 \text{ mm}.$$

Numerische Berechnung

Ziel dieses Tests ist es, den Einfluss der Approximation der Referenzgeometrie zu bewerten. Aufgrund der Ergebnisse der vorherigen Tests rechnen wir nur noch mit reduzierter Integration. Wir diskretisieren den Viertelkreisbogen mit nur einem Element. Einerseits wird ein 2-Knoten Element (*elmt82*) verwendet, andererseits ein 3-Knoten Element (*elmt83*) mit linearer, polynomialer (Lagrange) und rationaler Geometrie. Daneben steht eine spezielle Kreisgeometrie zur Verfügung. Zuletzt verwenden wir das 4-Knoten Element (*elmt84*) mit linearer, polynomialer und rationaler Geometrie. Das $C1$ -Spline Element können wir an dieser Stelle nicht einsetzen, da hierfür der Balken mit mindestens 2 Elementen diskretisiert werden muss. Die Last wird in einem einzigen Schritt aufgetragen. Das Residuum wird nach der ersten Potentialvariation berechnet. Außerdem wird nur die materielle tangentielle Steifigkeitsmatrix \mathbf{TANG}_{mat} berücksichtigt.

Analyse der numerischen Ergebnisse

Bei der Analyse der Ergebnisse nach dem ersten Iterationsschritt in Tabelle 7.5 fällt auf, dass bei linearen Geometrien die Verschiebung u_1 zu gering angegeben wird. Dies liegt vor

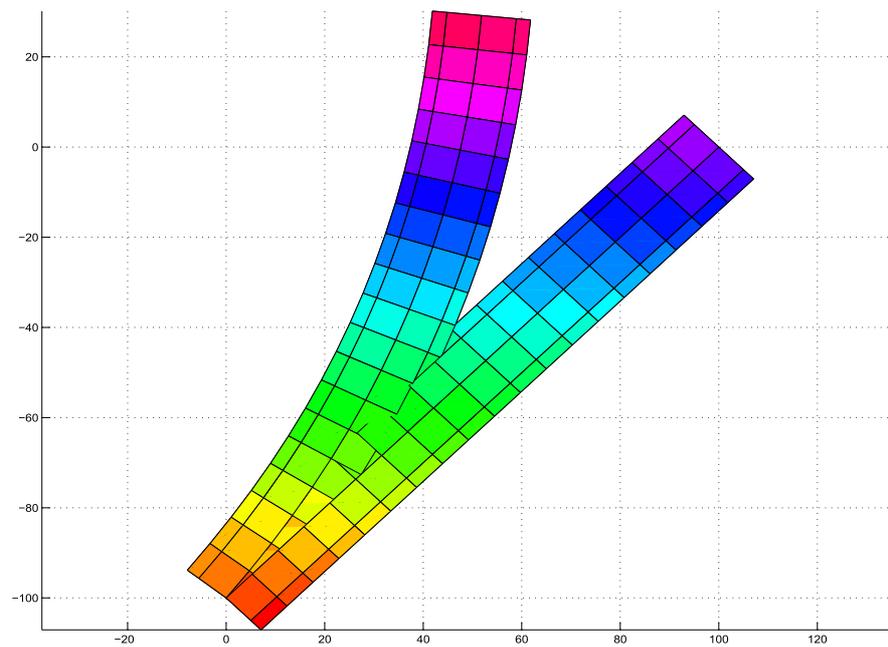


Abbildung 7.6: *elm83* mit linearer Geometrie, reduzierte Integration, 1 Element, Referenzgeometrie (rechts) und Lösung nach 12 Iterationsschritten (links)

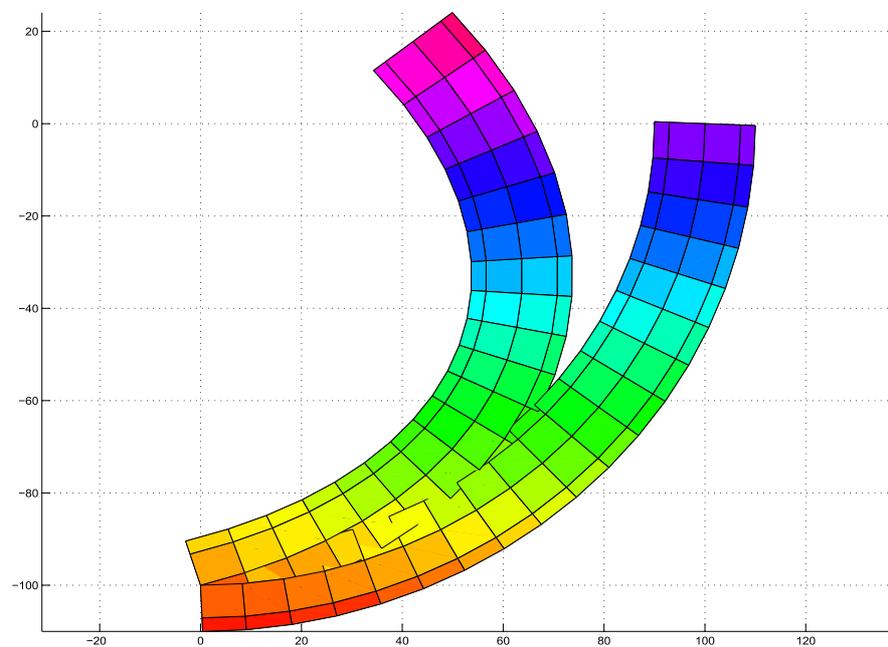


Abbildung 7.7: *elm83* mit Kreisgeometrie, reduzierte Integration, 1 Element, Referenzgeometrie (rechts) und Lösung nach 12 Iterationsschritten (links)

Art der Berechnung			Verschiebung am Endknoten		
Elementtyp	Referenzgeometrie	Iterationen	u_1	u_2	$\tilde{w}_3 = w_3$
<i>elmt82</i> , red. Integration	linear	11*	$-3,19224e + 1$	$2,38341e + 1$	$5,57449e - 1$
<i>elmt83</i> , hierarchisch reduzierte Integration	linear	12*	$-4,81751e + 1$	$2,91754e + 1$	$6,05381e - 1$
	Lagrange	12*	$-5,78714e + 1$	$1,80626e + 1$	$6,80647e - 1$
	Kreis	12*	$-5,78713e + 1$	$1,77908e + 1$	$6,80982e - 1$
<i>elmt84</i> , hierarchisch reduzierte Integration	rational	12*	$-5,30414e + 1$	$1,86750e + 1$	$6,48600e - 1$
	linear	12*	$-4,86526e + 1$	$2,91619e + 1$	$6,07731e - 1$
	Lagrange	12*	$-6,13057e + 1$	$1,87256e + 1$	$6,89677e - 1$
	rational	12*	$-6,09750e + 1$	$1,84119e + 1$	$6,89345e - 1$

Tabelle 7.6: Eingespannter Viertelkreisbogen mit Einzelkraft nach nichtlinearer Theorie

allein an der falsch approximierten Balkenlänge. Dabei liefern die Elemente *elmt83* und *elmt84* sogar die exakt gleichen Ergebnisse. Die Interpolation der Referenzgeometrie mittels Lagrange-Polynomen behebt einen Großteil des Fehlers, insbesondere im kubischen Fall (*elmt84*). Bei *elmt83* schneidet die spezielle Kreisgeometrie nochmals geringfügig besser ab. Der Hauptteil des Fehlers liegt jetzt offensichtlich in der Approximation der Verschiebung \mathbf{u} . Der Einsatz rationaler Funktionen zur exakten Geometriebeschreibung zahlt sich hingegen nicht aus. Dies folgt durch die ungleichmäßige Diskretisierung innerhalb des einzelnen Elements. Man kann diese an der Abhängigkeit der Jacobimatrix J vom Parameter ξ erkennen. Dieses Problem tritt bei *elmt83* eklatanter in Erscheinung, als bei *elmt84*.

Nach Konvergenz der Newton-Iteration bestätigen sich die obigen Beobachtungen des ersten Iterationsschrittes. Das lineare Element (*elmt82*) bleibt nun stärker hinter den Elementen höherer Ordnung mit linearer Geometrie zurück. Im Gegensatz zu den vorhergehenden Tests werden nun bis zu 12 Iterationsschritte benötigt.

7.1.4 Federmodell und Torsionsbelastung

Nun berechnen wir ein dreidimensionales Problem, das aufgrund der kleinen Deformationen noch gut mit der linearen Theorie übereinstimmt.

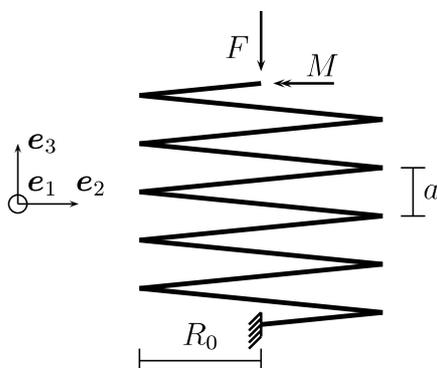


Abbildung 7.8: Aufbau des Tests “Federmodell und Torsionsbelastung”

Problemstellung

Gegeben ist ein Balken von 2,5 m Länge, der zu einer Feder mit 5 Windungen aufgewickelt ist. Der Radius des Balkenquerschnitts beträgt wie in den vorherigen Tests $r = 10 \text{ mm}$ und auch die Materialparameter bleiben gleich. Die Steigung entspricht $a = 50 \text{ mm}$ je Windung. Die Feder hat somit einen Durchmesser von 18 cm bei 27 cm Gesamthöhe. Sie wird zentral mit einer Vertikalkraft $F = -5 \text{ kN}$ zusammengedrückt. Das Problem ist dreidimensional, allerdings sind die Verschiebungen gering, sodass eine nicht allzu starke Abweichung der nichtlinearen Ergebnisse von jenen der linearen Theorie zu erwarten ist.

Analytische Lösung, linear

Ein Problem ist, dass die Last im Zentrum der Feder angreifen soll, dort jedoch keine Möglichkeit zur Lasteinwirkung besteht. Die Feder wird deshalb als eingespannter Balken modelliert, an dessen freiem Ende die Vertikallast F und ein Moment $M = R_0 F$ wirken. Bei einer Feder mit hoher Windungszahl wirken ausschließlich Querkraft und Torsionsmoment, die aufgrund der besonderen Geometrie konstant sind.

Der Radius der Feder ist $R_0 = \frac{\sqrt{L_0^2 - (5a)^2}}{10\pi} \approx 79 \text{ mm}$. Für die Querkraft gilt $Q_2(s_0) = F$ unter der Annahme, dass die Steigung der Feder klein ist. Das Torsionsmoment ist dann $M_T(s_0) = M = R_0 F$. Für die innere Energie ergibt sich folglich

$$\Pi_{int} = \int_0^{L_0} \frac{Q_2^2}{2GA_s} + \frac{M_T^2}{2GI_T} ds_0 = \int_0^{L_0} \frac{F^2}{2GA_s} + \frac{R_0^2 F^2}{2GI_T} ds_0 = \frac{L_0 F^2}{2} \left(\frac{1}{GA_s} + \frac{R_0^2}{GI_T} \right).$$

Nach dem Satz von Castigliano folgt für die Verschiebung u_{ztr} im Mittelpunkt der Feder

$$u_{ztr} = \frac{\partial}{\partial F} \Pi_{int} = \frac{L_0 F}{1} \left(\frac{1}{GA_s} + \frac{R_0^2}{GI_T} \right) \approx -65,4 \text{ mm}.$$

Dabei entfallen 64,9 mm auf Torsion und 0,5 mm auf Deformation durch Schub. Die Feder wird fast vollständig zusammengedrückt. Dies entspricht auch der Vertikalverschiebung u_3 am Balkenende. Am Endknoten findet keine Verschiebung in e_2 -Richtung statt.

Art der Berechnung				Verschiebung am Endknoten		
Elementtyp	Elemente	Lastschr.	Iterationen	u_1	u_2	u_3
lineare analytische Lösung nach Timoshenko					0	$-6,54305e + 1$
<i>elmt82</i> red. Integration	20	1	1 68*	$-5,65315e + 0$ $-2,43618e + 1$	$2,19953e - 11$ $-2,43257e - 1$	$-2,99603e + 1$ $-3,16231e + 1$
<i>elmt83</i> , hierarchisch Referenzgeom. Lagrange red. Integration	20	1	1 76*	$-7,96460e + 0$ $-3,72045e + 1$	$2,54612e - 11$ $-4,96955e - 1$	$-6,39094e + 1$ $-6,84955e + 1$
<i>elmt84</i> , hierarchisch Referenzgeom. Lagrange red. Integration	20	1	1 76*	$-8,00300e + 0$ $-3,73482e + 1$	$-3,16804e - 11$ $-5,29169e - 1$	$-6,54262e + 1$ $-7,00395e + 1$
<i>elmt84</i> , C1-Spline Referenzgeom. C1-Spline doppelt red. Integration	20 60	1 1	1 21* 1 60*	$-5,25682e + 0$ $-8,06486e + 0$ $-7,76837e + 0$ $-3,10294e + 1$	$-1,54462e + 0$ $-2,44273e + 0$ $3,88689e - 1$ $1,42742e + 0$	$-4,88930e + 1$ $-4,90462e + 1$ $-6,51624e + 1$ $-6,95096e + 1$

Tabelle 7.7: Federmodell unter zentraler Belastung

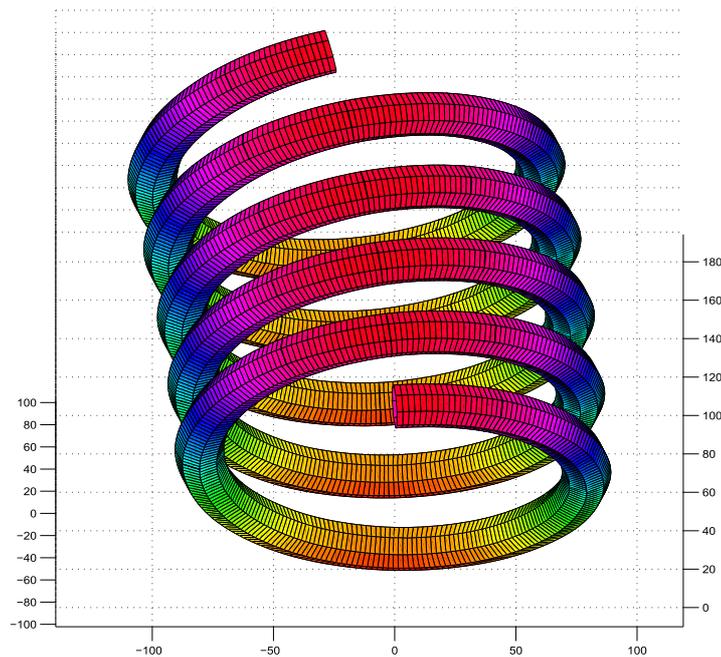


Abbildung 7.9: *elmt84* polynomialer Geometrie, reduzierte Integration, 20 Elemente, Lösung nach 76 Iterationsschritten

Numerische Berechnung

Wir diskretisieren die Feder mit vier Elementen je Windung, insgesamt also mit 20 Elementen. Die Berechnung erfolgt sowohl mit 2-Knoten Elementen, als auch mit 3- und 4-Knoten Elementen mit polynomialer Geometrie-Interpolation nach Lagrange. Die Integrationsordnung ist um einen Gauß-Punkt reduziert.

Daneben können wir $C1$ -Spline Elemente einsetzen. Zusätzlich zur Diskretisierung mit 20 Elementen verwenden wir eine Diskretisierung mit 60 Elementen, da dies ein Gleichungssystem der gleichen Größe wie beim Einsatz von 20 hierarchischen 4-Knoten Elementen ergibt. Wir rechnen dabei mit einer doppelt reduzierten Integrationsordnung, also nur noch mit zwei Integrationspunkten. Dies ergibt in diesem Fall die genauesten Ergebnisse.

Die Last wird in einem Schritt aufgetragen. Das Residuum wird, wie in den vorhergehenden Tests, über die erste Ableitung des Energiepotentials berechnet. Die tangentielle Steifigkeitsmatrix besteht nur aus dem materiellen Teil.

Analyse der numerischen Ergebnisse

Zuerst fällt auf, dass die Zahl der Iterationen trotz reduzierter Integration sehr hoch ist. Das lineare *elmt82* gibt nach dem ersten, linearen Iterationsschritt die Vertikalverschiebung über 50% zu gering an. Die 3-Knoten Elemente (*elmt83*) weichen leicht von der analytischen Lösung ab. Die kubischen 4-Knoten Elemente mit polynomialer Geometrie erfassen die analytische Lösung hingegen nahezu exakt. Bei den $C1$ -Spline Elementen

fällt auf, dass die feinere Diskretisierung einen sehr starken Einfluss auf die Genauigkeit des Ergebnisses hat.

Nach Berechnung der nichtlinearen Gleichgewichtslage erkennt man, dass die Vertikalverschiebung leicht zunimmt. Außerdem verstärkt sich das “Abkippen” der Feder in Richtung der negativen e_1 -Achse. Allerdings kann wegen der fehlenden analytischen Lösung keine genauere Analyse der Ergebnisse erfolgen.

7.2 Auf- und Abrollen von Balken

Das Aufrollen eines geraden Balkens zu einem geschlossenen Ring unter einem angreifenden Moment am Balkenende ist ein, in der Literatur häufig zu findendes, Beispiel. Die Arbeit von Simo [Sim86] sei hier exemplarisch genannt. Da die hier definierten Elemente, im Gegensatz zu jenen von Simo, auch gekrümmte Referenzgeometrien unterstützen, können wir wie Ibrahimbegovic [Ibr95] auch den inversen Fall eines Ringes, der zu einem geraden Balken zurückgebogen wird, berechnen. Durch den Vergleich der beiden Tests ist der, durch die Geometrie-Interpolation entstehende, Fehler sehr gut zu erkennen. Im letzten Abschnitt betrachten wir das Aufrollen eines Balkens unter einer kleinen Störbelastung, analog zu Ibrahimbegovic [IFK95]. Wir verlieren somit die Zweidimensionalität der beiden vorherigen Tests.

7.2.1 Aufrollen eines geraden Balkens

Problemstellung

Gegeben sei ein Balken der Länge $L_0 = 1$. Die Materialparameter sind $EA = GA_s = GI_t = EI = 1$. Diese Werte repräsentieren keinen realen Balken. Sie wurden analog zu [Sim86, Ibr95] gewählt, um numerische Instabilitäten zu vermeiden und eine Vergleichbarkeit der Ergebnisse sicherzustellen. Der Balken liegt auf der e_1 -Achse und das Moment M der Größe 2π am Balkenende dreht um die e_3 -Achse.

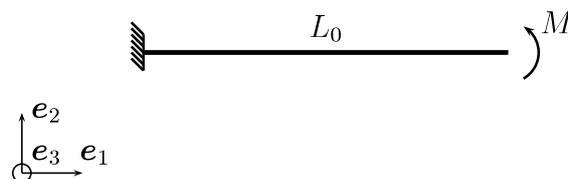


Abbildung 7.10: Aufbau des Tests “Aufrollen eines geraden Balkens”

Analytische Lösung, nichtlinear

Aus den Gleichgewichtsbedingungen erkennen wir, dass $M_3(s_0) \equiv 2\pi$. Mit Hilfe des Stoffgesetzes und der gegebenen Materialparameter folgt eine konstante Krümmung κ .

Art der Berechnung				Verschiebung am Endknoten		
Elementtyp	Elemente	Lastschritte	Iterationen	u_1	u_2	$w_3 = w_3$
analytische Lösung bei nichtlinearer Theorie				$-1,00000e + 0$	0	$6,28319e + 0$
lineare analytische Lösung nach Euler-Bernoulli				0	$3,14159e + 0$	$6,28319e + 0$
<i>elmt82</i> , red. Integration	4	1	3*	$-1,00000e + 0$	$1,06222e - 16$	$6,28319e + 0$
<i>elmt83</i> , hierarchisch reduzierte Integration	1	1	3*	$-7,59381e - 1$	$-2,25381e - 14$	$6,28319e + 0$
	2	1	3*	$-1,00000e + 0$	$-2,56010e - 16$	$6,28319e + 0$
<i>elmt84</i> , hierarchisch reduzierte Integration	4	1	3*	$-1,00000e + 0$	$-1,95976e - 16$	$6,28319e + 0$
	1	1	3*	$-1,02245e + 0$	$2,12817e - 15$	$6,28319e + 0$
<i>elmt84</i> , $C1$ -Spline doppelt red. Integration	2	1	3*	$-1,00000e + 0$	$1,72671e - 16$	$6,28319e + 0$
	4	1	3*	$-1,00000e + 0$	$-2,41977e - 16$	$6,28319e + 0$
<i>elmt84</i> , $C1$ -Spline doppelt red. Integration	4	1	7*	$-1,00345e + 0$	$3,21360e - 5$	$6,26453e + 0$
	12	1	5*	$-1,00001e + 0$	$2,80500e - 10$	$6,28309e + 0$
<i>elmt84</i> , hierarchisch reduzierte Integration keine Newton-Iteration	4	10	1	$-9,79490e - 1$	$-3,03084e - 2$	$6,28319e + 0$
		100	1	$-9,99894e - 1$	$-3,11367e - 4$	$6,28319e + 0$
		1000	1	$-9,99999e - 1$	$-3,13851e - 6$	$6,28319e + 0$

Tabelle 7.8: Aufrollen eines geraden Balkens mit Einzelmoment

Der deformierte Balken beschreibt somit einen Kreisbogen innerhalb der $\mathbf{e}_1, \mathbf{e}_2$ -Ebene. Wir berechnen die Verdrehung $w_3 = 2\pi$ wie im ersten Test, in Kapitel 7.1.1. Somit entsteht ein vollständiger Ring.

Numerische Berechnung

Der Balken wird mit vier Elementen diskretisiert. Wir vergleichen in diesem Test lineare 2-Knoten Elemente (*elmt82*), 3-Knoten Elemente (*elmt83*) mit quadratischen Ansatzfunktionen, 4-Knoten Elemente (*elmt84*) mit kubischen Ansatzfunktionen sowie $C1$ -Spline Elemente (*elmt84*). Bei *elmt83* und *elmt84* berechnen wir die Lösung außerdem mit einem und zwei Elementen, bei den $C1$ -Spline Elementen mit 12 Elementen. Die Integrationsordnung ist um einen Gauß-Punkt reduziert, bei den $C1$ -Spline Elementen um zwei Gauß-Punkte. Die Belastung wird dabei in einem Lastschritt aufgetragen.

Außerdem berechnen wir die Lösung ohne Newton-Iteration mit 10, 100 sowie 1000 inkrementellen Lastschritten.

Analyse der numerischen Ergebnisse

Die, in diesem Test auftretenden, Deformationen und Nichtlinearitäten sind extrem groß. Dies zeigt schon der Vergleich der geometrisch exakten, analytischen Lösung mit der linearen, analytischen Lösung. Dadurch werden insbesondere bei Diskretisierungen mit wenigen Elementen die höheren Ansatzfunktionen "bubble functions" stark angesprochen. Im Gegensatz zu den großen Nichtlinearitäten ist der Belastungszustand sehr einfach. Das Problem ist nur zweidimensional und außer dem konstanten Biegemoment M_3 treten keine weiteren Schnittkräfte auf.

Alle Ansätze, mit Ausnahme der $C1$ -Spline Elemente, konvergieren in drei Iterationsschritten. Die Diskretisierungen mit vier 2-Knoten Elementen, vier bzw. zwei hierarchischen 3-Knoten Elementen und vier bzw. zwei hierarchischen 4-Knoten Elementen sind bis auf Maschinengenauigkeit exakt. Das einzelne 4-Knoten Element weicht leicht von der analytischen Lösung ab. Das einzelne 3-Knoten Element um einiges stärker. Die Abbil-

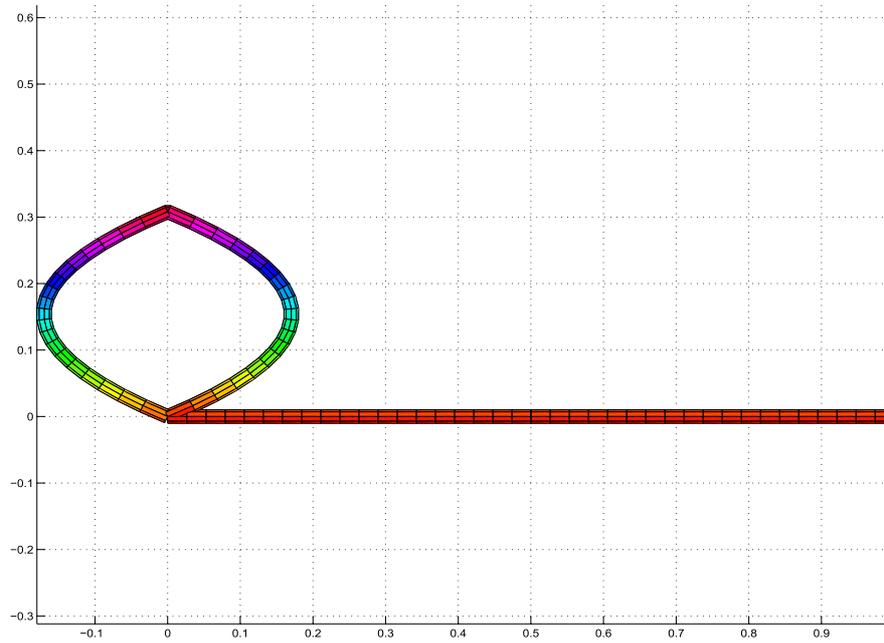


Abbildung 7.11: *elmt83* mit polynomialem Ansatz, 2 Elemente, Referenzgeometrie (rechts) und Lösung nach 3 Iterationsschritten (Ring)

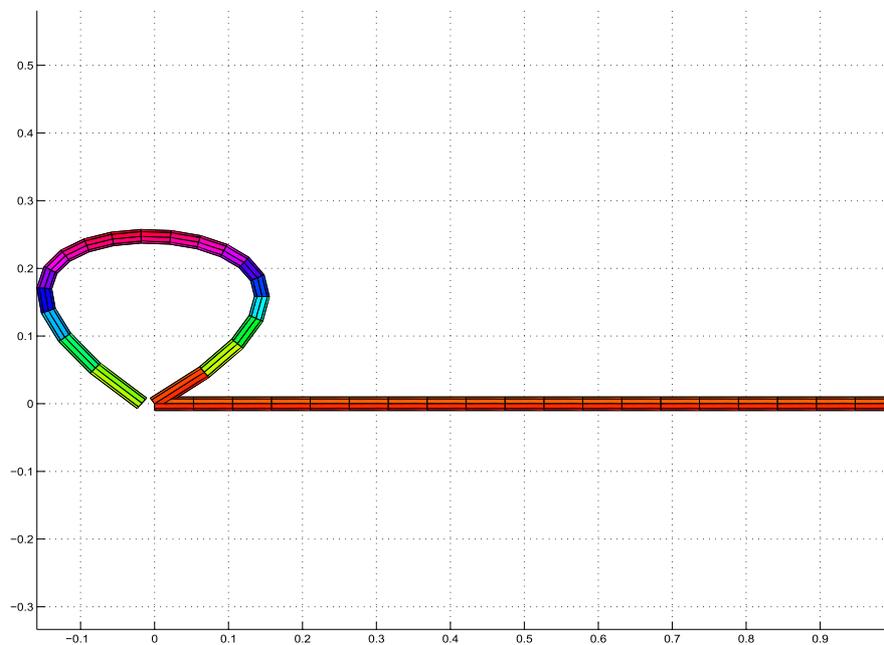


Abbildung 7.12: *elmt84* mit polynomialem Ansatz, 1 Element, Referenzgeometrie (rechts) und Lösung nach 3 Iterationsschritten (Ring)

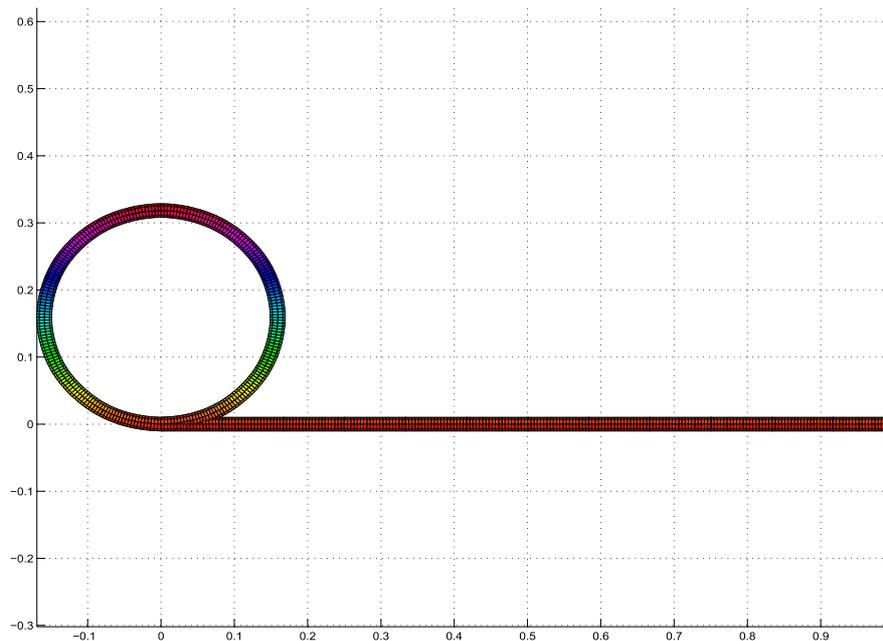


Abbildung 7.13: *elmt84* mit $C1$ -Spline Ansatz, 12 Elemente, Referenzgeometrie (rechts) und Lösung nach 5 Iterationsschritten (Ring)

dungen 7.11 und 7.12 zeigen jedoch, dass der Bogenverlauf trotz exakter Approximation des Endknotens stark von der Kreisform abweicht.

Die $C1$ -Spline Elemente konvergieren etwas langsamer. Außerdem wird die exakte Lösung nicht erreicht. Auch mit 12 Elementen ist noch ein, wenn auch sehr kleiner, Fehler zu erkennen. Dies folgt durch die nur approximative Berechnung der Verdrehung in den Knoten, insbesondere am ersten und letzten Element. Eine, in der Tabelle nicht aufgeführte, Berechnung mit nur einem einzigen Gauß-Punkt ergibt die, von den anderen Elementen bekannte, Konvergenz in drei Schritten. Allerdings weichen die Ergebnisse dann noch stärker von der exakten Lösung ab.

Zusätzlich wurde dieses Problem mit Hilfe kleiner Lastschritte ohne Newton-Iteration berechnet, vgl. Kapitel 4.3.2. Die Ergebnisse bleiben gegenüber der Newton-Iteration zurück. Außerdem steigt die Rechenzeit stark an, da das entsprechende lineare Gleichungssystem nicht nur in den drei Schritten der Newton-Iterationen gelöst werden muss, sondern in jedem Lastschritt, d.h. bis zu 1000 mal. Für dieses Problem eignet sich die Berechnung mittels kleiner Lastschritte somit nicht.

7.2.2 Abrollen eines Ringes

Problemstellung

Die Aufgabenstellung bei diesem Elementtest ist genau entgegengesetzt zum vorherigen Problem. Ein aufgeschnittener Ring ist an einem Ende eingespannt. Am anderen Ende

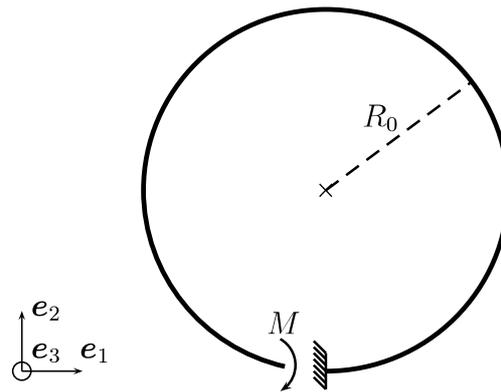


Abbildung 7.14: Aufbau des Tests "Abrollen eines Ringes"

wirkt ein Moment, das den Ring aufbiegt, bis sämtliche Krümmungen aufgehoben sind. Die Bogenlänge ist wiederum $L_0 = 1$, damit folgt für den Radius $R_0 = \frac{1}{2\pi}$. Die Materialparameter bleiben wie oben $EA = GA_s = GI_t = EI = 1$. Der Ring liegt in der e_1, e_2 -Ebene und das Moment der Größe 2π dreht nun negativ um die e_3 -Achse.

Analytische Lösung, nichtlinear

Entsprechend der analytischen Lösung beim Aufrollen eines Balkens folgt, dass der abgerollte Ring auf der e_1 -Achse zu liegen kommt. Die Länge bleibt konstant.

Numerische Berechnung

Wir diskretisieren den Ring mit vier Elementen. Aus dem Vergleich mit den Ergebnissen des vorhergehenden Tests können wir den Fehler der Geometrie-Approximation bestimmen.

Zur Approximation der Referenzgeometrie wenden wir zuerst die lineare Interpolation der Elementendknoten an, d.h. der Ring wird durch ein Viereck approximiert. Außerdem approximieren wir mittels quadratischen (*elmt83*) bzw. kubischen (*elmt84*) Polynomen. Daneben testen wir die exakte Beschreibung der Geometrie mit rationalen Funktionen sowie der speziellen Kreisgeometrie. Eine weitere Möglichkeit besteht darin, auch die Ansatzfunktionen für die Verschiebung durch rationale Funktionen zu ersetzen. Zuletzt testen wir $C1$ -Spline Elemente.

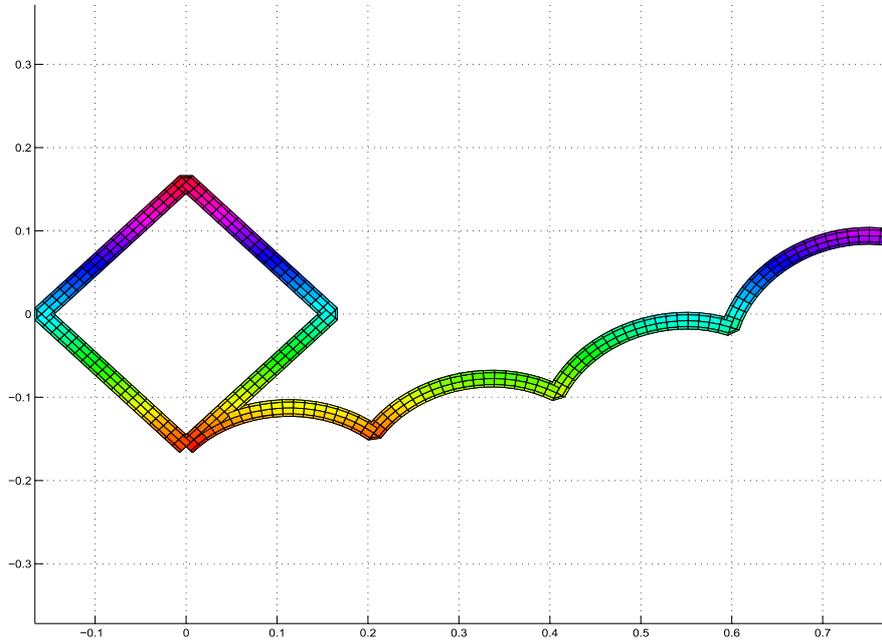
Die Berechnung erfolgt mit reduzierter Integrationsordnung, bei den $C1$ -Spline Elementen doppelt sowie dreifach reduziert. Es wird nur die materielle Steifigkeitsmatrix ausgewertet und die Last wird in einem Schritt aufgetragen.

Analyse der numerischen Ergebnisse

Die Ergebnisse der numerischen Berechnung weichen nun zum Teil um ein Vielfaches stärker von der analytischen Lösung ab. Insbesondere bei der linearen Interpolation der

Art der Berechnung			Verschiebung am Endknoten		
Elementtyp	Referenzgeometrie	Iterationen	u_1	u_2	$\tilde{w}_3 = w_3$
analytische Lösung			$1,00000e + 0$	0	$-6,28318e + 0$
<i>elmt82</i> , red. Integration	linear	3*	$8,43457e - 1$	$2,73130e - 1$	$-5,65685e + 0$
<i>elmt83</i> , hierarchisch reduzierte Integration	linear	3*	$7,74140e - 1$	$2,50683e - 1$	$-5,65685e + 0$
	Lagrange	3*	$9,97951e - 1$	$6,27911e - 3$	$-6,27060e + 0$
	Kreis	3*	$1,00000e + 0$	$-1,06080e - 15$	$-6,28319e + 0$
	rational	3*	$1,00177e + 0$	$1,48082e - 2$	$-6,29508e + 0$
<i>elmt83</i> , rational reduzierte Integration	rational	3*	$9,98871e - 1$	$4,05684e - 2$	$-6,28345e + 0$
<i>elmt84</i> , hierarchisch reduzierte Integration	linear	3*	$7,74909e - 1$	$2,50932e - 1$	$-5,65685e + 0$
	Lagrange	3*	$1,00085e + 0$	$2,69956e - 3$	$-6,28858e + 0$
	rational	3*	$9,99827e - 1$	$6,70243e - 4$	$-6,28214e + 0$
<i>elmt84</i> , rational reduzierte Integration	rational	3*	$1,00038e + 0$	$-3,93305e - 3$	$-6,28941e + 0$
<i>elmt84</i> , C^1 -Spline doppelt red. Integration	C^1 -Spline	7*	$8,51873e - 1$	$2,58636e - 1$	$-5,69366e + 0$
<i>elmt84</i> , C^1 -Spline dreifach red. Integration	C^1 -Spline	3*	$1,00035e + 0$	$-1,10454e - 3$	$-6,28539e + 0$

Tabelle 7.9: Abrollen eines ringförmigen Balkens mit Einzelmoment

Abbildung 7.15: *elmt84* mit hierarchischem Ansatz und linearer Geometrie, reduzierte Integration, 4 Elemente, Referenzgeometrie (Quadrat) und Lösung nach 3 Schritten (rechts)

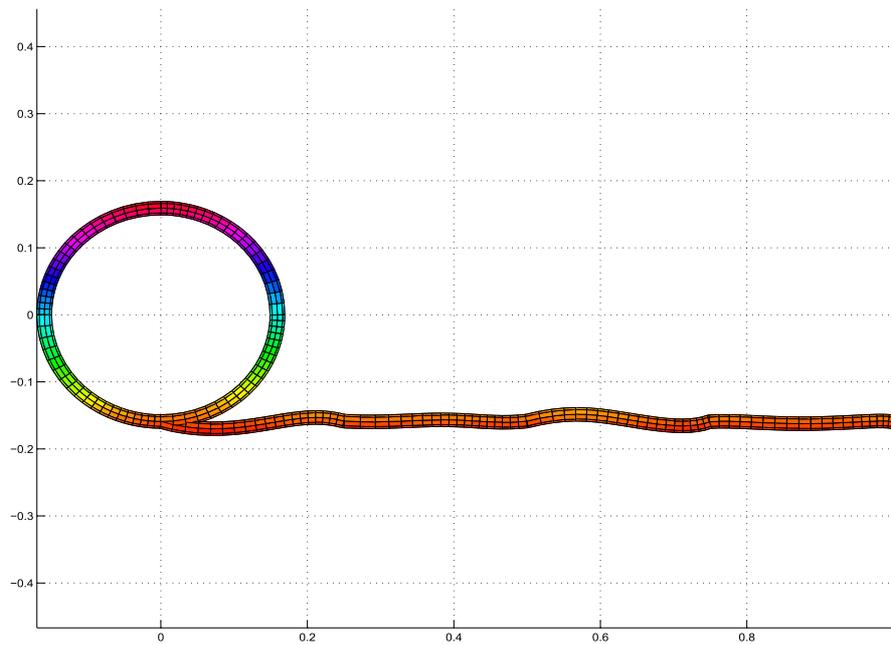


Abbildung 7.16: *elmt83* mit hierarchischem Ansatz und rationaler Geometrie, reduzierte Integration, 4 Elemente, Referenzgeometrie (Ring) und Lösung nach 3 Schritten (rechts)

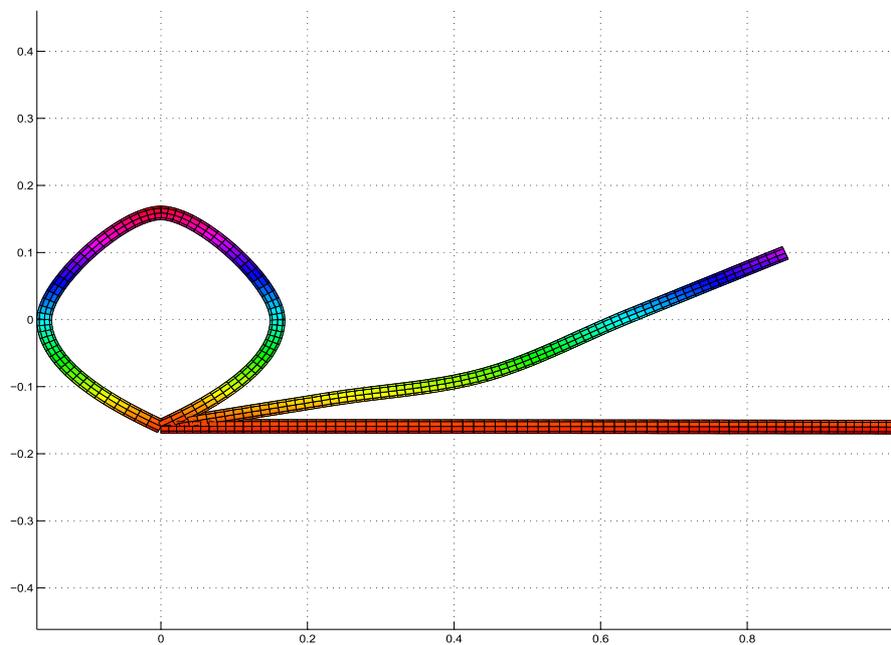


Abbildung 7.17: *elmt84* mit C_1 -Spline Ansatz und Geometrie, 4 Elemente, Referenzgeometrie (Ring), Lösung nach 7 Schritten bei doppelt reduzierter Integration (oben) und Lösung nach 3 Schritten bei dreifach reduzierter Integration (unten)

Referenzgeometrie sind große Fehler zu erkennen. Dies liegt vor allem an der falschen Berechnung der Balkenlänge $L_0 \approx 0,9003$, anstatt $L_0 = 1$, vgl. Abbildung 7.15.

Durch die Approximation der Referenzgeometrie mittels quadratischer (*elmt83*) bzw. kubischer (*elmt84*) Polynome kann der Fehler auf einfache Weise stark reduziert werden. Bei der exakten Beschreibung der Referenzkonfiguration mittels einer Kreisfunktion erhalten wir die exakte Lösung, wie beim Aufrollen eines geraden Balkens.

Eine weitere Möglichkeit die Kreisgeometrie exakt darzustellen besteht in der Beschreibung des Kreises mittels rationaler Funktionen. Allerdings verlieren wir so die Parametrisierung des Kreisbogens durch die Bogenlänge bzw. durch ein Vielfaches der Bogenlänge. Somit ist die Determinante der Jacobimatrix $\frac{\partial \xi}{\partial s_0}$ nicht länger konstant. Dies ruft zusätzliche Fehler und eventuell Instabilitäten hervor. Bei *elmt83* ist eine leichte Verschlechterung gegenüber der polynomialen Geometrie erkennbar, bei *elmt84* hingegen eine leichte Verbesserung.

Der Versuch, auch die Verschiebung mit rationalen Funktionen darzustellen, bringt keine Verbesserung. Bei Anwendungen stellt sich hier in besonderem Maße die Frage nach der Wahl der Gewichte der rationalen Basisfunktionen, da die zu beschreibende Verschiebung im Allgemeinen unbekannt ist.

Bei der Berechnung mit $C1$ -Spline Elementen erhalten wir bei doppelt reduzierter Integration Ergebnisse, die nur wenig besser sind als jene mit linearer Geometrie. Mit dreifach reduzierter Integration wird nahezu die analytische Lösung erreicht. Dieses Ergebnis kann jedoch nicht verallgemeinert werden, da dies nur bei konstanten Schnittkräften gilt. Die Werte, die in der Elementmitte ausgewertet werden, gelten somit für den ganzen Balken und die Integration ist exakt.

7.2.3 Aufrollen eines Balkens unter Störeinfluss

Problemstellung

Wir betrachten noch einmal den Fall des unter Momenteneinfluss aufzurollenden Balkens. Diesmal greift jedoch, zusätzlich zum Biegemoment, eine sehr kleine Einzelkraft $F = 0,01$ an der Balkenspitze an, die den Ring leicht aus der $\mathbf{e}_1, \mathbf{e}_2$ -Ebene heraus drückt. Die Kraft F weist in Richtung der negativen \mathbf{e}_3 -Achse. Das Problem ist somit dreidimensional, eine analytische Lösung ist nicht bekannt. Die exakte Lösung sollte jedoch nicht stark von der ungestörten Lösung abweichen, da die Kraft F klein ist.

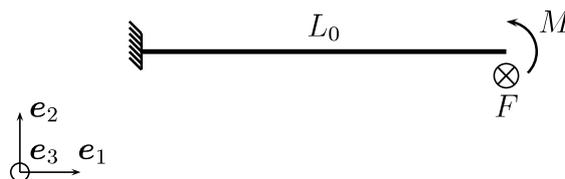


Abbildung 7.18: Aufbau des Tests “Aufrollen eines geraden Balkens unter Störeinfluss”

Art der Berechnung			Verschiebung am Endknoten			
Elementtyp	Lastschr.	Iterationen	u_1	u_2	u_3	\tilde{w}_3
<i>elmt82</i> red. Integration	1	126*	$-9,47316e-1$	$2,84949e-1$	$1,11116e-2$	$1,37768e+1$
	10	max. 119*	$-9,99348e-1$	$6,28925e-4$	$9,84556e-3$	$6,30214e+0$
	100	max. 116*	$-9,99986e-1$	$1,34021e-5$	$9,77818e-3$	$6,28361e+0$
<i>elmt83</i> hierarchisch red. Integration	1	36*	$-9,99590e-1$	$4,74026e-3$	$1,05746e-2$	$6,79481e+0$
	10	max. 34*	$-1,00000e+0$	$4,97835e-6$	$9,80547e-3$	$6,28406e+0$
	100	max. 34*	$-1,00000e+0$	$1,28457e-7$	$9,74952e-3$	$6,28322e+0$
<i>elmt84</i> hierarchisch red. Integration	1	34*	$-9,98913e-1$	$1,93996e-3$	$1,06079e-2$	$6,81229e+0$
	10	max. 32*	$-9,99999e-1$	$8,78001e-7$	$9,80807e-3$	$6,28408e+0$
	100	max. 31*	$-1,00000e+0$	$9,64618e-10$	$9,75213e-3$	$6,28322e+0$
<i>elmt84</i> <i>C1</i> -Spline red. Integration	1	184*	$-1,00351e+0$	$1,28554e-2$	$1,03935e-2$	$6,74820e+0$
	10	max. 202*	$-1,00325e+0$	$4,82453e-5$	$9,82892e-3$	$6,26437e+0$
	100	max. 197*	$-1,00324e+0$	$3,23202e-5$	$9,76640e-3$	$6,26358e+0$

Tabelle 7.10: Aufrollen eines geraden Balkens mit Einzelmoment und Störeinfluss

Numerische Berechnung

Wir wählen wieder eine Diskretisierung mit vier Elementen. Wir testen 2-Knoten Elemente (*elmt82*), 3-Knoten Elemente (*elmt83*), 4-Knoten Elemente (*elmt84*) und *C1*-Spline Elemente (*elmt84*). Die Integrationsordnung ist bei allen Elementen einfach reduziert. Auch bei den *C1*-Spline Elementen führt dies zu den besten Ergebnissen. Bei dreidimensionalen Problemen ist die Lösung im Allgemeinen von der Zahl der Lastschritte abhängig, vgl. Kapitel 4.3. Wir vergleichen die Ergebnisse bei Berechnungen mit 1, 10 und 100 Lastschritten.

Analyse der numerischen Ergebnisse

Bei der Analyse von Tabelle 7.10 fällt auf, dass die Zahl der benötigten Newton-Iterationen stark ansteigt, auf über 30 bis hin zu 200 Iterationsschritten. Auch bei einer Verkleinerung der einzelnen Lastschritte ist keine bessere Konvergenz zu beobachten. Lediglich die ersten Lastschritte konvergieren schnell, die letzten Lastschritte benötigen unabhängig von ihrer Größe etwa gleich viele Iterationen. Das Hauptproblem für die Konvergenz liegt somit nicht in der Größe der Lastschritte, sondern in der ungenauen Berechnung der tangentialen Steifigkeitsmatrix bei großen Deformationen. Auch durch die Zunahme des geometrischen Anteils \mathbf{TANG}_{geo} kann keine Verbesserung erzielt werden.

Die Zahl der Lastschritte beeinflusst die Lösung, jedoch nicht allzu stark. Insbesondere ist zu erkennen, dass die Lösungen mit 10 Lastschritten und 100 Lastschritten nahezu identisch sind.

Die Verdrehung des Balkenendes w_3 kann in der Tabelle nicht mehr direkt abgelesen werden, da $\tilde{w}_3 \neq w_3$. Dies ist insbesondere in der ersten Zeile bei *elmt82* zu erkennen. Nach \tilde{w}_3 müssten die Querschnitte zwei volle Drehungen vollziehen. Der Versuch, die Einzelkraft auf $F = 0,1$ zu erhöhen, schlägt bei manchen Diskretisierungen und Lastschrittweiten fehl. Die Newton-Iteration konvergiert dann nicht mehr. Aus Gründen der Übersichtlichkeit sind diese Lösungen jedoch nicht in der Tabelle aufgeführt.

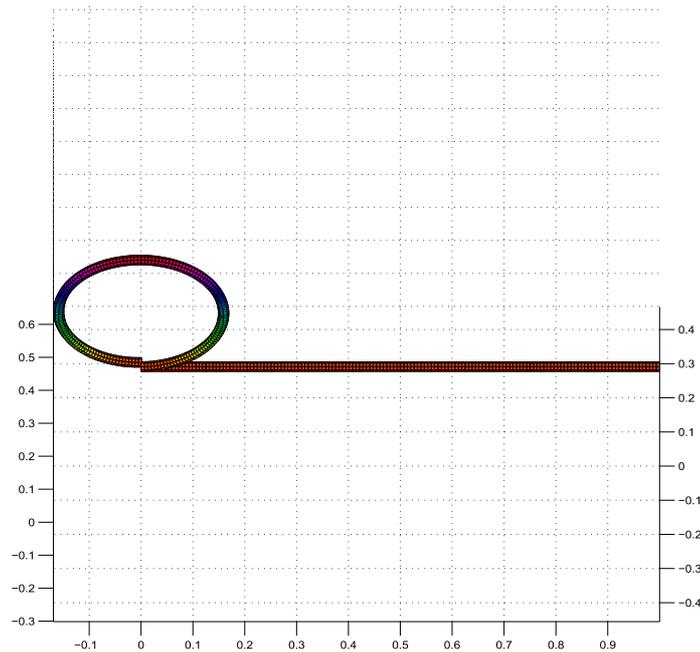


Abbildung 7.19: *elmt84* mit hierarchischem Ansatz, 8 Elemente, reduzierte Integration, 1 Lastschritt, Referenzgeometrie (rechts) und Endkonfiguration (Ring)

7.3 Ein dreidimensionaler Test nach Bathe

Die im Folgenden behandelte Testkonfiguration wurde zuerst von Bathe [BB79] präsentiert. Sie ist in vielen Arbeiten, wie beispielsweise [Sim86, Ibr95, IFK95, IT02] analysiert worden. Somit stehen, trotz nicht bekannter analytischer Lösung, verschiedene Vergleichslösungen zur Verfügung.

Problemstellung

Gegeben ist ein 45° Kreisbogen mit Radius $R_0 = 100$ der in der $\mathbf{e}_1, \mathbf{e}_2$ -Ebene liegt. Dies ist in Abbildung 7.20 illustriert. Der Bogen ist an einem Ende eingespannt. Am anderen Ende wirkt eine Einzelkraft $F = 600$ in \mathbf{e}_3 -Richtung. Der Querschnitt ist quadratisch mit Seitenlänge $a = 1$, der Elastizitätsmodul $E = 10^7$ und die Querkontraktion $\nu = 0$.

Numerische Berechnung

Um eine Vergleichbarkeit mit Bathe, Simo und Ibrahimbegovic zu gewährleisten, die den Bogen mit jeweils 8 Elementen diskretisiert haben, wählen wir die gleiche Zahl an Elementen. Wir testen 2-Knoten Elemente (*elmt82*), 3-Knoten Elemente (*elmt83*) mit Kreisgeometrie, polynomiale 4-Knoten Elemente (*elmt84*) und $C1$ -Spline Elemente (*elmt84*). Die Last wird in 10 Schritten aufgetragen. Die Integrationsordnung ist einfach reduziert, bei den $C1$ -Spline Elementen dreifach.

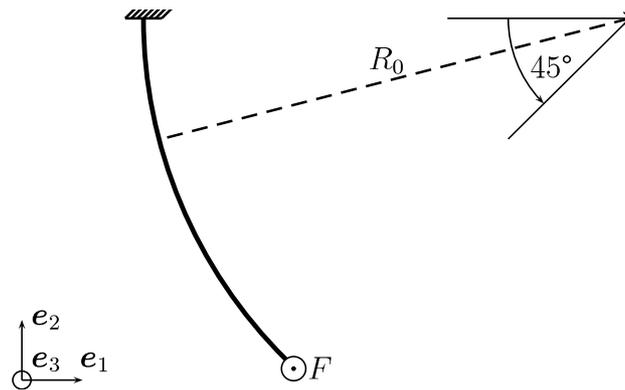


Abbildung 7.20: Aufbau des Tests nach Bathe

Art der Berechnung				Verschiebung am Endknoten		
Elementtyp	Elemente	Lastschr.	Iterationen	u_1	u_2	u_3
<i>elmt82</i> , red. Integration	8	10	max. 60*	$1,36091e+1$	$-2,34126e+1$	$5,33946e+1$
<i>elmt83</i> , hierarchisch Referenzgeom. Kreis reduzierte Integration	8	10	max. 62*	$1,37302e+1$	$-2,34923e+1$	$5,35003e+1$
<i>elmt84</i> , hierarchisch Referenzgeom. Lagrange reduzierte Integration	8	10	max. 62*	$1,37304e+1$	$-2,34925e+1$	$5,35008e+1$
<i>elmt84</i> , $C1$ -Spline Referenzgeom. $C1$ -Spline dreifach red. Integration	8	10	max. 63*	$1,38114e+1$	$-2,36220e+1$	$5,38431e+1$
Bathe, [BB79]	8	60		$1,34e+1$	$-2,35e+1$	$5,34e+1$
Simo, [Sim86]	8	3		$1,351e+1$	$-2,347e+1$	$5,337e+1$
Ibrahimbegovic, [Ibr95]	8	6		$1,3601e+1$	$-2,3764e+1$	$5,3407e+1$
Ibrahimbegovic, [IFK95]	8	6		$1,3668e+1$	$-2,3697e+1$	$5,3498e+1$
Ibrahimbegovic, [IT02]	8	10		$1,3601e+1$	$-2,3557e+1$	$5,3467e+1$

Tabelle 7.11: Eingespannter Balken nach Bathe, Variation der Ansätze und Vergleich mit Literaturwerten

In einem zweiten Test verringern wir die Zahl der Lastschritte, um den Einfluss dieses Parameters zu ermitteln. Außerdem führen wir Berechnungen mit kleinen Lastschritten, jedoch ohne Newton-Iteration durch.

Analyse der numerischen Ergebnisse

Die Verschiebungen in Tabelle 7.11 bestätigen die Ergebnisse von Bathe, Simo und Ibrahimbegovic. Die Werte der Verschiebung u_1 sind leicht erhöht. Außerdem ist das $C1$ -Spline Element etwas weicher im Vergleich zu den anderen Ansätzen. Es ist jedoch zu erkennen, dass eine sehr hohe Anzahl an Newton-Iterationen benötigt wird. Dies steht im Widerspruch zu Beobachtungen in [IFK95]. Die quadratische Konvergenz der Newton-Iteration wird bei Weitem nicht erreicht. Außerdem muss der Lösungsalgorithmus angepasst werden, um überhaupt ein konvergentes Verfahren zu erhalten. In FEAP steht hierzu der Befehl "*solv,line*" zur Verfügung. Wenn die Energienorm des Residuums von einem zum nächsten Iterationsschritt ansteigt, wird eine "*line search*" Prozedur ausgeführt, um die Energienorm zu verringern und Konvergenz der Newton-Iteration zu erreichen. Näheres hierzu ist beispielsweise in [Tay08] zu finden. Leider fehlt dort eine detaillierte

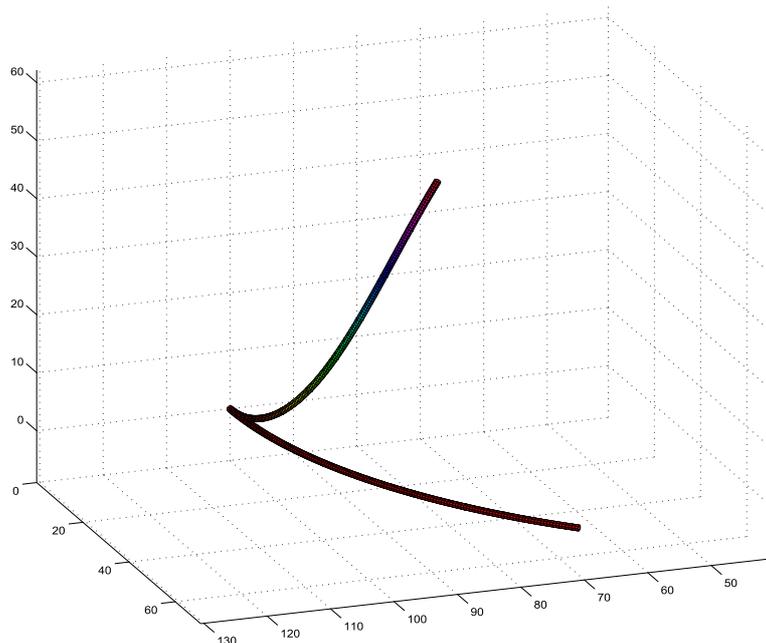


Abbildung 7.21: *elmt84* mit polynomialem Ansatz und Geometrie, 8 Elemente, 10 Lastschritte, Referenzgeometrie (unten) und Endkonfiguration (oben)

Art der Berechnung			Verschiebung am Endknoten		
Elementtyp	Lastschr.	Iterationen	u_1	u_2	u_3
<i>elmt84</i> , hierarchisch	1	43*	$1,50088e+1$	$-2,28309e+1$	$5,37168e+1$
Referenzgeom. Lagrange	10	max. 62*	$1,37304e+1$	$-2,34925e+1$	$5,35008e+1$
reduzierte Integration	100	max. 135*	$1,36165e+1$	$-2,35541e+1$	$5,34779e+1$
8 Elemente	10	1	$1,41371e+1$	$-2,46232e+1$	$5,43345e+1$
	100	1	$1,36568e+1$	$-2,36655e+1$	$5,35565e+1$

Tabelle 7.12: Eingespannter Balken nach Bathe, Variation der Lastschritte und Iterationen

Ausführung der zu Grunde liegenden Mathematik. Neu ist, dass im Gegensatz zu den bisherigen Tests die Zahl der benötigten Iterationen nicht von Lastschritt zu Lastschritt ansteigt. Das Maximum der Iterationschritte wird etwa bei halber Belastung, d.h. im fünften Lastschritt erreicht.

Tabelle 7.12 verdeutlicht den Einfluss der Anzahl der Lastschritte. Mit 100 Schritten liegen die Ergebnisse noch näher an den Literaturwerten, allerdings ist der Unterschied zur Lösung mit 10 Schritten recht klein. Die Differenz der Ergebnisse bei 1 und 10 Lastschritten ist bedeutend größer, sodass eine Berechnung mit etwa 10 Lastschritten sinnvoll erscheint. Die Zahl der benötigten Newton-Iterationen steigt bei Verkleinerung der Lastschritte sogar an.

Daneben wurde die Verschiebung des Balkens ohne Newton-Iteration berechnet. Insbesondere bei 100 Lastschritten weichen die Ergebnisse wenig von jenen mit Newton-Iteration ab. Ein Versuch mit 1000 Lastschritten schlug aufgrund fehlender Konvergenz fehl. Der Grund ist vermutlich, dass aufgrund der fehlenden Newton-Iteration keine Energiemini-

mierung des Residuums “*line search*” möglich ist. Da sich die Rechenzeit durch Ausparung der Newton-Iterationen stark verringert, bietet diese Methode, insbesondere bei Problemen mit schlechtem Konvergenzverhalten, eine Alternative.

Zusammenfassend lässt sich sagen, dass die aus der Literatur bekannten Ergebnisse reproduziert werden, wenn auch die Konvergenz schlechter ist. Die Zahl der Lastschritte hat für Werte über 10 nur noch sehr begrenzten Einfluss auf das Ergebnis.

7.4 Virtueller Ringzugversuch

Dieser Test ist der Dissertation von Fleischer [Flegt] entnommen. Die Arbeit beinhaltet unter anderem eine Verifikation zur Verwendung von Schalenelementen bei der Berechnung von Blechbördeln, d.h. Abkantungen von Blechen mit großen Deformationen. Das Simulationsmodell wurde dabei mit einer geringen Zahl von Schalenelementen in dem kommerziellen Finite Elemente Programm LS-DYNA berechnet. Gleichzeitig wurden die Ergebnisse mit einer Referenzlösung verglichen, die mittels einer großen Zahl an Kontinuums-elementen bestimmt wurde. Das Ergebnis dieser Arbeit lautet, dass unter der Annahme von elastisch-plastischem Materialverhalten bei hinreichend geringer Blechdicke einige Elementroutinen existieren, die das Verhalten bis zu der vorgegebenen Verformung nahezu korrekt darstellen. Da die im Zuge dieser Arbeit implementierten Elementroutinen nur linear-elastisches Materialverhalten wiedergeben, vergleichen wir an dieser Stelle die Ergebnisse der in [Flegt] verifizierten Schalenelemente mit den hier implementierten linearen, quadratischen und kubischen Balkenelementen unter der Annahme von linear-elastischem Material.

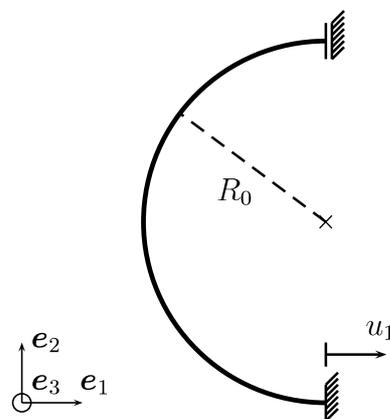


Abbildung 7.22: Aufbau des virtuellen Ringzugversuchs

Problemstellung

Das Problem ist dreidimensional, jedoch in der dritten Dimension konstant. Somit kann es auf die Einheitslänge 1 mm reduziert werden. Gegeben ist ein Ring mit Radius $R_0 = 5\text{ mm}$, der am unteren Ende aufgetrennt ist. Die Materialdicke ist 1 mm . Das linke Ende

wird mit einer Wegerandbedingung nach rechts verschoben, das Rechte nach links, sodass sich der Ring zusammenzieht. Der, in der Realität auftretende, Kontakt zwischen den beiden Ringhälften wird nicht modelliert. Da das Problem symmetrisch ist, muss nur eine Hälfte berechnet werden.

Das obere Ende ist in e_1 -Richtung fixiert. Aus Symmetriegründen die Verdrehungen dort gleich 0, in e_2 -Richtung kann sich das obere Ende jedoch frei verschieben. Das untere Ende ist fest eingespannt und wird in 142 Schritten um insgesamt $u_1 = 14,2 \text{ mm}$ in e_1 -Richtung verschoben. Es handelt sich somit um ein Dirichlet-Problem. Die Werte der Materialparameter sind $E = 195 \text{ kN/mm}^2$ und $\nu = 0,33$. Die Querschnittsfläche ist quadratisch mit 1 mm Seitenlänge.

Numerische Berechnung

In [Flegt] wurden 16 Schalenelemente in Längsrichtung verwendet. Wir berechnen das Ergebnis sowohl mit 16 linearen Elementen (*elmt82*) und quadratischen Elementen mit Kreisgeometrie (*elmt83*). Außerdem verwenden wir kubische Elemente mit polynomialer Geometrie (*elmt84*) und $C1$ -Spline Elemente (*elmt84*). Da bei den $C1$ -Spline Elementen die Verdrehung am letzten Element nur approximativ bestimmt werden kann und bei diesem Element die größten Deformationen auftreten, berechnen wir die Lösung auch mit 32 Elementen am ganzen Ring, ohne Ausnutzung der Symmetrie.

Da sich herausgestellt hat, dass die Zahl der Lastschritte geringen Einfluss hat, rechnen wir nur mit einem Viertel der Schritte der Vergleichslösung. Das Residuum berechnen wir über die erste Variation des Potentials.

Die horizontale Lagerkraft wird am letzten, nahezu ungekrümmten Element berechnet. Hierzu wird die, an den Gaußpunkten ohnehin berechnete, Kraft in e_1 -Richtung gewichtet, aufsummiert und ausgegeben.

Für die Berechnung des Radius nutzen wir das an der Symmetrieachse liegende Element. Dieses werten wir an der Stelle $\xi = -1$ direkt an der Symmetrieachse und bei $\xi = 0$ in der Elementmitte aus. Mit Hilfe dieser beiden Punkte und unter Ausnutzung der Symmetrie kann der Radius berechnet werden.

Elementtyp	Art der Berechnung			Ergebnisse		
	Elemente	Lastschritte	Iterationen	u_1	r	F
<i>elmt82</i> reduzierte Integration	16	142	max. 21*	$1,42e+1$	$6,31778e-1$	$1,16628e+1$
<i>elmt83</i> , hierarchisch Referenzgeom. Kreis reduzierte Integration	16	142	max. 21*	$1,42e+1$	$5,23965e-1$	$1,04363e+1$
<i>elmt84</i> , hierarchisch Referenzgeom. Lagrange reduzierte Integration	16	142	max. 21*	$1,42e+1$	$5,02396e-1$	$1,04172e+1$
<i>elmt84</i> , $C1$ -Spline Referenzgeom. $C1$ -Spline doppelt red. Integration	16	142	max. 21*	$1,42e+1$	$4,00197e-1$	$1,21697e+1$
	32 (ganzer Ring diskretisiert)	142	max. 21*	$1,42e+1$	$3,15888e-1$	$1,11662e+1$
Fleischer, [Flegt], LS-DYNA	16	568		$1,42e+1$		$1,2434e+1$

Tabelle 7.13: Virtueller Ringzugversuch

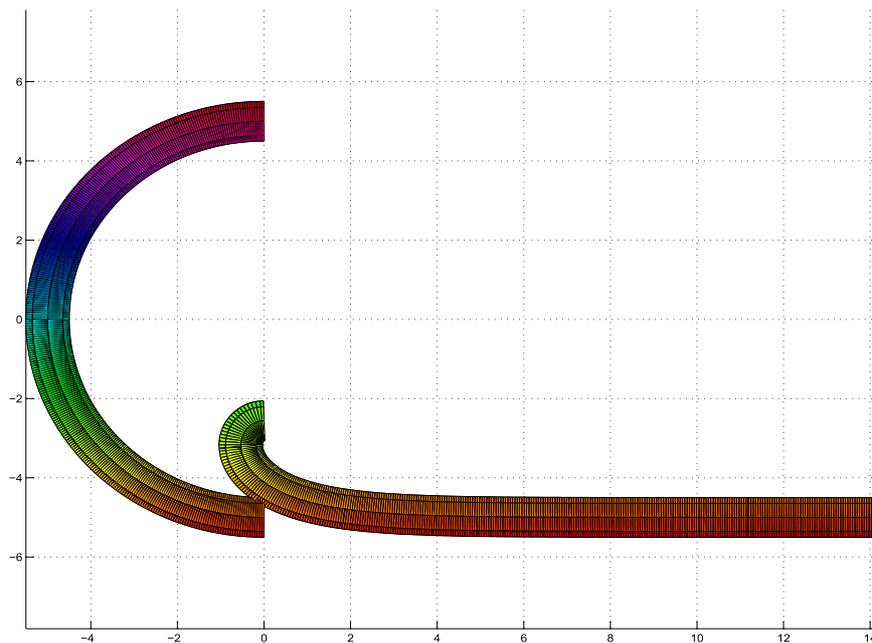


Abbildung 7.23: *elmt84* mit kubischem Ansatz und polynomialer Geometrie, 16 Elemente, 142 Lastschritte, Referenzgeometrie (rechts) und Endkonfiguration (links)

Analyse der numerischen Ergebnisse

Bei Berechnungen nur mit dem materiellen Teil der tangentialen Steifigkeitsmatrix wird keine Konvergenz erreicht. Durch Hinzunahme des geometrischen Teils kann dies behoben werden. Die beste Konvergenzrate wird mit der geometrischen Steifigkeitsmatrix nach Simo [Sim86] erreicht, welche im Folgenden eingesetzt wird. Dadurch konvergiert die Newton-Iteration bei allen Diskretisierungen in maximal 21 Schritten. Die $C1$ -Spline Elemente werden mit zwei Gauß-Punkten aufintegriert.

In Abbildung 7.24 ist die horizontale Lagerkraft in Abhängigkeit der Verschiebung u_1 am Endknoten für verschiedene Ansätze eingezeichnet. Man erkennt dort und in Tabelle 7.13, dass die Kraft für alle Ansätze geringer als bei den Schalenelementen ist. Bei Erhöhung der Ansatzordnung von *elmt82* hin zu *elmt84* tendieren die Werte gegen $F \approx 10$. Die Ergebnisse der $C1$ -Spline Elemente liegen näher an jenen der Schalenelemente. Jedoch sind die Ergebnisse mit Vorsicht zu betrachten, da eine wichtige Voraussetzung, die gleichmäßige Krümmung, nicht erfüllt ist.

Ein Vergleich der minimalen Radien r an der Symmetrieachse in Tabelle 7.13 und in Abbildung 7.25 zeigt, dass sich diese stärker unterscheiden. Je höher die Ansatzordnung ist, desto geringer sind die berechneten Radien. Die $C1$ -Spline Elemente geben einen nochmals geringeren Radius an. Insbesondere fällt auf, dass sich der Radius stark verändert, wenn der gesamte Ring diskretisiert wird. Für die Vergleichslösung der Schalenelemente stehen hier leider keine Werte zur Verfügung.

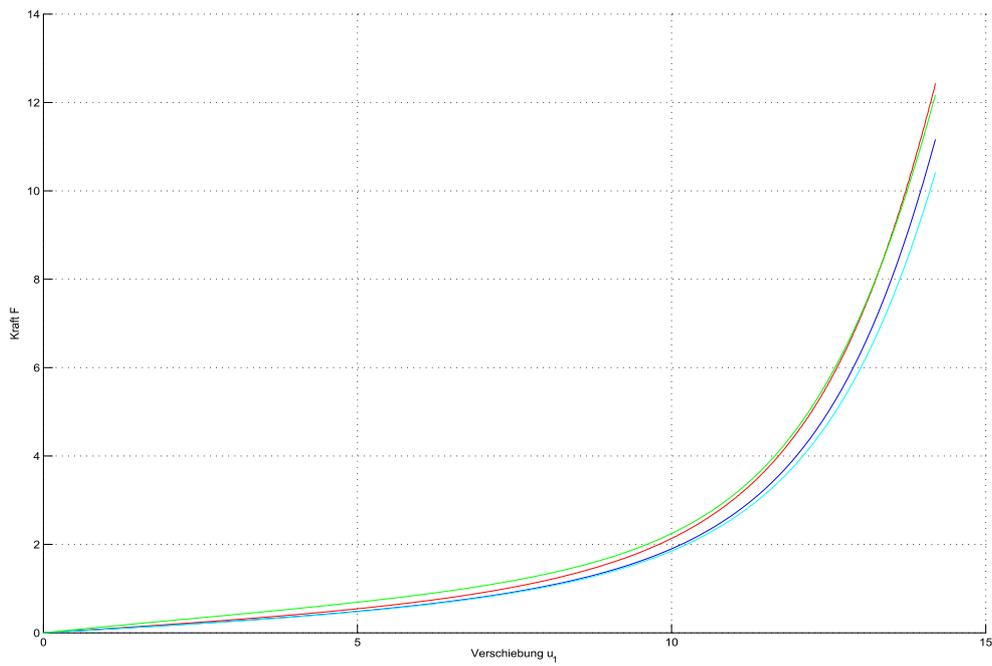


Abbildung 7.24: Vergleich der horizontalen Lagerkraft am unteren Balkenende; Vergleichslösung von LS-DYNA (rot, oben), *elmt84* mit $C1$ -Spline Ansatz (grün), *elmt82* (blau) und *elmt83* mit hierarchischem Ansatz (türkis)

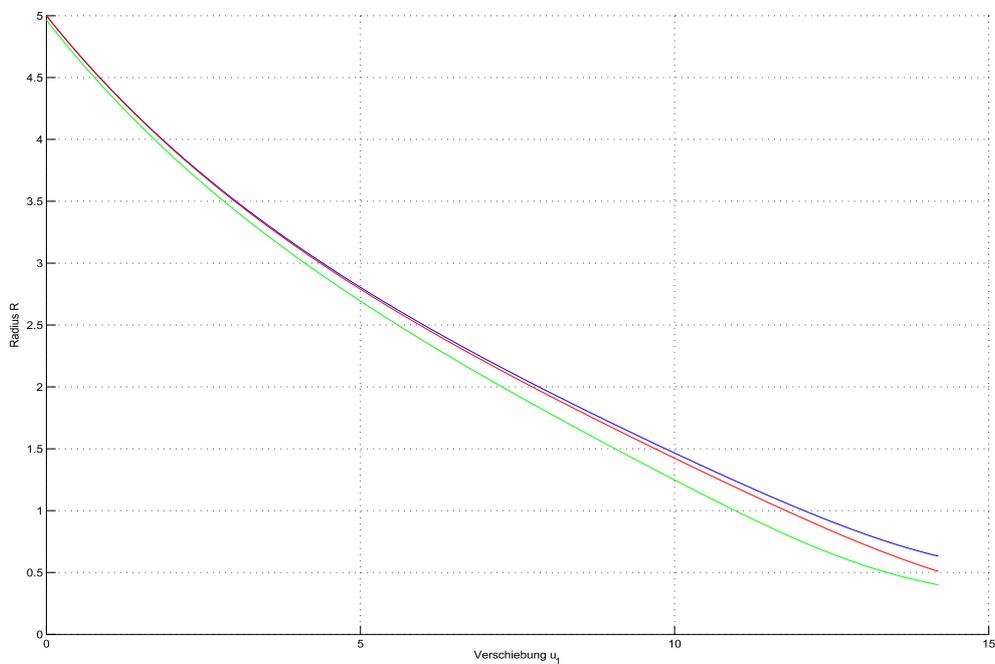


Abbildung 7.25: Vergleich der Radien am oberen Balkenende; *elmt82* (blau), *elmt83* und *elmt84* mit hierarchischem Ansatz (rot) und *elmt84* mit Spline Ansatz (grün)

7.5 Kontaktprobleme

In diesem Abschnitt werden drei Probleme unter Berücksichtigung von Kontakt zwischen den Balken vorgestellt. Dabei wird der Kontakt durch Kontaktelemente (*elmt90*, *elmt91* und *elmt92*) berücksichtigt. Zuerst betrachten wir zwei sich kreuzende Balken. Im nächsten Abschnitt modellieren wir den Fall zweier ineinandergreifender Tori, die auseinander gezogen werden. Zum Abschluss wird ein Knoten zweier Balken bzw. Kabel betrachtet. Dabei soll die Frage erörtert werden, ob sich der Knoten zusammenzieht oder nicht. Ein weiterer Gesichtspunkt ist die Frage, ob flexible Kabel mit den hier vorgestellten Balken- und Kontaktelementen dargestellt werden können.

7.5.1 Kontakt zwischen Balken

Problemstellung

Gegeben ist ein, an einem Ende eingespannter, gerader Balken der Länge 1 mit einem Querschnittsradius von $r_1 = \frac{1}{100}$. Dieser wird durch einen zweiten, starren Balken mit gleichem Radius r_2 nach oben gedrückt und somit deformiert. Reibung wird dabei nicht berücksichtigt. Der erste Balken weist in Richtung der positiven e_1 -Achse. Die Materialparameter sind $E = 200$ und $\nu = 0,3$. Der zweite Balken weist in Richtung der e_2 -Achse und wird von seinem Ausgangspunkt, direkt unter dem freien Ende des ersten Balkens, diagonal nach rechts oben verschoben, bis er direkt über der Einspannung des ersten Balkens liegt. Der Verschiebungsvektor ist $\mathbf{u} = \begin{pmatrix} -1 \\ 0 \\ 0,6 \end{pmatrix}$. Der Aufbau des Tests ist in Abbildung 7.26 dargestellt.

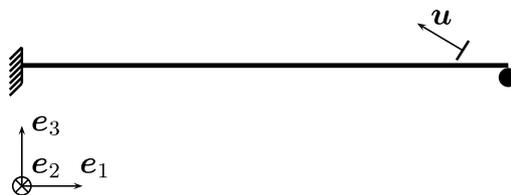


Abbildung 7.26: Aufbau des Tests “Kontakt zwischen Balken”

Numerische Berechnung

Wir diskretisieren den ersten Balken mit vier Elementen. Für den zweiten starren Balken reicht ein einzelnes Element, dessen Knoten alle festgehalten und parallel verschoben werden. Wir wählen für die Berechnung 2-Knoten Elemente (*elmt86*), polynomiale 3-Knoten Elemente (*elmt87*), polynomiale 4-Knoten Elemente (*elmt88*) und C^1 -Spline Elemente (auch *elmt88*). Die Integration ist bei polynomialen Elementen einfach reduziert, bei den C^1 -Spline Elementen jedoch doppelt. Die Verschiebung \mathbf{u} wird in 100 gleichen Lastschritten aufgetragen, um den Kontakt nicht zu verlieren. Wir werten hier nur die materielle tangentielle Steifigkeitsmatrix \mathbf{TANG}_{mat} aus und berechnen das Residuum über die erste Potentialvariation.

Art der Berechnung				Kontaktkraft	Verschiebung am freien Ende	
Elementtyp	Elemente	Lastschritte	Iterationen	F	u_1	u_3
<i>elmt82</i> reduzierte Integration	4	100	max. 61*	$-5,35676e-5$	$-1,18929e+0$	$6,59574e-1$
<i>elmt83</i> , hierarchisch reduzierte Integration	4	100	max. 46*	$-4,37949e-5$	$-1,16674e+0$	$6,77767e-1$
<i>elmt84</i> , hierarchisch reduzierte Integration	4	100	max. 48*	$-4,36787e-5$	$-1,16593e+0$	$6,78843e-1$

Tabelle 7.14: Kontakt zwischen Balken, hohe Schubsteifigkeit

Art der Berechnung				Kontaktkraft	Verschiebung am freien Ende	
Elementtyp	Elemente	Lastschritte	Iterationen	F	u_1	u_3
<i>elmt82</i> reduzierte Integration	4	70	max. 14*	$-3.62691e-6$	$-1.60214e-2$	$5.24410e-1$
<i>elmt83</i> , hierarchisch reduzierte Integration	4	70	max. 21*	$-3.87672e-6$	$-2.12042e-2$	$4.91109e-1$
<i>elmt84</i> , hierarchisch reduzierte Integration	4	70	max. 27*	$-3.64670e-6$	$-1.91191e-2$	$4.71569e-1$
<i>elmt84</i> , $C1$ -Spline doppelt red. Integration	4	70	max. 10*	$-4.02836e-6$	$-8.65495e-5$	$4.62466e-1$

Tabelle 7.15: Kontakt zwischen Balken, geringe Schubsteifigkeit

Außerdem berechnen wir das gleiche Problem mit um den Faktor 10000 verringerter Schubsteifigkeit, verglichen mit dem ersten Balken. Dadurch kann das Verhalten von Kabeln modelliert werden. Die sonstigen Berechnungsparameter bleiben gleich, allerdings werden nur 70% der Verschiebung \mathbf{u} aufgebracht.

Analyse der numerischen Ergebnisse

Ein Vergleich mit der analytischen Lösung ist hier leider nicht möglich. Abbildung 7.27 zeigt exemplarisch die deformierte Konfiguration nach dem letzten Belastungsschritt bei Verwendung von 6-Knoten Elementen (*elmt87*) und dem zugehörigen 6-Knoten Kontaktelement (*elmt91*). Die Ergebnisse bei Verwendung der anderen Elemente sind vergleichbar. Tabelle 7.14 gibt die Kontaktkraft und die Lage des freien Endes bei voller Belastung wieder. Die Ergebnisse der 6-Knoten Elemente (*elmt87*) und der 8-Knoten Elemente (*elmt88*) sind nahezu identisch. Bei Verwendung von $C1$ -Spline Elementen wird die Berechnung bei etwa 90% der Belastung instabil. Daher sind keine Ergebnisse verfügbar. Der Verlauf der Kontaktkraft bei den einzelnen Belastungsschritten kann Abbildung 7.28 entnommen werden. Die Abweichungen bei $u_1 \approx 0,27$ und $u_1 \approx 0,95$ gehen auf den Übergang von einem Kontaktelement zu nächsten zurück und spiegeln keine Instabilitäten wider. Dabei werden lokal die Kontaktkräfte auf zwei Kontaktelemente verteilt, sodass die geplotteten Werte nach unten abweichen.

Die Ergebnisse des zweiten Tests mit geringerer Schubsteifigkeit sind in Abbildung 7.29 exemplarisch für polynomiale 4-Knoten Elemente (*elmt88*) zu sehen. Aufgrund der hohen Biegesteifigkeit sind die Balkenquerschnitte nicht verdreht. Die Deformation zwischen der Einspannung links und der Kontaktstelle ist nur auf Schubverzerrung zurückzuführen. Dies erklärt auch den, auf den ersten Blick überraschenden, Knick im Bereich der Kontaktstelle. Das rechte Ende ist frei verschiebbar, dort greift ist auch keinerlei Belastung

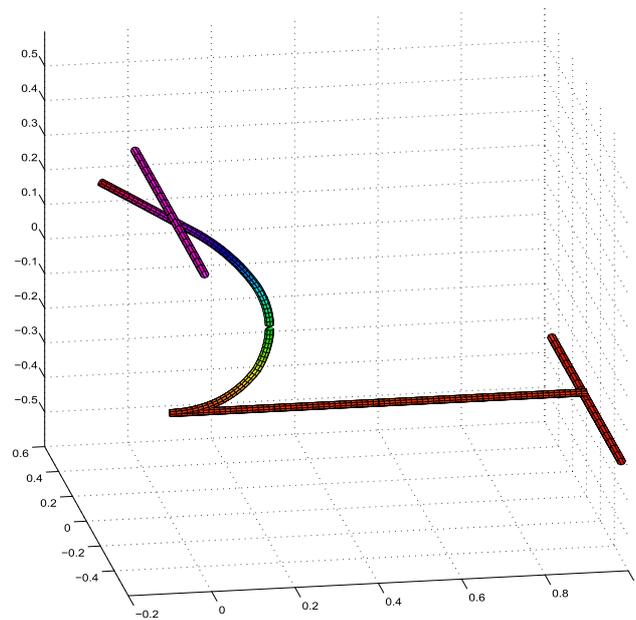


Abbildung 7.27: *elmt87* mit polynomialem Ansatz, 4 Elemente, hohe Schubsteifigkeit, Referenzgeometrie (unten) und Endkonfiguration (oben)

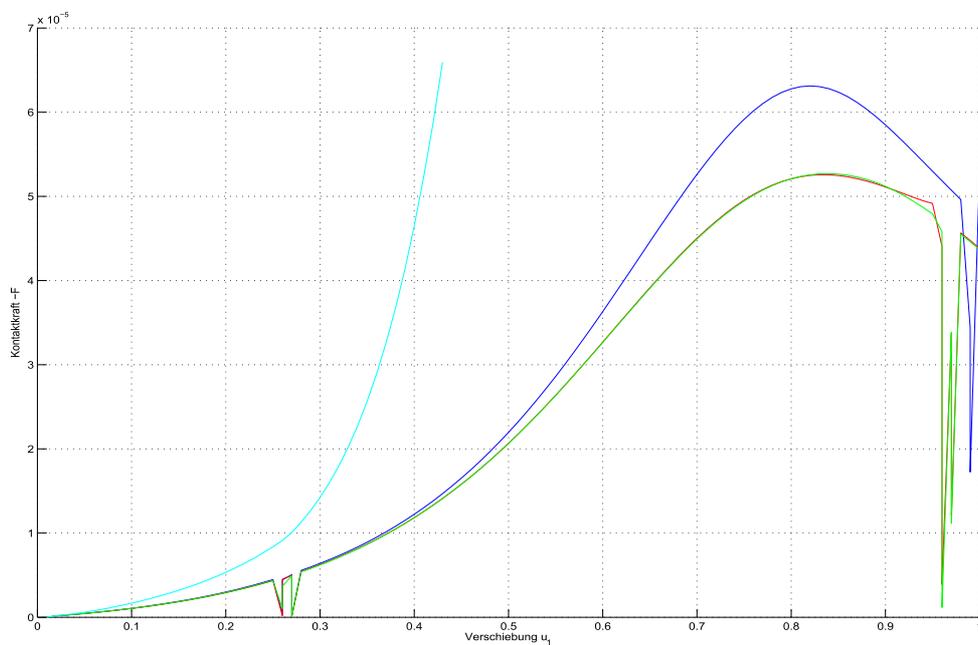


Abbildung 7.28: Verlauf der Kontaktkraft bei hoher Schubsteifigkeit; C^1 -Spline Ansatz (türkis), 2-Knoten Elemente (blau), 3-Knoten Elemente (rot) und 4-Knoten Elemente (grün)

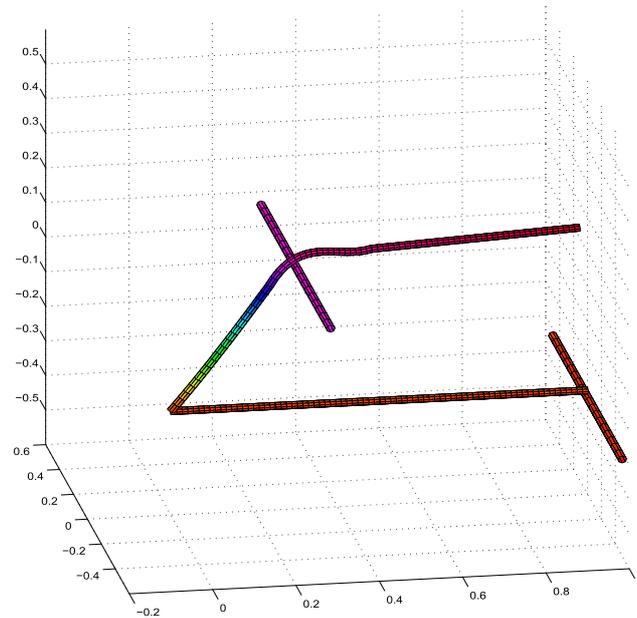


Abbildung 7.29: *elmt88* mit polynomialem Ansatz, 4 Elemente, geringe Schubsteifigkeit, Referenzgeometrie (unten) und Endkonfiguration (oben)

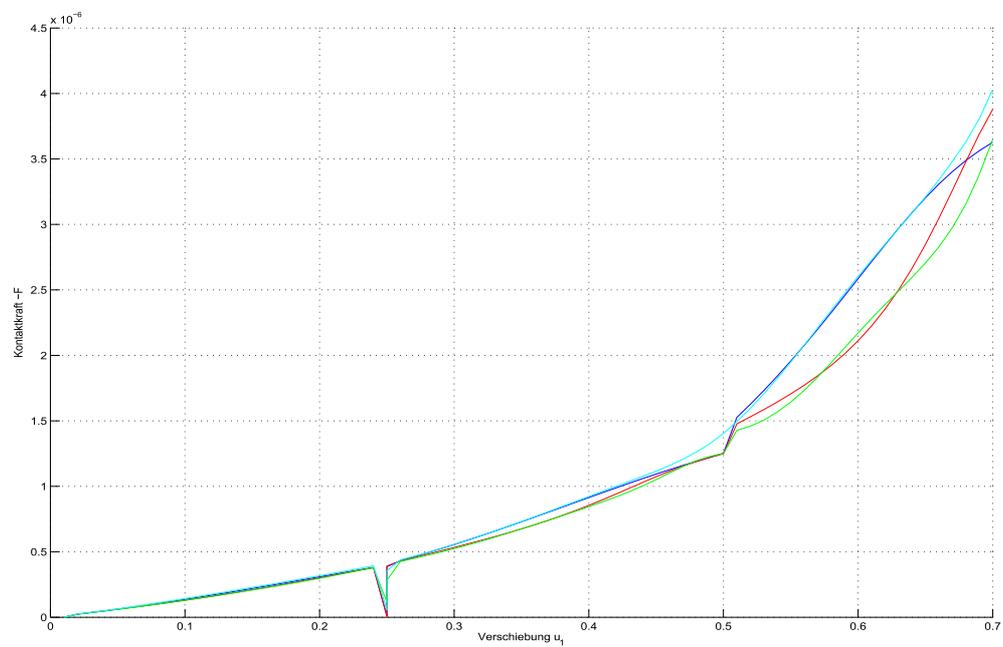


Abbildung 7.30: Verlauf der Kontaktkraft bei geringer Schubsteifigkeit; C^1 -Spline Ansatz (türkis), 2-Knoten Elemente (blau), 3-Knoten Elemente (rot), 4-Knoten Elemente (grün)

an. In Tabelle 7.15 erkennt man, dass in diesem Fall weniger Newton-Iterationen benötigt werden. Die Ergebnisse der 3-Knoten und 4-Knoten Elemente weichen nun etwas stärker voneinander ab. Dies ist auf den fehlenden Freiheitsgrad von *elmt87* zurückzuführen. Dadurch lässt sich der Knick an der Kontaktstelle nur unzureichend wiedergeben. Die volle Belastung kann nun auch bei den $C1$ -Spline Elementen aufgetragen werden, sodass nun Ergebnisse zur Verfügung stehen. Die Horizontalverschiebung u_1 wird von den $C1$ -Spline Elementen um Größenordnungen geringer angegeben. In Abbildung 7.30 ist zu erkennen, dass bei $u_1 = 0,25$ und $u_1 = 0,5$ wiederum Elementübergänge stattfinden. Der erste Übergang hat ausschließlich lokale Auswirkungen, der zweite Übergang beeinflusst den Kraft-Verschiebungsverlauf jedoch nachhaltig.

7.5.2 Kontakt zwischen Ringen

Problemstellung

In diesem Test sind zwei Tori gegeben. Der Erste liegt in der e_1, e_2 -Ebene und der Zweite in der e_1, e_3 -Ebene. Die beiden Kreise der Schwerpunktslinien haben einen Radius $R_0 = 1$, der Querschnittsradius beträgt jeweils $r = 0,2$. Die Materialparameter haben wieder die Werte $E = 200$ und $\nu = 0,3$. In Abbildung 7.31 ist die genaue Ausgangslage der beiden Tori illustriert. Der rechte Torus wird in die positive e_1 -Richtung verschoben, sodass sich die Tori deformieren. Der Kontakt findet bei exakter Darstellung der Geometrie ab einer Verschiebung $b_1 = 0,6$ statt.

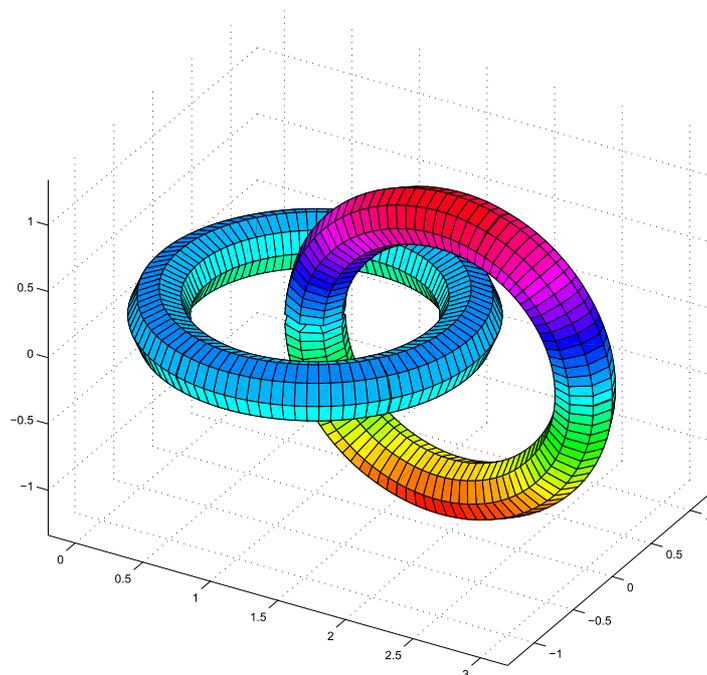


Abbildung 7.31: *elmt88* mit kubischem Ansatz, 2×5 Elemente, Referenzkonfiguration

Numerische Berechnung

Die beiden Tori müssen jeweils mit einer ungeraden Zahl an Elementen diskretisiert werden, wenn die Kontaktstelle nicht direkt am Übergang zweier Elemente ist und im Ring gegenüber ein Knoten für die feste Einspannung liegt. Wir diskretisieren die Tori deshalb mit jeweils fünf Elementen. Dabei werden 2-Knoten Elemente (*elmt86*), polynomiale 3-Knoten Elemente mit Kreisgeometrie (*elmt87*) und polynomiale 4-Knoten Elemente mit Lagrange Geometrie (*elmt88*) getestet. Daneben rechnen wir auch mit 15 $C1$ -Spline Elementen (*elmt88*) je Torus. Da die beiden Tori kein Ende besitzen, müssen wir für die $C1$ -Spline Diskretisierung keine zusätzlichen Anfangs- und Endelemente definieren. Die Verschiebung wird solange in kleinen Schritten $\Delta u_1 = 0,001$ erhöht, bis die Berechnung instabil wird. Das Residuum wird über die erste Variation des Potentials berechnet.

Dieser Test eignet sich sehr gut, um die Stabilität der verschiedenen tangentialen Steifigkeitsmatrizen zu ermitteln. Wir testen somit neben der materiellen tangentialen Steifigkeitsmatrix \mathbf{TANG}_{mat} auch die verschiedenen geometrischen tangentialen Steifigkeitsmatrizen \mathbf{TANG}_{geo} aus Kapitel 4.2.7, von Ibrahimbegovic [Ibr95] und von Simo [Sim86]. In Tabelle 7.16 ist aufgeführt bis zu welcher Verschiebung u_1 die einzelnen Berechnungen stabil bleiben und welche Kontaktkraft dann wirkt. Außerdem ist zum Vergleich die Kontaktkraft bei einer Verschiebung $u_1 = 0,75$ aufgeführt.

Analyse der numerischen Ergebnisse

In Tabelle 7.16 und Abbildung 7.33 fällt zuerst auf, dass die Ergebnisse der linearen Elemente (*elmt86*) stark von den anderen Elementen abweichen. Dies ist auf die nur ungenaue Geometrieapproximation zurückzuführen, sodass der Kontakt schon ab einer Verschiebung $u_1 = 0,22$ auftritt. Außerdem zeigt die Steigung der Kurve, dass die Elemente zu steif sind. Die Wahl der geometrischen tangentialen Steifigkeitsmatrix ändert hierbei nichts. Ein vergleichbares Phänomen tritt bei einer Diskretisierung mit nur fünf $C1$ -Spline Elementen auf.

Die Ergebnisse der restlichen Berechnungen weichen kaum voneinander ab, nur die maximale Belastung unterscheidet sich stark. Ein Vergleich der Kontaktkräfte bei $u_1 = 0,75$ zeigt, dass die $C1$ -Spline Elemente die höchste Steifheit besitzen, während die kubischen Elemente die geringste Kontaktkraft zeigen. Die Wahl der geometrischen tangentialen Steifigkeitsmatrix hat auch hier keinen Einfluss auf das Ergebnis.

Die Rechnung mit quadratischen Elementen (*elmt87*) ist instabiler als mit kubischen (*elmt88*). Die $C1$ -Spline Elemente zeigen aufgrund der größeren Zahl an Elementen nochmals eine größere Stabilität. Interessant ist hier außerdem der Einfluss der geometrischen tangentialen Steifigkeitsmatrix \mathbf{TANG}_{geo} . So kann die Verschiebung u_1 teilweise stark erhöht werden, wenn neben der materiellen tangentialen Steifigkeitsmatrix auch der geometrische Teil berücksichtigt wird. Die Matrizen von Ibrahimbegovic und Simo liefern nahezu identische Ergebnisse. Die in Kapitel 4.2.7 hergeleitete geometrische Steifigkeitsmatrix zeigt in diesem Test ein deutlich stabileres Verhalten.

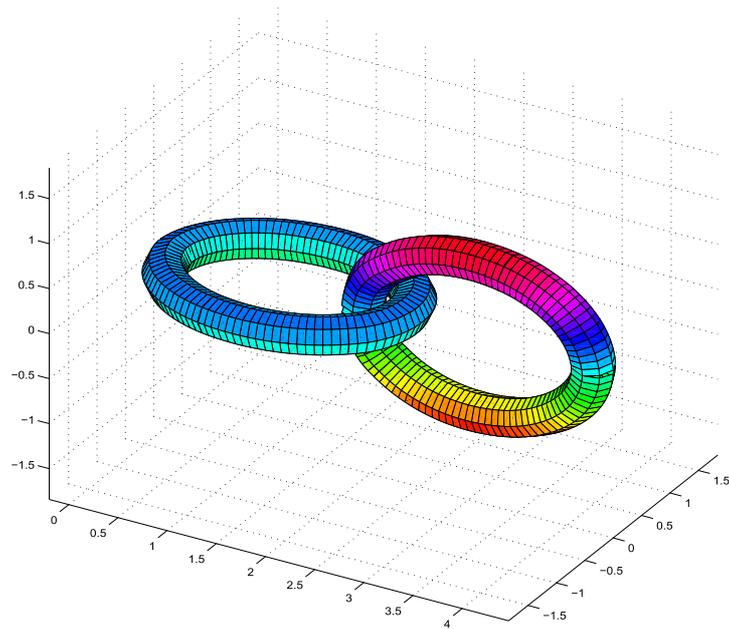


Abbildung 7.32: *elmt88* mit kubischem Ansatz, 2×5 Elemente, Konfiguration mit $u_1 = 1,273$

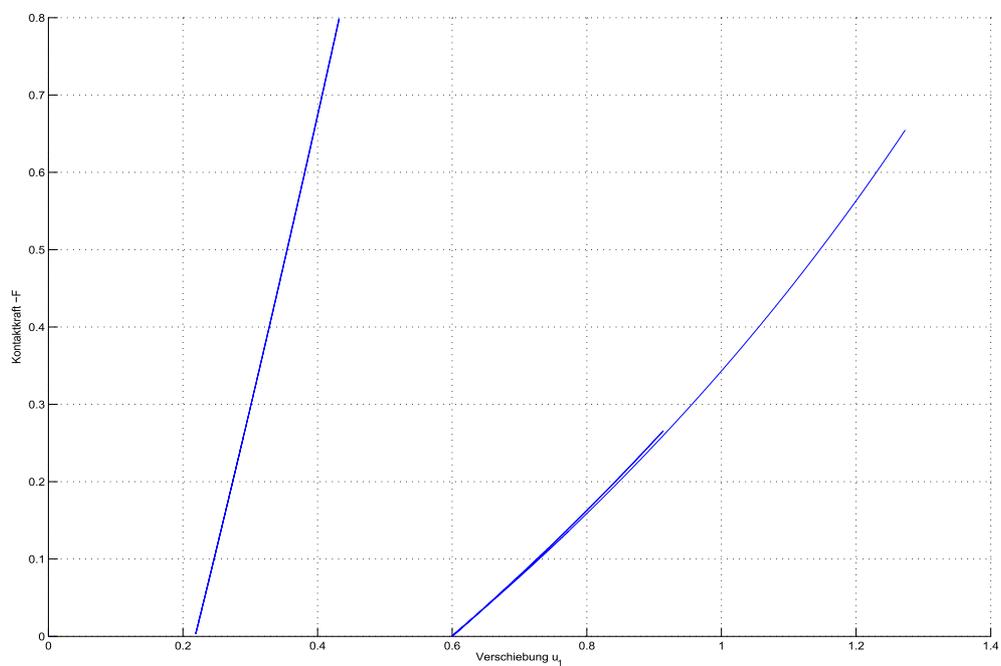


Abbildung 7.33: Vergleich der Verschiebungen und Kontaktkräfte; *elmt86* (links), *elmt87* (rechts, kurze Linie), *elmt88* polynomial und C^1 -Spline (rechts, lange Linie)

Art der Berechnung		Kontaktkraft		maximale Verschiebung	
Elementtyp	Elemente	$TANG_{geo}$	$F @ u_1 = 0,75$	$F @ u_1 = \max$	u_1
<i>elmt86</i> reduzierte Integration	5	ohne	-	$-7,98659e - 1$	0,432
		Kapitel 4.2.7	-	$-7,98659e - 1$	0,432
		Ibrahimbegovic	-	$-7,98659e - 1$	0,432
		Simo	-	$-7,98659e - 1$	0,432
<i>elmt87</i> , hierarchisch Kreisgeometrie reduzierte Integration	5	ohne	$-1,19965e - 1$	$-1,28398e - 1$	0,760
		Kapitel 4.2.7	$-1,19965e - 1$	$-2,66149e - 1$	0,914
		Ibrahimbegovic	$-1,19965e - 1$	$-2,64229e - 1$	0,912
		Simo	$-1,19965e - 1$	$-2,65189e - 1$	0,913
<i>elmt88</i> , hierarchisch Lagrange Geometrie reduzierte Integration	5	ohne	$-1,17170e - 1$	$-1,36125e - 1$	0,773
		Kapitel 4.2.7	$-1,17170e - 1$	$-6,54637e - 1$	1,273
		Ibrahimbegovic	$-1,17170e - 1$	$-2,94098e - 1$	0,950
		Simo	$-1,17170e - 1$	$-2,95059e - 1$	0,951
<i>elmt88</i> , $C1$ -Spline $C1$ -Spline Geometrie doppelt red. Integration	15	ohne	$-1,23282e - 1$	$-1,55615e - 1$	0,787
		Kapitel 4.2.7	$-1,23282e - 1$	$-7,78731e - 1$	1,330
		Ibrahimbegovic	$-1,23282e - 1$	$-6,03697e - 1$	1,205
		Simo	$-1,23282e - 1$	$-6,04999e - 1$	1,206

Tabelle 7.16: Kontakt zwischen Ringen

7.5.3 Kontakt bei Knoten

Bei diesem Test soll herausgefunden werden, inwieweit die hier implementierten Finiten Elemente einen Kreuzknoten modellieren können. Wir wollen insbesondere die Fragestellung betrachten, ob sich der Knoten fest zusammenzieht, wenn die Enden auseinander gezogen werden.

Problemstellung

Gegeben sind zwei Balken in Form von langgezogenen Schlingen, die einen kreisförmigem Querschnitt besitzen. Sie sind durch einen Kreuzknoten miteinander verflochten. Dies ist auch aus Abbildung 7.34 ersichtlich. Die vier Enden sind in e_3 -Richtung frei verschiebbar, d.h. der Abstand der beiden Endknoten eines Balkens ist nicht fest vorgegeben. Außerdem ist bei beiden Balken jeweils der mittlere Knoten in e_3 -Richtung festgehalten. Somit ergibt sich ein System mit zwei Symmetrieachsen, wobei der Abstand der Balkenenden des gleichen Balkens veränderlich ist. Wir belasten den Knoten durch Auseinanderziehen der Balken in e_1 -Richtung. Als Material modellieren wir einerseits einen runden Stahldraht mit Radius $r = 1$, andererseits auch ein Kabel, dessen Biege-, Torsions- und Schubsteifigkeit im Verhältnis zur Zugfestigkeit stark verringert ist.

Numerische Berechnung

Bei Kabeln ist die $C1$ -Kontinuität eine wichtige physikalische Eigenschaft, deshalb rechnen wir vor allem mit $C1$ -Spline Elementen. Zum Vergleich wählen wir aber auch kubische hierarchische Elemente. Es existieren drei Diskretisierungen, einmal mit 16 $C1$ -Spline Elementen je Balken, einmal mit 32 $C1$ -Spline Elementen und schließlich mit 12 kubischen Elementen. Hinzu kommen 40, 176 bzw. 50 Kontaktelemente.

Als Materialparameter wählen wir für den Draht den Elastizitätsmodul $E = 200$ und die Querkontraktion $\nu = 0,3$. Für die Kabel wird die Zugfestigkeit des Drahtes übernommen, die anderen Steifigkeiten werden um den Faktor 1000 verringert.

Die Belastung wird mit einer Schrittweite von $\Delta u_1 = 0,02$ aufgetragen. Bis zum Auftreten des ersten Kontaktes kann jedoch ein großer Lastschritt von $\Delta u_1 = 7$ realisiert werden. Das Residuum wird über die erste Potentialvariation berechnet. Nach den Ergebnissen im vorhergehenden Test, Kapitel 7.5.2, rechnen wir unter Hinzunahme der geometrischen Steifigkeitsmatrix $TANG_{geo}$ aus Kapitel 4.2.7.

Analyse der numerischen Ergebnisse

Der erste Kontakt findet bei einer Verschiebung $u_1 \approx 7,2$ statt. Dabei berühren sich die beiden Flanken, was auch in Abbildung 7.35 genau in der Bildmitte zu erkennen ist. Bei einer Verschiebung $u_1 \approx 14$ kommen vier weitere Kontaktstellen hinzu, die rechts und links der Bildmitte liegen. In den Abbildungen 7.36 bis 7.38 sind die beiden Stellen gut zu erkennen.

Ein großes Problem ist jedoch die Wahl eines optimalen Strafparameters ϵ_N . Wählen wir einen zu großen Wert, wird die Berechnung beim ersten Auftreten des Kontaktes instabil. Wenn ϵ_N zu klein ist, durchdringen sich die Balken gegenseitig, was die Güte des Ergebnisses beeinträchtigt. Insbesondere an den Flanken ist die Kontaktkraft hoch, sodass dort die Kontaktbedingung stärker verletzt wird.

Zuerst stellen wir fest, dass sich die kubischen Elemente (*elmt88*) ohne $C1$ -Kontinuität nicht zur Berechnung dieses Problems eignen. In den folgenden Plots sind schon ab einer Verschiebung $u_1 \approx 12$ Instabilitäten zu erkennen. Wir gehen deshalb nicht weiter auf die Ergebnisse der kubischen Elemente ein und untersuchen im Folgenden nur noch die Ergebnisse der beiden Diskretisierungen mit 16 und 32 $C1$ -Spline Elementen.

In Abbildung 7.36 sehen wir, dass sich die beiden Enden der Balken erkennbar annähern, sobald der zweite Kontakt bei $u_1 \approx 14$ auftritt. Der Knoten beginnt sich leicht zusammenzuziehen. Allerdings wird die Berechnung instabil, bevor sich der Knoten erkennbar festgezogen hat. Die Ergebnisse der beiden Diskretisierungen unterscheiden sich hier nur geringfügig. Ein ähnliches Bild zeigt sich bei der Analyse der Lagerreaktionskraft in Abbildung 7.38. Diese bleibt zu Beginn gleich null. Mit dem ersten Kontakt steigt die Lagerreaktionskraft linear an und beschreibt ab dem zweiten Kontakt eine steilere und nichtlineare Kurve. Die Reaktionskräfte sind bei der feineren Diskretisierung jedoch nur einen Bruchteil so groß.

Abbildung 7.37 zeigt, dass sich die Ergebnisse bei Kabeln grundlegend gegenüber jenen bei Balken unterscheiden. Dort beginnen sich die Enden auseinander zu bewegen, sobald der erste Kontakt auftritt. Mit dem zweiten Kontakt wird dies nochmals verstärkt. Da sich hier auch die Verschiebungen der beiden Diskretisierungen stark unterscheiden, stellt sich die Frage nach der Aussagekraft der Ergebnisse. Insbesondere Abbildung 7.39 verstärkt diesen Eindruck. Eine Verringerung der Lagerreaktionskraft bei Erhöhung der Belastung, wie sie hier durch die obere, blaue Kurve dargestellt ist, widerspricht dem mechanischen Modell.

Zusammenfassend kann gesagt werden, dass die Ergebnisse der Berechnung mit Materialparametern für Balken nachvollziehbar sind. Allerdings bricht die Berechnung aus Stabilitätsgründen bei der Kontaktformulierung ab. Somit kann die Frage, wie weit sich der Knoten zusammenziehen wird, nicht beantwortet werden. Bei Kabeln weichen die Ergebnisse stark untereinander ab und widersprechen grundlegenden mechanischen Prinzipien.

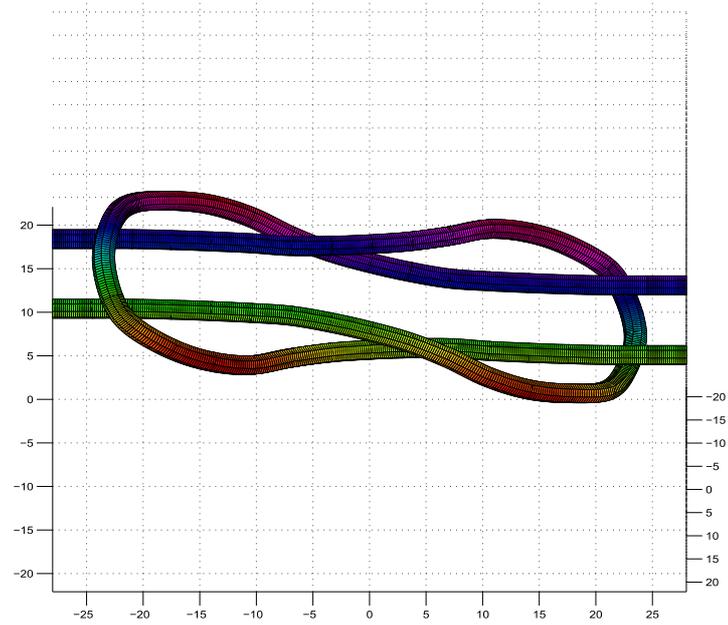


Abbildung 7.34: Referenzkonfiguration des Tests “Kontakt bei Knoten” mit 32 C_1 -Spline Elementen je Balken

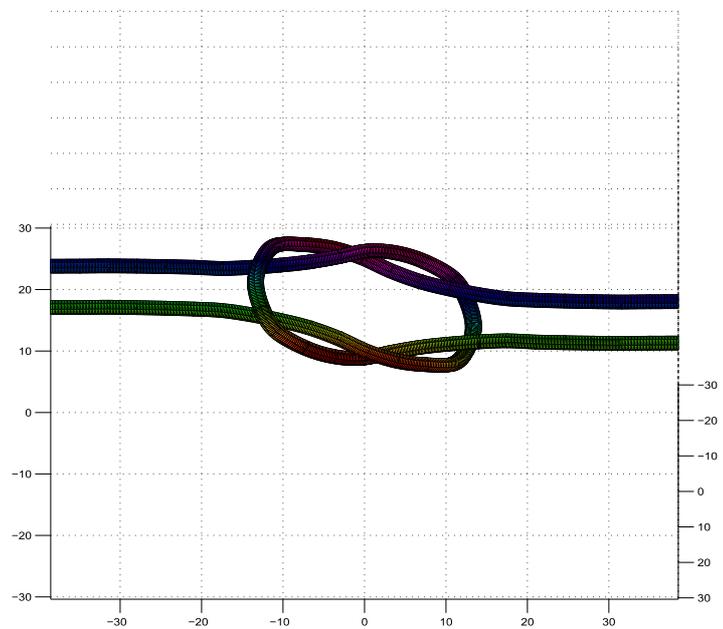


Abbildung 7.35: Aktuelle Konfiguration mit 32 C_1 -Spline Elementen je Balken bei einer Verschiebung $u_1 = 21$

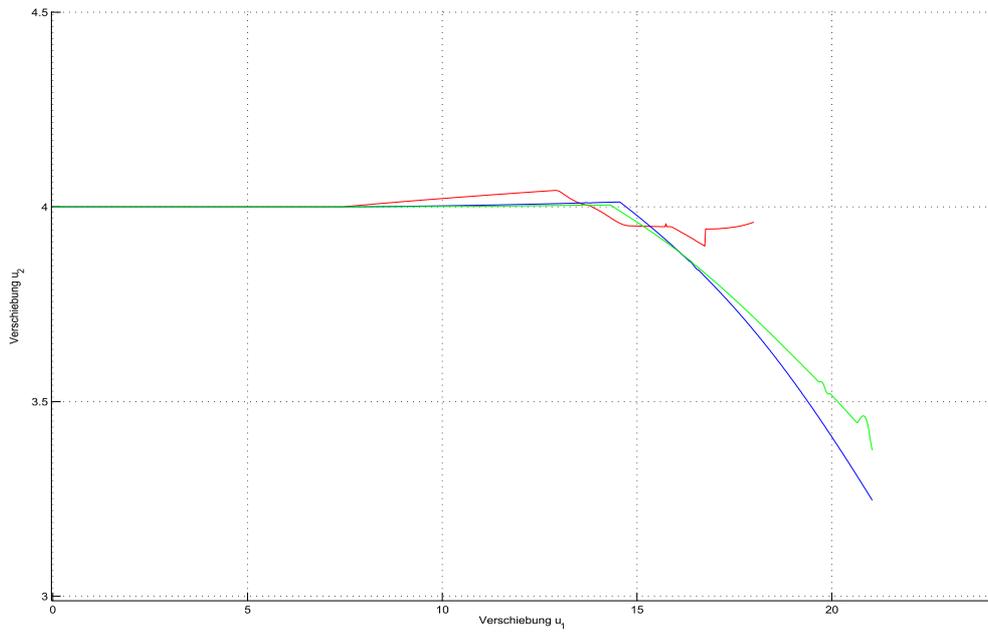


Abbildung 7.36: Vergleich der Verschiebungen u_3 bei Balken, 16 C^1 -Spline Elemente (blau), 32 C^1 -Spline Elemente (grün) und 10 kubische Elemente (rot)

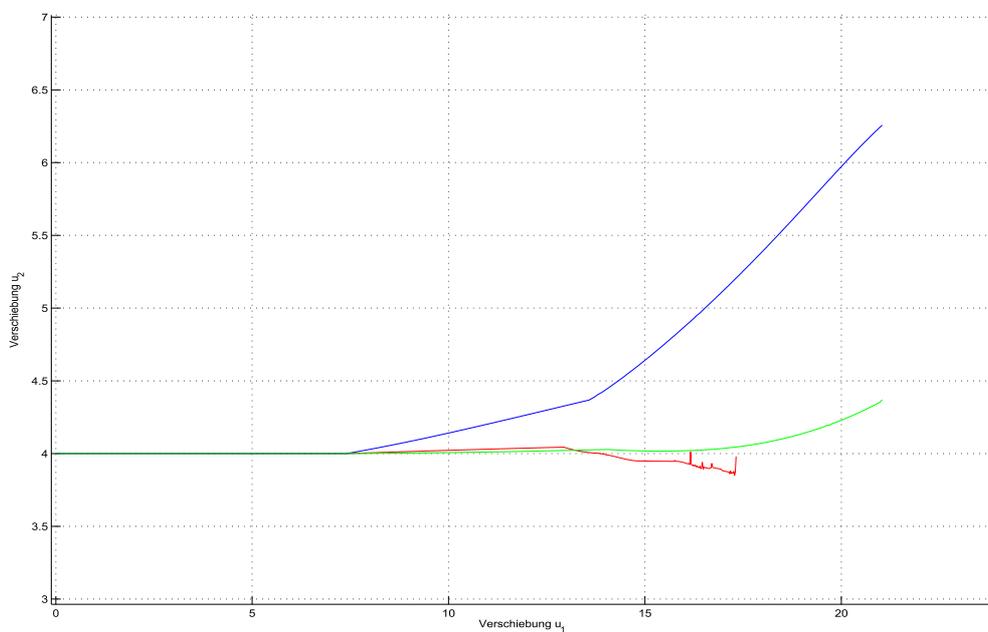


Abbildung 7.37: Vergleich der Verschiebungen u_3 bei Kabeln, 16 C^1 -Spline Elemente (blau), 32 C^1 -Spline Elemente (grün) und 10 kubische Elemente (rot)

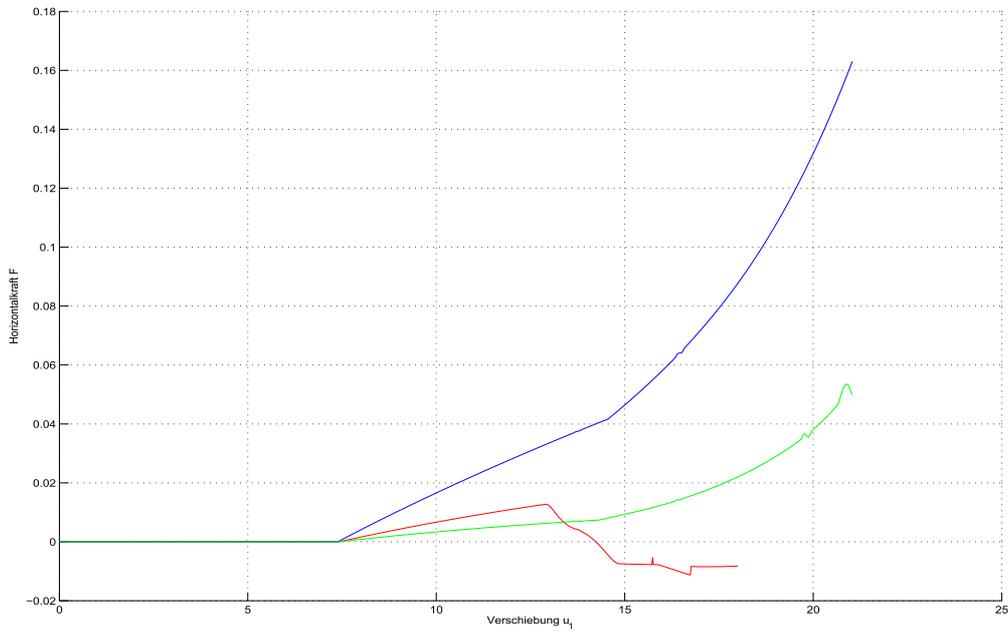


Abbildung 7.38: Vergleich der Einspannkraft in e_1 -Richtung bei Balken, 16 C^1 -Spline Elemente (blau), 32 C^1 -Spline Elemente (grün) und 10 kubische Elemente (rot)

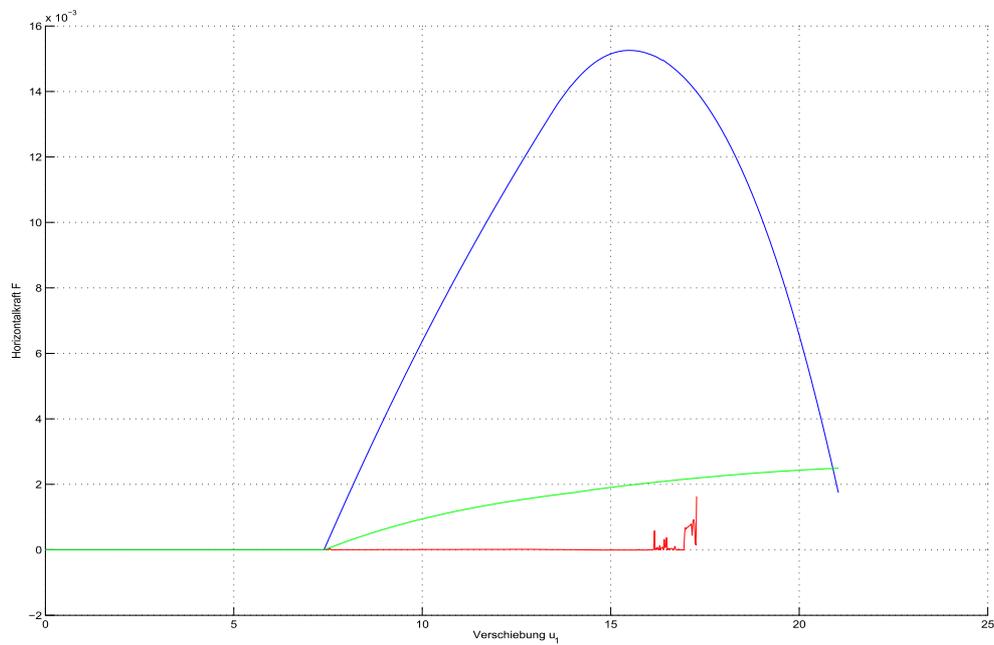


Abbildung 7.39: Vergleich der Einspannkraft in e_1 -Richtung bei Kabeln, 16 C^1 -Spline Elemente (blau), 32 C^1 -Spline Elemente (grün) und 10 kubische Elemente (rot)

Kapitel 8

Ergebnisse und Ausblick

In diesem Kapitel werden die Ergebnisse der Elementtests kurz diskutiert. Dabei wird auf vier Hauptaspekte näher eingegangen. Zuerst betrachten wir den Einfluss der reduzierten Integration. Danach wird die Verwendung der geometrischen Steifigkeitsmatrix \mathbf{TANG}_{geo} erläutert. Außerdem erörtern wir die Frage nach dem schlechten Konvergenzverhalten des Newton-Verfahrens bei dreidimensionalen Problemen. Zuletzt werden einige Aspekte zur Kontaktformulierung betrachtet.

8.1 Reduzierte Integration

Im Zuge der Elementtests ist eine starke Verbesserung der Ergebnisse bei reduzierter Integration zu beobachten. Die polynomialen Ansätze (*elmt82*, *elmt83*, *elmt84*, *elmt86*, *elmt87* und *elmt88*), als auch die rationalen Ansätze (*elmt83* und *elmt84*) zeigen die exaktesten Ergebnisse, wenn die Zahl der Gauß-Punkte um eins geringer als die Zahl der Knotenpunkte ist. Wenn weitere Gauß-Punkte ausgelassen werden, wird die globale Steifigkeitsmatrix \mathbf{TANG} singulär.

Bei den C^1 -Spline Elementen (*elmt84* und *elmt88*) ist das Ergebnis jedoch weniger eindeutig. Meist sind die besten Ergebnisse mit zwei Gauß-Punkten erzielbar. Die Unterschiede bei Verwendung von drei bzw. vier Integrationspunkten sind nur sehr klein. Bei einfachen Belastungsfällen reicht teilweise ein einziger Gauß-Punkt aus, um die exaktesten Verschiebungen zu erhalten oder um die beste Konvergenzrate zu erzielen. Jedoch tritt auch der umgekehrte Fall, dass drei Integrationspunkte benötigt werden, ein.

Die Verbesserung der Ergebnisse bei reduzierter Integration lässt sich durch geometrisches Locking erklären. Alle hier verwendeten Ansätze haben die gleiche Ansatzordnung für die Verschiebung \mathbf{u} , wie auch für die Verdrehung \mathbf{w} . Nach dem mechanischen Modell in Kapitel 3 berechnen sich die Normal- und Schubverzerrungen durch $\boldsymbol{\varepsilon} = \bar{\mathbf{u}}' - \mathbf{a}_1$. Für kleine Verschiebungen gilt

$$\boldsymbol{\varepsilon} = \bar{\mathbf{u}}' - \exp \widehat{\mathbf{W}} \boldsymbol{\Lambda}_0 \mathbf{e}_1 \approx \bar{\mathbf{u}}' - \exp \widehat{\mathbf{W}} \boldsymbol{\Lambda}_0 \mathbf{e}_1 = \bar{\mathbf{u}}' - (\mathbf{I}_3 + \widehat{\mathbf{W}}) \boldsymbol{\Lambda}_0 \mathbf{e}_1 = \mathbf{u}' - \widehat{\mathbf{W}} \boldsymbol{\Lambda}_0 \mathbf{e}_1,$$

d.h. die Schubverzerrungen $\boldsymbol{\varepsilon}$ hängen direkt von \mathbf{u}' und \mathbf{w} ab. Infolgedessen können parasitäre Schubverzerrungen entstehen, da sich nun die Ansatzordnungen von \mathbf{u}' und \mathbf{w} unterscheiden. Diese versteifen das System, sodass die Deformationen zu gering ausgege-

ben werden. Bei reduzierter Integration werden die parasitären Schubverzerrungen nicht erkannt, da diese an den verbliebenen Integrationspunkten verschwinden. Wenn die Zahl der Integrationspunkte weiter verringert wird, können auch real existierende Verzerrungen nicht erkannt werden. Die Steifigkeitsmatrix **TANG** wird dadurch singulär.

Das Verhalten der $C1$ -Spline Elemente kann dadurch nicht vollständig erklärt werden. Insbesondere die Frage, warum die globale Steifigkeitsmatrix bei mehrfach reduzierter Integration nicht unbedingt singulär wird, bleibt offen.

Eine weitere Möglichkeit zur Vermeidung von Locking, besteht in der Wahl unterschiedlicher Ansatzordnungen für \mathbf{u} und \mathbf{w} . Die Arbeit von Ibrahimbegovic [Ibr95] folgt diesem Ansatz und kann Locking dadurch weitgehend vermeiden. Die Implementierung unterschiedlicher Ansatzordnungen erfordert die Neukodierung großer Teile der Elementroutinen. Insbesondere entstehen dabei Knoten, die nur eine verringerte Zahl an Freiheitsgraden besitzen.

8.2 Tangentiale Steifigkeitsmatrix **TANG**

In Kapitel 4.2.7 wird eine tangentielle Steifigkeitsmatrix hergeleitet, die in einen materiellen Teil **TANG**_{mat} und einen geometrischen Teil **TANG**_{geo} aufgespalten wird. Ibrahimbegovic [Ibr95] und Simo [Sim86] definieren in ihren Arbeiten leicht veränderte geometrische Steifigkeitsmatrizen, leiten diese jedoch nicht ausführlich her. Da die materielle Steifigkeitsmatrix symmetrisch ist und somit effizientere Algorithmen zur Lösung des linearen Gleichungssystems zur Verfügung stehen, stellt sich außerdem die Frage, inwieweit der geometrische Teil ganz weggelassen werden kann.

Bei den ersten Elementtests, bis hin zum Test von Bathe in Kapitel 7.3, ist die Konvergenzrate bei alleiniger Verwendung der materiellen Steifigkeitsmatrix **TANG**_{mat} am höchsten. Teilweise kann bei Berücksichtigung der geometrischen Steifigkeitsmatrix **TANG**_{geo}, unabhängig von der Wahl der Definition, keine Konvergenz erzielt werden. Dieses Ergebnis überrascht auf den ersten Blick, da mit der Berücksichtigung der geometrischen Steifigkeitsmatrix eine Verbesserung einhergehen sollte. Da bei der Variationsformulierung Dirichlet-Randbedingungen angenommen wurden, bei all diesen Test jedoch gemischte Randbedingungen gelten, ist die Abweichung von der Theorie erklärbar.

Beim virtuellen Ringzugversuch in Kapitel 7.4, der nun auch Dirichlet-Randbedingungen beinhaltet, wird mit der materiellen Steifigkeitsmatrix **TANG**_{mat} keine Konvergenz erreicht. Hier muss die geometrische Steifigkeitsmatrix **TANG**_{geo} mitberücksichtigt werden, um die gesamte Verschiebung auftragen zu können. Dabei ist die Konvergenzrate mit der geometrischen Steifigkeitsmatrix nach Simo am höchsten, die Unterschiede sind jedoch nur marginal.

Beim Kontaktproblem der Tori in Kapitel 7.5.2 kann die Stabilität des Newton-Verfahrens bestimmt werden. Bei Mitberücksichtigung der geometrischen Steifigkeitsmatrix **TANG**_{geo} stabilisiert sich das Newton-Verfahren, sodass höhere Belastungen aufgetragen werden können. Insbesondere bei kubischen und $C1$ -Spline Elementen zeigt die, in dieser Arbeit definierte, geometrische Steifigkeitsmatrix **TANG**_{geo} nochmals verbesserte Ergebnisse. Aufgrund der fehlenden Herleitungen in den Arbeiten von Ibrahimbegovic und Simo ist eine genauere Betrachtung dieses Phänomens leider nicht möglich. Obwohl

das Kontaktproblem, auf einen Ring reduziert, kein Dirichlet-Randwertproblem darstellt, ist das Verhalten des Gesamtsystems vergleichbar mit dem virtuellen Ringzugversuch.

8.3 Zahl der Iterationsschritte

Bei zweidimensionalen Problemen wird Konvergenz meist in sehr wenigen Iterationsschritten erreicht. Bei der Berechnung dreidimensionaler Probleme, wie dem Federmodell, dem Aufrollen eines Balkens unter Störeinfluss und dem Test nach Bathe, ist die Zahl der Newton-Iterationsschritte jedoch sehr groß. Beim Test nach Bathe ist außerdem gut zu erkennen, dass sich die Zahl der Newton-Iterationen auch bei Verkleinerung der Lastschritte nicht verringert.

Das Aufrollen eines Balkens unter Störeinfluss wird auch bei Ibrahimbegovic [IFK95] betrachtet. In dieser Arbeit wird auch auf das schlechte Konvergenzverhalten eingegangen. Allerdings verbessert sich dort die Konvergenzrate bei Berücksichtigung der geometrischen Steifigkeitsmatrix $TANG_{geo}$, was hier nicht der Fall ist.

Eine Ursache kann eine nur ungenaue Berechnung der zweiten Ableitung des Energiepotentials sein. Es stellt sich die Frage, ob die in Kapitel 4.2.7 definierte tangentielle Steifigkeitsmatrix $TANG$ bei gemischten Randbedingungen eine sinnvolle Approximation darstellt. Eventuell ergibt sich durch eine Herleitung bei expliziter Berücksichtigung der gemischten Randbedingungen eine leicht veränderte Matrix, die dann auch ein besseres Konvergenzverhalten zeigt.

8.4 Kontakt

Bei der Berechnung von Kontaktproblemen kommen neben den Balkenelementen auch Kontaktelemente zur Anwendung. Bei deren Implementierung wurden zusätzliche Annahmen getroffen, die sich auf die Ergebnisse auswirken, z.B. die Vernachlässigung von Reibung. Außerdem entstehen durch die Nebenbedingungen neue Nichtlinearitäten. Deshalb ist die Analyse der Ergebnisse bei Kontaktproblemen entsprechend aufwändiger.

Das Newton-Verfahren wird zur Berechnung des geringsten Abstands der beiden Balken verwendet. Bei der Definition wurde jedoch kein Beweis zur Konvergenz des Verfahrens geführt. Auch die Invertierbarkeit der Hesse-Matrix ist bei beliebigen Geometrien nicht gegeben. Dennoch konvergiert das Verfahren in der Anwendung zuverlässig.

Beim Auftreten einer neuen Kontaktstelle ist die Existenz der zweiten Ableitung des Energiepotentials im Allgemeinen nicht gesichert. Bei der Wahl eines sehr hohen Strafparameters ϵ_N und eines großen Lastschrittes ist deshalb ein oszillatorisches Verhalten der Newton-Iteration zu beobachten. Dies kann durch Wahl eines kleineren Strafparameters ϵ_N oder eines kleineren Lastschrittes behoben werden. Dabei wird jedoch wahlweise die Kontaktbedingung stärker verletzt oder die benötigte Rechenzeit steigt an. Als mögliche Alternativen bieten sich beispielsweise die Lagrange-Multiplikatoren bzw. die erweiterten Lagrange-Multiplikatoren an, wie sie bei anderen Kontaktalgorithmen verwendet werden. Eine andere Möglichkeit besteht in der Änderung der Kontaktkraftfunktion N in Abhängigkeit von der Durchdringung d . Beispielsweise kann $N := \epsilon_N \cdot d^2$, anstatt

$N := \epsilon_N \cdot d$ definiert werden oder der aktuelle Iterationsschritt wird mit einbezogen. Je weiter die Newton-Iteration vorangeschritten ist, desto genauer soll die Kontaktbedingung erfüllt sein.

Das letzte Testproblem, der Kreuzknoten in Kapitel 7.5.3, zeigt einige Grenzen der hier implementierten Balken- und Kontaktformulierung auf. Dies zeigt sich vor allem in der mangelnden Stabilität der Berechnung. Bevor die Frage geklärt werden kann, ob sich der Balken zusammenzieht oder nicht, bricht die Berechnung ab. Entweder konvergiert das Newton-Schema nicht länger oder der Kontakt geht verloren. Durch die fehlende Berücksichtigung von Reibung kann die Wirklichkeit nur unzureichend dargestellt werden. Außerdem ist die Annahme von punktwisem Kontakt nicht länger gerechtfertigt, wenn die Balken fast parallel aufeinander zu liegen kommen.

Insbesondere bei der Modellierung von Kabeln treten die eben angesprochenen Probleme stark hervor. Die verringerte Schubsteifigkeit verstärkt das oben angesprochene Problem der Oszillation zusätzlich.

8.5 Fazit

Die, in dieser Arbeit implementierte, geometrisch exakte Balkentheorie ermöglicht die Modellierung und Berechnung von stark deformierten Strukturen. Durch die Berücksichtigung höherer Ansatzordnungen bei der Implementierung mit der Finite Elemente Methode, kann die Genauigkeit der Ergebnisse stark verbessert werden, insbesondere bei gekrümmten Referenzgeometrien. Die Verwendung von rationalen Ansatzfunktionen bringt keine Vorteile gegenüber polynomialen Ansätzen. Dagegen bewährt sich die Berücksichtigung von Kreisgeometrien, die nicht durch eine Linearkombination aus Ansatzfunktionen und Knotenwerten dargestellt werden können.

Bei zweidimensionalen Problemen wird speziell mit reduzierter Integration eine nahezu exakte Approximation der analytischen Ergebnisse erreicht. Die schlechte Konvergenzrate des Newton-Verfahrens bei dreidimensionalen Problemen bleibt jedoch ein starker Kritikpunkt.

Die Ergebnisse des Kontaktalgorithmus geben auf Grund der Einschränkung auf punktwisem, reibungslosen Kontakt die Wirklichkeit nur unzureichend wieder. Außerdem bestehen Probleme mit der Stabilität der Kontaktformulierung. Insbesondere bei Kabelstrukturen ist dies verstärkt zu beobachten.

Anhang A

Rechenregeln und Beweise

In diesem Anhang sind einige Formeln dargestellt, die in engem Zusammenhang mit dem Vektorprodukt und der Rotationsgruppe SO_3 stehen. Danach folgen Beweise zu Sätzen aus Kapitel 2 und 3.

A.1 Rechenregeln

Definition A.1 (Vektorprodukt):

Seien $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ und $\mathbf{C} = (\mathbf{c}_1 \mid \mathbf{c}_2 \mid \mathbf{c}_3) \in \mathbb{R}^{3 \times 3}$. Dann definieren wir das Vektorprodukt

$$\mathbf{a} \times \mathbf{b} := \begin{pmatrix} a_2 b_3 - a_3 b_2 \\ a_3 b_1 - a_1 b_3 \\ a_1 b_2 - a_2 b_1 \end{pmatrix} \quad \mathbf{a} \times \mathbf{C} := (\mathbf{a} \times \mathbf{c}_1 \mid \mathbf{a} \times \mathbf{c}_2 \mid \mathbf{a} \times \mathbf{c}_3).$$

Lemma A.2

Sei $\mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \in \mathbb{R}^3$ und $\widehat{\mathbf{B}} := \begin{pmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{pmatrix}$, dann gilt $\forall \mathbf{a} \in \mathbb{R}^3$

$$\mathbf{b} \times \mathbf{a} = \widehat{\mathbf{B}} \mathbf{a}. \quad (\text{A.1})$$

Sei zusätzlich $\mathbf{c} \in \mathbb{R}^3$ und $\widehat{\mathbf{C}}$ definiert wie oben, dann gilt weiter

$$\mathbf{b} \times (\mathbf{c} \times \mathbf{a}) = \widehat{\mathbf{B}} \widehat{\mathbf{C}} \mathbf{a} = (\mathbf{c} \mathbf{b}^T - (\mathbf{b}^T \mathbf{c}) \mathbf{I}_3) \mathbf{a} \quad (\text{A.2})$$

Beweis. Gleichung (A.1) folgt direkt aus der Definition.

Bei (A.2) folgt das erste Gleichheitszeichen durch zweimalige Anwendung von (A.1). Der Rest der Behauptung folgt mit

$$\begin{aligned} \widehat{\mathbf{B}} \widehat{\mathbf{C}} - (\mathbf{c} \mathbf{b}^T - (\mathbf{b}^T \mathbf{c}) \mathbf{I}_3) &= \begin{pmatrix} -b_2 c_2 - b_3 c_3 & b_2 c_1 & b_3 c_1 \\ b_1 c_2 & -b_1 c_1 - b_3 c_3 & b_3 c_2 \\ b_1 c_3 & b_2 c_3 & -b_1 c_1 - b_2 c_2 \end{pmatrix} - \begin{pmatrix} b_1 c_1 & b_1 c_2 & b_1 c_3 \\ b_2 c_1 & b_2 c_2 & b_2 c_3 \\ b_3 c_1 & b_3 c_2 & b_3 c_3 \end{pmatrix} \\ &+ \begin{pmatrix} b_1 c_1 + b_2 c_2 + b_3 c_3 & 0 & 0 \\ 0 & b_1 c_1 + b_2 c_2 + b_3 c_3 & 0 \\ 0 & 0 & b_1 c_1 + b_2 c_2 + b_3 c_3 \end{pmatrix} \\ &= 0. \end{aligned}$$

□

Lemma A.3 (Jacobi-Identität):

Seien $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ und $\widehat{\mathbf{A}}, \widehat{\mathbf{B}}$ die zugehörigen schiefsymmetrischen Matrizen, dann gilt

$$\widehat{\mathbf{A}}\widehat{\mathbf{B}} - \widehat{\mathbf{B}}\widehat{\mathbf{A}} = \widehat{\mathbf{a} \times \mathbf{b}} \quad (\text{A.3})$$

Beweis.

$$\begin{aligned} & \widehat{\mathbf{A}}\widehat{\mathbf{B}} - \widehat{\mathbf{B}}\widehat{\mathbf{A}} \\ &= \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{pmatrix} \begin{pmatrix} 0 & -b_3 & b_2 \\ b_3 & 0 & -b_1 \\ -b_2 & b_1 & 0 \end{pmatrix} \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \\ &= \begin{pmatrix} -a_2b_2 - a_3b_3 & a_2b_1 & a_3b_1 \\ a_1b_2 & -a_3b_3 - a_1b_1 & a_3b_2 \\ a_1b_3 & a_2b_3 & -a_1b_1 - a_2b_2 \end{pmatrix} \begin{pmatrix} -a_2b_2 - a_3b_3 & a_1b_2 & a_1b_3 \\ a_2b_1 & -a_3b_3 - a_1b_1 & a_2b_3 \\ a_3b_1 & a_3b_2 & -a_1b_1 - a_2b_2 \end{pmatrix} \\ &= \begin{pmatrix} 0 & a_2b_1 - a_1b_2 & a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 & 0 & a_3b_2 - a_2b_3 \\ a_1b_3 - a_3b_1 & a_2b_3 - a_3b_2 & 0 \end{pmatrix} = \begin{pmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \widehat{\mathbf{a} \times \mathbf{b}} \end{aligned}$$

□

Folgerung A.4

Seien $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{R}^3$ dann gilt

$$\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) - \mathbf{b} \times (\mathbf{a} \times \mathbf{c}) = (\mathbf{a} \times \mathbf{b}) \times \mathbf{c}. \quad (\text{A.4})$$

Beweis. Mit (A.1) und (A.3) folgt $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) - \mathbf{b} \times (\mathbf{a} \times \mathbf{c}) = (\widehat{\mathbf{A}}\widehat{\mathbf{B}} - \widehat{\mathbf{B}}\widehat{\mathbf{A}})\mathbf{c} = (\mathbf{a} \times \mathbf{b}) \times \mathbf{c}$. □

Lemma A.5

Seien $\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d} \in \mathbb{R}^3$, sowie $\widehat{\mathbf{C}}, \widehat{\mathbf{D}}$ die zu \mathbf{c} und \mathbf{d} gehörenden schiefsymmetrischen Matrizen. Dann gilt

$$(\mathbf{a} \times \mathbf{b})^T \mathbf{c} = -\mathbf{a}^T \widehat{\mathbf{C}} \mathbf{b} \quad (\text{A.5})$$

$$(\mathbf{b} \times (\mathbf{a} \times \mathbf{c}))^T \mathbf{d} = \mathbf{a}^T \widehat{\mathbf{C}} \widehat{\mathbf{D}} \mathbf{b}. \quad (\text{A.6})$$

Beweis.

$$\begin{aligned} (\mathbf{a} \times \mathbf{b})^T \mathbf{c} &= \left(\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \right)^T \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} a_2b_3 - a_3b_2 \\ a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} \\ &= a_2b_3c_1 - a_3b_2c_1 + a_3b_1c_2 - a_1b_3c_2 + a_1b_2c_3 - a_2b_1c_3 \\ &= - \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}^T \begin{pmatrix} b_3c_2 - b_2c_3 \\ b_1c_3 - b_3c_1 \\ b_2c_1 - b_1c_2 \end{pmatrix} = - \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}^T \begin{pmatrix} 0 & -c_3 & c_2 \\ c_3 & 0 & -c_1 \\ -c_2 & c_1 & 0 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \\ &= -\mathbf{a}^T \widehat{\mathbf{C}} \mathbf{b} \end{aligned}$$

$$\begin{aligned}
(\mathbf{b} \times (\mathbf{a} \times \mathbf{a}))^T \mathbf{d} &= \left(\begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \times \begin{pmatrix} a_2 c_3 - a_3 c_2 \\ a_3 c_1 - a_1 c_3 \\ a_1 c_2 - a_2 c_1 \end{pmatrix} \right)^T \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} \\
&= \begin{pmatrix} a_1 b_2 c_2 - a_2 b_2 c_1 - a_3 b_3 c_1 + a_1 b_3 c_3 \\ a_2 b_3 c_3 - a_3 b_3 c_2 - a_1 b_1 c_2 + a_2 b_1 c_1 \\ a_3 b_1 c_1 - a_1 b_1 c_3 - a_2 b_2 c_3 + a_3 b_2 c_2 \end{pmatrix}^T \begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix} \\
&= a_1 b_2 c_2 d_1 - a_2 b_2 c_1 d_1 - a_3 b_3 c_1 d_1 + a_1 b_3 c_3 d_1 + a_2 b_3 c_3 d_2 - a_3 b_3 c_2 d_2 \\
&\quad - a_1 b_1 c_2 d_2 + a_2 b_1 c_1 d_2 + a_3 b_1 c_1 d_3 - a_1 b_1 c_3 d_3 - a_2 b_2 c_3 d_3 + a_3 b_2 c_2 d_3 \\
&= \begin{pmatrix} a_2 c_3 - a_3 c_2 \\ a_3 c_1 - a_1 c_3 \\ a_1 c_2 - a_2 c_1 \end{pmatrix}^T \begin{pmatrix} b_3 d_2 - b_2 d_3 \\ b_1 d_3 - b_3 d_1 \\ b_2 d_1 - b_1 d_2 \end{pmatrix} = \mathbf{a}^T \widehat{\mathbf{C}} \widehat{\mathbf{D}} \mathbf{b}
\end{aligned}$$

□

Lemma A.6

Sei $\mathbf{a} \in \mathbb{R}^3$ und $\mathbf{\Lambda} \in SO_3$. Definiere $\widehat{\mathbf{A}}$ und $\widehat{\mathbf{\Lambda a}}$ als zu \mathbf{a} und $\mathbf{\Lambda a}$ zugehörige schiefsymmetrische Matrizen. Dann gilt

$$\widehat{\mathbf{\Lambda a}} = \mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^T. \quad (\text{A.7})$$

Beweis. Zuerst weisen wir die Schiefymmetrie von $\mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^T$ nach.

$$\left(\mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^T \right)^T = \mathbf{\Lambda} \widehat{\mathbf{A}}^T \mathbf{\Lambda}^T = -\mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^T$$

Sei \mathbf{c} der zu $\mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^T$ gehörende axiale Vektor und $\mathbf{\Lambda} = \begin{pmatrix} \lambda_{11} & \lambda_{12} & \lambda_{13} \\ \lambda_{21} & \lambda_{22} & \lambda_{23} \\ \lambda_{31} & \lambda_{32} & \lambda_{33} \end{pmatrix}$, dann folgt durch direktes Nachrechnen

$$\mathbf{c} = \begin{pmatrix} -\lambda_{32} \lambda_{21} a_3 + \lambda_{33} \lambda_{21} a_2 + \lambda_{31} \lambda_{22} a_3 - \lambda_{33} \lambda_{22} a_1 - \lambda_{31} \lambda_{23} a_2 + \lambda_{32} \lambda_{23} a_1 \\ -\lambda_{31} \lambda_{12} a_3 + \lambda_{31} \lambda_{13} a_2 + \lambda_{32} \lambda_{11} a_3 - \lambda_{32} \lambda_{13} a_1 - \lambda_{33} \lambda_{11} a_2 + \lambda_{33} \lambda_{12} a_1 \\ -\lambda_{22} \lambda_{11} a_3 + \lambda_{23} \lambda_{11} a_2 + \lambda_{21} \lambda_{12} a_3 - \lambda_{23} \lambda_{12} a_1 - \lambda_{21} \lambda_{13} a_2 + \lambda_{22} \lambda_{13} a_1 \end{pmatrix}.$$

Wir berechnen nun $\mathbf{\Lambda}^T \mathbf{c}$. Die erste Zeile ergibt

$$\begin{aligned}
(\mathbf{\Lambda}^T \mathbf{c})_1 &= \lambda_{11} (-\lambda_{32} \lambda_{21} a_3 + \lambda_{33} \lambda_{21} a_2 + \lambda_{31} \lambda_{22} a_3 - \lambda_{33} \lambda_{22} a_1 - \lambda_{31} \lambda_{23} a_2 + \lambda_{32} \lambda_{23} a_1) \\
&\quad + \lambda_{21} (-\lambda_{31} \lambda_{12} a_3 + \lambda_{31} \lambda_{13} a_2 + \lambda_{32} \lambda_{11} a_3 - \lambda_{32} \lambda_{13} a_1 - \lambda_{33} \lambda_{11} a_2 + \lambda_{33} \lambda_{12} a_1) \\
&\quad + \lambda_{31} (-\lambda_{22} \lambda_{11} a_3 + \lambda_{23} \lambda_{11} a_2 + \lambda_{21} \lambda_{12} a_3 - \lambda_{23} \lambda_{12} a_1 - \lambda_{21} \lambda_{13} a_2 + \lambda_{22} \lambda_{13} a_1) \\
&= a_1 (-\lambda_{11} \lambda_{33} \lambda_{22} + \lambda_{11} \lambda_{32} \lambda_{23} - \lambda_{21} \lambda_{32} \lambda_{13} + \lambda_{21} \lambda_{33} \lambda_{12} - \lambda_{31} \lambda_{23} \lambda_{12} + \lambda_{31} \lambda_{22} \lambda_{13}) \\
&\quad + a_2 (\lambda_{11} \lambda_{33} \lambda_{21} - \lambda_{11} \lambda_{31} \lambda_{23} + \lambda_{21} \lambda_{31} \lambda_{13} - \lambda_{21} \lambda_{33} \lambda_{11} + \lambda_{31} \lambda_{23} \lambda_{11} - \lambda_{31} \lambda_{21} \lambda_{13}) \\
&\quad + a_3 (-\lambda_{11} \lambda_{32} \lambda_{21} + \lambda_{11} \lambda_{31} \lambda_{22} - \lambda_{21} \lambda_{31} \lambda_{12} + \lambda_{21} \lambda_{32} \lambda_{11} - \lambda_{31} \lambda_{22} \lambda_{11} + \lambda_{31} \lambda_{21} \lambda_{12}) \\
&= a_1 \cdot \det \mathbf{\Lambda} + a_2 \cdot 0 + a_3 \cdot 0 = a_1.
\end{aligned}$$

Die beiden anderen Zeilen ergeben analog a_2 bzw. a_3 . Somit gilt $\mathbf{\Lambda}^T \mathbf{c} = \mathbf{a} \Leftrightarrow \mathbf{c} = \mathbf{\Lambda a}$. Damit ist die Behauptung gezeigt. □

Bemerkung A.7

Das vorhergehende Lemma gilt nur für Rotationen $\mathbf{\Lambda} \in SO_3$, d.h. sowohl die Orthonormalität der Zeilen- bzw. Spaltenvektoren, als auch die Determinante $\det \mathbf{\Lambda} = 1$ sind notwendige Bedingungen. Dies zeigen die folgenden Beispiele.

- Sei $\mathbf{\Lambda} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$ und $\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$. Dann gilt $\mathbf{\Lambda} \mathbf{\Lambda}^T = I_3$, allerdings ist $\det \mathbf{\Lambda} = -1$. Nach-

rechnen ergibt einerseits

$$\mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} = \begin{pmatrix} 0 & -a_3 & -a_2 \\ a_3 & 0 & a_1 \\ a_2 & -a_1 & 0 \end{pmatrix},$$

andererseits jedoch

$$\mathbf{\Lambda} \mathbf{a} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ -a_3 \end{pmatrix} \implies \widehat{\mathbf{\Lambda}} \mathbf{a} = \begin{pmatrix} 0 & a_3 & a_2 \\ -a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix}.$$

Also gilt das obige Lemma nicht.

- Sei $\mathbf{\Lambda} = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}$ und $\mathbf{a} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$. Nun ist $\det \mathbf{\Lambda} = 1$ und auch die Spaltenvektoren stehen senkrecht zueinander, allerdings sind sie nicht normiert. Dann folgt zum einen

$$\mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^T = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 0 & -\frac{1}{2}a_3 & a_2 \\ \frac{1}{2}a_3 & 0 & -2a_1 \\ -a_2 & 2a_1 & 0 \end{pmatrix},$$

allerdings gilt

$$\mathbf{\Lambda} \mathbf{a} = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} \frac{1}{2}a_1 \\ a_2 \\ 2a_3 \end{pmatrix} \implies \widehat{\mathbf{\Lambda}} \mathbf{a} = \begin{pmatrix} 0 & -2a_3 & a_2 \\ 2a_3 & 0 & -\frac{1}{2}a_1 \\ -a_2 & \frac{1}{2}a_1 & 0 \end{pmatrix}.$$

Auch der Versuch die Matrix $\mathbf{\Lambda}$ zu invertieren, anstatt zu transponieren, was nun nicht mehr das Gleiche ergibt, schlägt fehl.

$$\mathbf{\Lambda} \widehat{\mathbf{A}} \mathbf{\Lambda}^{-1} = \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \begin{pmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} \end{pmatrix} = \begin{pmatrix} 0 & -\frac{1}{2}a_3 & \frac{1}{4}a_2 \\ 2a_3 & 0 & -\frac{1}{2}a_1 \\ -4a_2 & 2a_1 & 0 \end{pmatrix}$$

A.2 Beweis zu Satz 2.3 der Rotationsabbildung

Beweis. Der Beweis entstammt der Arbeit von Argyris [Arg82], Kapitel 1.

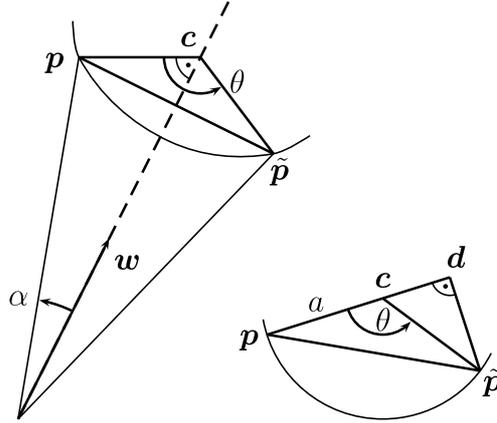
Wir zeigen zuerst Gleichung (2.2). Dabei beschränken wir uns dabei auf Rotationsvektoren \mathbf{w} mit $\|\mathbf{w}\| < \pi$. Sei $\mathbf{p} \in \mathbb{R}^3$ beliebig. $\tilde{\mathbf{p}} := \mathbf{\Lambda} \mathbf{p}$ ist der durch $\mathbf{\Lambda}$ rotierte Vektor. Ziel ist es $\tilde{\mathbf{p}}$ in Abhängigkeit von \mathbf{p} und \mathbf{w} darzustellen und somit die Matrix $\mathbf{\Lambda}$ zu berechnen.

Der Vektor $\mathbf{c} := \frac{1}{\theta^2} (\mathbf{p}^T \mathbf{w}) \mathbf{w}$ zeigt in Richtung \mathbf{w} . Die drei Punkte \mathbf{p} , $\tilde{\mathbf{p}}$, \mathbf{c} liegen somit in einer Ebene, die orthogonal zu \mathbf{w} liegt. Siehe dazu auch Abbildung A.1.

Wir bezeichnen mit $a := \|\mathbf{p} - \mathbf{c}\|$ den Abstand der Punkte \mathbf{p} und \mathbf{c} . Es gilt offensichtlich auch $\|\tilde{\mathbf{p}} - \mathbf{c}\| = a$. Der Rotationswinkel θ liegt nun zwischen den beiden Vektoren $\mathbf{p} - \mathbf{c}$ und $\tilde{\mathbf{p}} - \mathbf{c}$. Es gilt $\cos \theta = \frac{1}{a^2} (\mathbf{p} - \mathbf{c})^T (\tilde{\mathbf{p}} - \mathbf{c})$.

Wir definieren in dieser Ebene zusätzlich den Punkt $\mathbf{d} := \frac{1}{a^2} \left((\mathbf{p} - \tilde{\mathbf{p}})^T (\mathbf{p} - \mathbf{c}) \right) (\mathbf{p} - \mathbf{c})$, sodass die Punkte \mathbf{p} , \mathbf{c} und \mathbf{d} auf einer Linie liegen und außerdem $(\mathbf{p} - \mathbf{c}) \perp (\tilde{\mathbf{p}} - \mathbf{c})$ gilt. Es folgt $\|\tilde{\mathbf{p}} - \mathbf{d}\| = a \sin \theta$ und $\|\mathbf{d} - \mathbf{p}\| = a - a \cos \theta$.

Der Vektor $\tilde{\mathbf{p}} - \mathbf{d}$ steht senkrecht zu \mathbf{w} und zu \mathbf{p} , somit gilt $(\tilde{\mathbf{p}} - \mathbf{d}) \parallel (\mathbf{w} \times \mathbf{p})$. Die Länge von $\|\mathbf{w} \times \mathbf{p}\| = \theta \|\mathbf{p}\| \sin \alpha = \theta a$, wobei α der Winkel zwischen \mathbf{w} und \mathbf{p} ist. Somit folgt $\tilde{\mathbf{p}} - \mathbf{d} = \frac{\sin \theta}{\theta} (\mathbf{w} \times \mathbf{p})$.

Abbildung A.1: Geometrische Konstruktion der Rotationsmatrix Λ

Der Vektor $\mathbf{d} - \mathbf{p}$ steht senkrecht zu \mathbf{w} und zu $\mathbf{w} \times \mathbf{p}$, somit gilt nun $(\mathbf{d} - \mathbf{p}) \parallel (\mathbf{w} \times (\mathbf{w} \times \mathbf{p}))$. Die Länge von $\|\mathbf{w} \times (\mathbf{w} \times \mathbf{p})\| = \theta^2 a$, da $\mathbf{w} \perp (\mathbf{w} \times \mathbf{p})$. Somit gilt $\mathbf{d} - \mathbf{p} = \frac{1 - \cos \theta}{\theta^2} (\mathbf{w} \times (\mathbf{w} \times \mathbf{p}))$. Es folgt dann

$$\tilde{\mathbf{p}} = \Lambda \mathbf{p} = \mathbf{p} + (\tilde{\mathbf{p}} - \mathbf{d}) + (\mathbf{d} - \mathbf{p}) = \mathbf{p} + \frac{\sin \theta}{\theta} (\mathbf{w} \times \mathbf{p}) + \frac{1 - \cos \theta}{\theta^2} (\mathbf{w} \times (\mathbf{w} \times \mathbf{p})).$$

Unter Verwendung der Gleichungen (A.1) und (A.2) aus Lemma A.2 hat die Matrix Λ folgende Form

$$\Lambda = \mathbf{I}_3 + \frac{\sin \theta}{\theta} \widehat{\mathbf{W}} + \frac{1 - \cos \theta}{\theta^2} \widehat{\mathbf{W}}^2.$$

Die Gleichungen (2.3) und (2.4) folgen nun direkt durch Ausnutzen der trigonometrischen Beziehung $1 - \cos \theta = 2 \sin^2 \theta/2$ und wiederum Lemma A.2.

Jetzt zeigen wir Gleichung (2.1), die Darstellung der Matrix Λ als Exponentialreihe.

Es gilt $\widehat{\mathbf{W}}^3 = -\theta^2 \widehat{\mathbf{W}}$. Daraus können wir folgende allgemeine Beziehungen ableiten.

$$\widehat{\mathbf{W}}^{2n+1} = (-1)^n \theta^{2n} \widehat{\mathbf{W}}, \quad \widehat{\mathbf{W}}^{2n} = (-1)^{n-1} \theta^{2n-2} \widehat{\mathbf{W}}^2 \quad \forall n \in \mathbb{N}^+$$

Wir setzen dies in die Reihenentwicklung von Sinus und Cosinus ein, womit die Aussage folgt.

$$\begin{aligned} \Lambda &= \mathbf{I}_3 + \frac{\sin \theta}{\theta} \widehat{\mathbf{W}} + \frac{1}{2} \left(\frac{\sin(\theta/2)}{\theta/2} \right)^2 \widehat{\mathbf{W}}^2 \\ &= \mathbf{I}_3 + \frac{1}{\theta} \left(\sum_{n=0}^{\infty} (-1)^n \frac{\theta^{2n+1}}{(2n+1)!} \right) \widehat{\mathbf{W}} + \frac{1}{\theta^2} \left(1 - \sum_{n=0}^{\infty} (-1)^n \frac{\theta^{2n}}{(2n)!} \right) \widehat{\mathbf{W}}^2 \\ &= \mathbf{I}_3 + \left(\sum_{n=0}^{\infty} (-1)^n \frac{\theta^{2n}}{(2n+1)!} \right) \widehat{\mathbf{W}} + \left(- \sum_{n=1}^{\infty} (-1)^n \frac{\theta^{2n-2}}{(2n)!} \right) \widehat{\mathbf{W}}^2 \\ &= \mathbf{I}_3 + \left(\sum_{n=0}^{\infty} \frac{1}{(2n+1)!} \widehat{\mathbf{W}}^{2n+1} \right) + \left(\sum_{n=1}^{\infty} \frac{1}{(2n)!} \widehat{\mathbf{W}}^{2n} \right) \\ &= \sum_{n=0}^{\infty} \frac{1}{n!} \widehat{\mathbf{W}}^n \\ &= \exp \widehat{\mathbf{W}} \end{aligned}$$

Es bleibt noch zu zeigen, dass $\Lambda \in SO_3$, also dass $\Lambda \Lambda^T = \mathbf{I}_3$ und $\det \Lambda = 1$. Mit Gleichung (2.2) kann leicht gezeigt werden, dass $\exp -\widehat{\mathbf{W}} = \Lambda^T$. Somit gilt

$$\mathbf{I}_3 = \exp \mathbf{0} = \exp (\widehat{\mathbf{W}} - \widehat{\mathbf{W}}) = \exp \widehat{\mathbf{W}} (\exp \widehat{\mathbf{W}})^T = \Lambda \Lambda^T.$$

Die Determinante von $\mathbf{\Lambda}$ muss entweder 1 oder -1 sein, wie wir folgender Gleichung entnehmen.

$$1 = \det \mathbf{I}_3 = \det (\mathbf{\Lambda} \mathbf{\Lambda}^T) = \det \mathbf{\Lambda} \det \mathbf{\Lambda}^T = \det^2 \mathbf{\Lambda}$$

Da die Determinante und die Exponentialabbildung stetige Funktionen sind und $1 = \det \mathbf{I}_3 = \det \exp \mathbf{0}$, muss gelten

$$\det \mathbf{\Lambda} = \det \exp \widehat{\mathbf{W}} = 1 \quad \forall \widehat{\mathbf{W}} \in so_3.$$

□

A.3 Beweis zu Satz 3.2 der endlichen Verzerrungen

Beweis. Wir berechnen zuerst die Variation des lokalen Koordinatensystems $\mathbf{a}_i = \overline{\mathbf{\Lambda}} \mathbf{e}_i$. Für $i = 1, 2, 3$ gilt

$$\delta \mathbf{a}_i = \delta (\overline{\mathbf{\Lambda}} \mathbf{e}_i) = \delta \mathbf{\Lambda} \mathbf{\Lambda}_0 \mathbf{e}_i + \underbrace{\mathbf{\Lambda} \delta \mathbf{\Lambda}_0}_{=0} \mathbf{e}_i + \mathbf{\Lambda} \mathbf{\Lambda}_0 \underbrace{\delta \mathbf{e}_i}_{=0} = \delta \mathbf{W} \mathbf{\Lambda} \mathbf{\Lambda}_0 \mathbf{e}_i = \delta \mathbf{W} \mathbf{a}_i = \delta \mathbf{w} \times \mathbf{a}_i. \quad (\text{A.8})$$

Dabei folgern wir aus Satz 2.2, dass die Variation der Rotation $\delta \mathbf{\Lambda} = \delta \widehat{\mathbf{W}} \mathbf{\Lambda}$ ist. Der axiale Vektor zu $\delta \widehat{\mathbf{W}}$ wird wieder mit $\delta \widehat{\mathbf{w}}$ bezeichnet. Analog berechnen wir die Ableitung von \mathbf{a}_i nach dem Weglängenparameter s_0 .

$$\mathbf{a}'_i = (\overline{\mathbf{\Lambda}} \mathbf{e}_i)' = \mathbf{\Lambda}' \mathbf{\Lambda}_0 \mathbf{e}_i + \mathbf{\Lambda} \mathbf{\Lambda}'_0 \mathbf{e}_i + \mathbf{\Lambda} \mathbf{\Lambda}_0 \underbrace{\mathbf{e}'_i}_{=0} = \mathbf{\Lambda}' \mathbf{\Lambda}^T \mathbf{a}_i + \mathbf{\Lambda} \mathbf{\Lambda}'_0 \mathbf{\Lambda}_0^T \mathbf{\Lambda}^T \mathbf{a}_i \quad (\text{A.9})$$

Mit $\widehat{\mathbf{\Omega}} := \mathbf{\Lambda}' \mathbf{\Lambda}^T$ und $\widehat{\mathbf{\Omega}}_0 := \mathbf{\Lambda}'_0 \mathbf{\Lambda}_0^T$ folgt zusammen mit Gleichung (A.7) aus Lemma A.6.

$$\mathbf{a}'_i = \mathbf{\Lambda}' \mathbf{\Lambda}^T \mathbf{a}_i + \mathbf{\Lambda} \mathbf{\Lambda}'_0 \mathbf{\Lambda}_0^T \mathbf{\Lambda}^T \mathbf{a}_i = \widehat{\mathbf{\Omega}} \mathbf{a}_i + \mathbf{\Lambda} \widehat{\mathbf{\Omega}}_0 \mathbf{\Lambda}^T \mathbf{a}_i = \boldsymbol{\omega} \times \mathbf{a}_i + (\mathbf{\Lambda} \boldsymbol{\omega}_0) \times \mathbf{a}_i = \overline{\boldsymbol{\omega}} \times \mathbf{a}_i,$$

wobei $\overline{\boldsymbol{\omega}} := \boldsymbol{\omega} + \mathbf{\Lambda} \boldsymbol{\omega}_0$.

Wir müssen noch die Schiefsymmetrie von $\mathbf{\Omega}$ bzw. $\mathbf{\Omega}_0$ zeigen. Dabei nutzen wir die in Satz 2.3 bewiesene Darstellung $\mathbf{\Lambda} = \exp \widehat{\mathbf{W}}$.

$$\mathbf{\Omega}^T = (\mathbf{\Lambda}' \mathbf{\Lambda}^T)^T = \left((\exp \widehat{\mathbf{W}})' \mathbf{\Lambda}^T \right)^T = \left(\exp \widehat{\mathbf{W}} \widehat{\mathbf{W}}' \mathbf{\Lambda}^T \right)^T = \mathbf{\Lambda} \widehat{\mathbf{W}}'^T \mathbf{\Lambda}^T = -\mathbf{\Lambda} \widehat{\mathbf{W}}' \mathbf{\Lambda}^T = -\mathbf{\Lambda}' \mathbf{\Lambda}^T = -\widehat{\mathbf{\Omega}}$$

Wir definieren $\delta \overline{\boldsymbol{\omega}}$ als Variation von $\overline{\boldsymbol{\omega}}$. Außerdem nehmen wir an, dass die Variation δ und die Ableitung nach der Bogenlänge s_0 vertauschbar sind $\delta (\mathbf{a}'_i) = (\delta \mathbf{a}_i)'$. Dann folgt mit der Jacobi-Identität (A.4), sowie (A.8) und (A.9) von oben

$$\begin{aligned} (\delta \overline{\boldsymbol{\omega}} - \delta \mathbf{w} \times \overline{\boldsymbol{\omega}}) \times \mathbf{a}_i &= \delta \overline{\boldsymbol{\omega}} \times \mathbf{a}_i - (\delta \mathbf{w} \times \overline{\boldsymbol{\omega}}) \times \mathbf{a}_i \\ &= \delta \overline{\boldsymbol{\omega}} \times \mathbf{a}_i + \overline{\boldsymbol{\omega}} \times (\delta \mathbf{w} \times \mathbf{a}_i) - \delta \mathbf{w} \times (\overline{\boldsymbol{\omega}} \times \mathbf{a}_i) \\ &= \delta \overline{\boldsymbol{\omega}} \times \mathbf{a}_i + \overline{\boldsymbol{\omega}} \times \delta \mathbf{a}_i - \delta \mathbf{w} \times \mathbf{a}'_i \\ &= \delta (\overline{\boldsymbol{\omega}} \times \mathbf{a}_i) - \delta \mathbf{w} \times \mathbf{a}_i \\ &= \delta (\mathbf{a}'_i) - \delta \mathbf{w} \times \mathbf{a}_i \\ &= (\delta \mathbf{a}_i)' - \delta \mathbf{w} \times \mathbf{a}_i \\ &= (\delta \mathbf{w} \times \mathbf{a}_i)' - (\delta \mathbf{w} \times \mathbf{a}_i)' + \delta \mathbf{w}' \times \mathbf{a}_i \\ &= \delta \mathbf{w}' \times \mathbf{a}_i. \end{aligned}$$

Da \mathbf{a}_i ($i = 1, 2, 3$) eine Basis des \mathbb{R}^3 ist, folgt direkt

$$\delta \overline{\boldsymbol{\omega}} - \delta \mathbf{w} \times \overline{\boldsymbol{\omega}} = \delta \mathbf{w}'. \quad (\text{A.10})$$

Außerdem gilt

$$\delta \mathbf{\Lambda} \boldsymbol{\omega}_0 = \delta \widehat{\mathbf{W}} \mathbf{\Lambda} \boldsymbol{\omega}_0 = \delta \mathbf{w} \times (\mathbf{\Lambda} \boldsymbol{\omega}_0).$$

Einsetzen in (A.10) ergibt dann

$$\delta \mathbf{w}' = \delta \bar{\boldsymbol{\omega}} - \delta \mathbf{w} \times \bar{\boldsymbol{\omega}} = \delta \boldsymbol{\omega} + \delta \boldsymbol{\Lambda} \boldsymbol{\omega}_0 - \delta \mathbf{w} \times (\boldsymbol{\omega} + \boldsymbol{\Lambda} \boldsymbol{\omega}_0) = \delta \boldsymbol{\omega} - \delta \mathbf{w} \times \boldsymbol{\omega}. \quad (\text{A.11})$$

Somit folgt für die virtuellen Verzerrungen $\delta \tilde{\boldsymbol{\varepsilon}}$ und $\delta \tilde{\boldsymbol{\kappa}}$ aus Kapitel 3.6, Gleichung (3.4), zusammen mit den obigen Gleichungen (A.8) und (A.11)

$$\delta \tilde{\boldsymbol{\varepsilon}} = \delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}' - \delta \mathbf{a}_1 + \delta \mathbf{w} \times \mathbf{a}_1, \quad \delta \tilde{\boldsymbol{\kappa}} = \delta \boldsymbol{\omega} - \delta \mathbf{w} \times \boldsymbol{\omega}. \quad (\text{A.12})$$

Dies kann jetzt noch einmal umgeformt werden, um die Abhängigkeiten von $\delta \tilde{\boldsymbol{\varepsilon}}$ und $\delta \tilde{\boldsymbol{\kappa}}$ zu verdeutlichen.

$$\begin{aligned} \delta \tilde{\boldsymbol{\varepsilon}} &= \delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}' - \delta \mathbf{a}_1 + \delta \mathbf{w} \times \mathbf{a}_1 & \delta \tilde{\boldsymbol{\kappa}} &= \delta \boldsymbol{\omega} - \delta \mathbf{w} \times \boldsymbol{\omega} \\ &= -\delta \mathbf{w} \times \bar{\mathbf{u}}' + \delta \mathbf{w} \times \mathbf{a}_1 + \delta \mathbf{u}' - \delta \mathbf{a}_1 & &= \boldsymbol{\Lambda} \boldsymbol{\Lambda}^T \delta \mathbf{W}^T \boldsymbol{\omega} + \delta \boldsymbol{\omega} \\ &= \boldsymbol{\Lambda} \boldsymbol{\Lambda}^T \delta \mathbf{W}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + \delta \mathbf{u}' - \delta \mathbf{a}_1 & &= \boldsymbol{\Lambda} \delta \boldsymbol{\Lambda}^T \boldsymbol{\omega} + \boldsymbol{\Lambda} \boldsymbol{\Lambda}^T \delta \boldsymbol{\omega} \\ &= \boldsymbol{\Lambda} \delta \boldsymbol{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + \boldsymbol{\Lambda} \boldsymbol{\Lambda}^T \delta (\bar{\mathbf{u}}' - \mathbf{a}_1) & &= \boldsymbol{\Lambda} \delta (\boldsymbol{\Lambda}^T \boldsymbol{\omega}) \\ &= \boldsymbol{\Lambda} \delta (\boldsymbol{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1)) & &= \mathcal{L}(\boldsymbol{\omega}) \\ &= \mathcal{L}(\bar{\mathbf{u}}' - \mathbf{a}_1) \end{aligned}$$

Dabei ist \mathcal{L} eine Abbildung, die die Koordinatentransformationen (“*push-forward*” und “*pull-back*”) übernimmt. Damit folgt die Behauptung. \square

A.4 Beweis zu Satz 3.3 der ersten Potentialvariation

Beweis. Wir wählen eine Kurve $\gamma \in \mathbb{R}^3 \times SO_3$ für die Berechnung der ersten kovarianten Ableitung. Der Vektor \mathbf{u} wird durch die aufaddierte virtuelle Verschiebung $\delta \mathbf{u}$ variiert. Wir definieren hierzu

$$\mathbf{u}_\alpha := \mathbf{u} + \alpha \delta \mathbf{u} \quad \alpha > 0.$$

Bei $\boldsymbol{\Lambda} \in SO_3$ müssen wir die zur virtuellen Verdrehung $\delta \mathbf{w}$ gehörende Rotationsmatrix von links anmultiplizieren, um eine Kurve in SO_3 zu erhalten und somit die Variation berechnen zu können. Die variierte Konfiguration hat folgende Form

$$\boldsymbol{\Lambda}_\alpha := \exp\left(\alpha \delta \widehat{\mathbf{W}}\right) \boldsymbol{\Lambda} \quad \alpha > 0.$$

Da nur der durch die Belastung ausgelenkte Anteil variiert wird, folgt für die Gesamtkonfiguration

$$\begin{aligned} \bar{\mathbf{u}}_\alpha &= \mathbf{u}_0 + \mathbf{u}_\alpha = \mathbf{u}_0 + \mathbf{u} + \alpha \delta \mathbf{u} = \bar{\mathbf{u}} + \alpha \delta \mathbf{u} \\ \bar{\boldsymbol{\Lambda}}_\alpha &= \boldsymbol{\Lambda}_\alpha \boldsymbol{\Lambda}_0 = \exp\left(\alpha \delta \widehat{\mathbf{W}}\right) \boldsymbol{\Lambda} \boldsymbol{\Lambda}_0 = \exp\left(\alpha \delta \widehat{\mathbf{W}}\right) \bar{\boldsymbol{\Lambda}}. \end{aligned}$$

Die Kurve γ ist nun folgendermaßen definiert.

$$\gamma := (\bar{\mathbf{u}}_\alpha, \bar{\boldsymbol{\Lambda}}_\alpha) = \left(\bar{\mathbf{u}} + \alpha \delta \mathbf{u}, \exp\left(\alpha \delta \widehat{\mathbf{W}}\right) \bar{\boldsymbol{\Lambda}}\right)$$

Die Glattheit der Kurve ist ersichtlich, wenn wir SO_3 mit der, von der Spektralnorm stammenden, Metrik betrachten. Einsetzen ergibt

$$\gamma \Big|_{\alpha=0} = (\bar{\mathbf{u}}, \bar{\boldsymbol{\Lambda}}) \quad \frac{\partial}{\partial \alpha} \gamma \Big|_{\alpha=0} = (\delta \mathbf{u}, \delta \widehat{\mathbf{W}} \bar{\boldsymbol{\Lambda}}).$$

Die erste Variation kann durch

$$\delta \Pi(\bar{\boldsymbol{\Phi}}) = \lim_{\alpha \rightarrow 0} \frac{\partial}{\partial \alpha} \Pi(\gamma)$$

berechnet werden. Da im fraglichen Grenzübergang sowohl $\bar{\mathbf{u}}$ und $\bar{\boldsymbol{\Lambda}}$ also auch Π stetig sind, kann anstatt dem Limes auch die Funktionsauswertung direkt bei $\alpha = 0$ betrachtet werden. Dabei ist zu

beachten, dass die Reihenfolge der Ableitung nach α sowie der Ableitung nach s_0 , der Integration bzgl. s_0 und der Transponierung vertauscht werden dürfen. Wir berechnen nun $\delta\Pi_{int}$.

$$\begin{aligned}
& \frac{\partial}{\partial\alpha} \Pi_{int}(\bar{\Phi}_\alpha) \\
&= \frac{\partial}{\partial\alpha} \frac{1}{2} \int_0^{L_0} (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda}_\alpha^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) + \boldsymbol{\omega}_\alpha^T \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha \, ds_0 \\
&= \frac{1}{2} \int_0^{L_0} \left[\frac{\partial}{\partial\alpha} (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \right] \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda}_\alpha^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \left[\frac{\partial}{\partial\alpha} \bar{\Lambda}_\alpha \right] \mathbf{C}_n \bar{\Lambda}_\alpha^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \bar{\Lambda}_\alpha \mathbf{C}_n \left[\frac{\partial}{\partial\alpha} \bar{\Lambda}_\alpha^T \right] (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda}_\alpha^T \left[\frac{\partial}{\partial\alpha} (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \right] \\
&\quad + \left[\frac{\partial}{\partial\alpha} \boldsymbol{\omega}_\alpha^T \right] \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha + \boldsymbol{\omega}_\alpha^T \left[\frac{\partial}{\partial\alpha} \bar{\Lambda}_\alpha \right] \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha \\
&\quad + \boldsymbol{\omega}_\alpha^T \bar{\Lambda}_\alpha \mathbf{C}_m \left[\frac{\partial}{\partial\alpha} \bar{\Lambda}_\alpha^T \right] \boldsymbol{\omega}_\alpha + \boldsymbol{\omega}_\alpha^T \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda}_\alpha^T \left[\frac{\partial}{\partial\alpha} \boldsymbol{\omega}_\alpha \right] \, ds_0 \\
&= \frac{1}{2} \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{a}_1)^T \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda}_\alpha^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right) \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda} \delta\widehat{\mathbf{W}}^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right)^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda}_\alpha^T (\delta\mathbf{u}' - \delta\mathbf{a}_1) \\
&\quad + \delta\boldsymbol{\omega}^T \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha + \boldsymbol{\omega}_\alpha^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right) \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\omega}_\alpha \\
&\quad + \boldsymbol{\omega}_\alpha^T \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda} \delta\widehat{\mathbf{W}}^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right)^T \boldsymbol{\omega}_\alpha + \boldsymbol{\omega}_\alpha^T \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda}_\alpha^T \delta\boldsymbol{\omega} \, ds_0 \\
&= \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{a}_1)^T \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda}_\alpha^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right) \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + \delta\boldsymbol{\omega}^T \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha + \boldsymbol{\omega}_\alpha^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right) \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha \, ds_0.
\end{aligned}$$

Der Grenzübergang $\alpha \rightarrow 0$ ergibt dann wegen der Stetigkeit aller Funktionen bzgl. α

$$\begin{aligned}
& \lim_{\alpha \rightarrow 0} \frac{\partial}{\partial\alpha} \Pi_{int}(\bar{\Phi}_\alpha) \\
&= \lim_{\alpha \rightarrow 0} \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{a}_1)^T \bar{\Lambda}_\alpha \mathbf{C}_n \bar{\Lambda}_\alpha^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1})^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right) \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}'_\alpha - \mathbf{a}_{\alpha 1}) \\
&\quad + \delta\boldsymbol{\omega}^T \bar{\Lambda}_\alpha \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha + \boldsymbol{\omega}_\alpha^T \exp\left(\alpha \delta\widehat{\mathbf{W}}\right) \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}_\alpha^T \boldsymbol{\omega}_\alpha \, ds_0 \\
&= \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{a}_1)^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + (\bar{\mathbf{u}}' - \mathbf{a}_1)^T \mathbf{I}_3 \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) \\
&\quad + \delta\boldsymbol{\omega}^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\omega} + \boldsymbol{\omega}^T \mathbf{I}_3 \delta\widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\omega} \, ds_0 \\
&= \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{a}_1 + \delta\mathbf{W}^T \bar{\mathbf{u}}' - \delta\mathbf{W}^T \mathbf{a}_1)^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + (\delta\boldsymbol{\omega} + \delta\mathbf{W}^T \boldsymbol{\omega})^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\omega} \, ds_0 \\
&= \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{a}_1 - \delta\boldsymbol{\omega} \times \bar{\mathbf{u}}' + \delta\boldsymbol{\omega} \times \mathbf{a}_1)^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + (\delta\boldsymbol{\omega} - \delta\boldsymbol{\omega} \times \boldsymbol{\omega})^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\omega} \, ds_0 \\
&= \int_0^{L_0} \delta\tilde{\boldsymbol{\varepsilon}}^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \boldsymbol{\varepsilon} + \delta\tilde{\boldsymbol{\kappa}}^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\kappa} \, ds_0 = \int_0^{L_0} \delta\tilde{\boldsymbol{\varepsilon}}^T \mathbf{n} + \delta\tilde{\boldsymbol{\kappa}}^T \mathbf{m} \, ds_0.
\end{aligned}$$

□

A.5 Beweis zu Satz 3.3 der zweiten Potentialvariation

Beweis. Um das Variationsprinzip anwenden zu können, wird die erste Variation wie folgt dargestellt

$$\delta\Pi_{int} = \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}')^T \bar{\mathbf{\Lambda}} \mathbf{C}_n \bar{\mathbf{\Lambda}}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + \delta\mathbf{w}'^T \bar{\mathbf{\Lambda}} \mathbf{C}_m \bar{\mathbf{\Lambda}}^T \boldsymbol{\omega} ds_0.$$

Wir verfahren wie beim obigen Beweis zur Berechnung der ersten Potentialvariation. Die variierte Konfiguration wird durch $\bar{\Phi}_\beta := \tilde{\gamma} = (\bar{\mathbf{u}}_\beta, \bar{\mathbf{\Lambda}}_\beta)$ dargestellt, wobei für $\bar{\mathbf{u}}_\beta$ und $\bar{\mathbf{\Lambda}}_\beta$ gilt

$$\begin{aligned} \bar{\mathbf{u}}_\beta &= \mathbf{u}_0 + \mathbf{u}_\beta = \mathbf{u}_0 + \mathbf{u} + \beta \Delta \mathbf{u} = \bar{\mathbf{u}} + \beta \Delta \mathbf{u} \\ \bar{\mathbf{\Lambda}}_\beta &= \mathbf{\Lambda}_\beta \mathbf{\Lambda}_0 = \exp\left(\beta \Delta \widehat{\mathbf{W}}\right) \mathbf{\Lambda} \mathbf{\Lambda}_0 = \exp\left(\beta \Delta \widehat{\mathbf{W}}\right) \bar{\mathbf{\Lambda}}. \end{aligned}$$

Die Glattheit der Kurve $\tilde{\gamma}$ der Wert $\tilde{\gamma}|_{\beta=0}$ sowie die Ableitung $\frac{\partial}{\partial \beta} \tilde{\gamma}|_{\beta=0}$ folgen wie im vorhergehenden Beweis. Die zweite Variation ist definiert durch

$$\Delta \delta \Pi(\bar{\Phi}, \delta \Phi, \Delta \Phi) := \lim_{\beta \rightarrow 0} \frac{\partial}{\partial \beta} \delta \Pi(\bar{\Phi}_\beta, \delta \Phi).$$

Zur Berechnung der zweiten Ableitung nehmen wir an, dass die Richtung $\delta \Phi$ der ersten Ableitung unabhängig von β ist.

Berechne $\Delta \delta \Pi_{int}$.

$$\begin{aligned} & \frac{\partial}{\partial \beta} \delta \Pi_{int}(\bar{\Phi}_\beta) \\ &= \frac{\partial}{\partial \beta} \int_0^{L_0} (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_n \bar{\mathbf{\Lambda}}_\beta^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) + \delta\mathbf{w}'^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_m \bar{\mathbf{\Lambda}}_\beta^T \boldsymbol{\omega}_\beta ds_0 \\ &= \int_0^{L_0} \left[\frac{\partial}{\partial \beta} (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \right] \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_n \bar{\mathbf{\Lambda}}_\beta^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) \\ & \quad + (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \left[\frac{\partial}{\partial \beta} \bar{\mathbf{\Lambda}}_\beta \right] \mathbf{C}_n \bar{\mathbf{\Lambda}}_\beta^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) \\ & \quad + (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_n \left[\frac{\partial}{\partial \beta} \bar{\mathbf{\Lambda}}_\beta^T \right] (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) \\ & \quad + (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_n \bar{\mathbf{\Lambda}}_\beta^T \left[\frac{\partial}{\partial \beta} (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) \right] \\ & \quad + \delta\mathbf{w}'^T \left[\frac{\partial}{\partial \beta} \bar{\mathbf{\Lambda}}_\beta \right] \mathbf{C}_m \bar{\mathbf{\Lambda}}_\beta^T \boldsymbol{\omega}_\beta + \delta\mathbf{w}'^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_m \left[\frac{\partial}{\partial \beta} \bar{\mathbf{\Lambda}}_\beta^T \right] \boldsymbol{\omega}_\beta + \delta\mathbf{w}'^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_m \bar{\mathbf{\Lambda}}_\beta^T \left[\frac{\partial}{\partial \beta} \boldsymbol{\omega}_\beta \right] ds_0 \\ &= \int_0^{L_0} (-\delta\mathbf{w} \times \Delta \mathbf{u}')^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_n \bar{\mathbf{\Lambda}}_\beta^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) \\ & \quad + (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \exp\left(\beta \Delta \widehat{\mathbf{W}}\right) \Delta \widehat{\mathbf{W}} \bar{\mathbf{\Lambda}} \mathbf{C}_n \bar{\mathbf{\Lambda}}_\beta^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) \\ & \quad + (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_n \bar{\mathbf{\Lambda}}^T \Delta \widehat{\mathbf{W}}^T \exp\left(\beta \Delta \widehat{\mathbf{W}}\right)^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta_1}) \\ & \quad + (\delta\mathbf{u}' - \delta\mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_n \bar{\mathbf{\Lambda}}_\beta^T (\Delta \mathbf{u}' - \Delta \mathbf{a}_1) \\ & \quad + \delta\mathbf{w}'^T \exp\left(\beta \Delta \widehat{\mathbf{W}}\right) \Delta \widehat{\mathbf{W}} \bar{\mathbf{\Lambda}} \mathbf{C}_m \bar{\mathbf{\Lambda}}_\beta^T \boldsymbol{\omega}_\beta + \delta\mathbf{w}'^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_m \bar{\mathbf{\Lambda}}^T \Delta \widehat{\mathbf{W}}^T \exp\left(\beta \Delta \widehat{\mathbf{W}}\right)^T \boldsymbol{\omega}_\beta \\ & \quad + \delta\mathbf{w}'^T \bar{\mathbf{\Lambda}}_\beta \mathbf{C}_m \bar{\mathbf{\Lambda}}_\beta^T \Delta \boldsymbol{\omega} ds_0 \end{aligned}$$

Der Grenzübergang $\beta \rightarrow 0$ ergibt dann wegen der Stetigkeit aller Funktionen bzgl. β

$$\begin{aligned}
& \lim_{\beta \rightarrow 0} \frac{\partial}{\partial \beta} \delta \Pi_{int}(\bar{\Phi}_\beta, \delta \Phi) \\
= & \lim_{\beta \rightarrow 0} \int_0^{L_0} (-\delta \mathbf{w} \times \Delta \mathbf{u}')^T \bar{\Lambda}_\beta \mathbf{C}_n \bar{\Lambda}_\beta^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta 1}) \\
& + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \exp(\beta \Delta \widehat{\mathbf{W}}) \Delta \widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta 1}) \\
& + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \bar{\Lambda}_\beta \mathbf{C}_n \bar{\Lambda}^T \Delta \widehat{\mathbf{W}}^T \exp(\beta \Delta \widehat{\mathbf{W}})^T (\bar{\mathbf{u}}'_\beta - \mathbf{a}_{\beta 1}) \\
& + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}'_\beta)^T \bar{\Lambda}_\beta \mathbf{C}_n \bar{\Lambda}_\beta^T (\Delta \mathbf{u}' - \Delta \mathbf{a}_1) \\
& + \delta \mathbf{w}'^T \exp(\beta \Delta \widehat{\mathbf{W}}) \Delta \widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}_\beta^T \boldsymbol{\omega}_\beta + \delta \mathbf{w}'^T \bar{\Lambda}_\beta \mathbf{C}_m \bar{\Lambda}^T \Delta \widehat{\mathbf{W}}^T \exp(\beta \Delta \widehat{\mathbf{W}})^T \boldsymbol{\omega}_\beta \\
& + \delta \mathbf{w}'^T \bar{\Lambda}_\beta \mathbf{C}_m \bar{\Lambda}_\beta^T \Delta \boldsymbol{\omega} \, ds_0 \\
= & \int_0^{L_0} (-\delta \mathbf{w} \times \Delta \mathbf{u}')^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}')^T \mathbf{I}_3 \Delta \widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) \\
& + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}')^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \Delta \widehat{\mathbf{W}}^T \mathbf{I}_3^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}')^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\Delta \mathbf{u}' - \Delta \mathbf{a}_1) \\
& + \delta \mathbf{w}'^T \mathbf{I}_3 \Delta \widehat{\mathbf{W}} \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\omega} + \delta \mathbf{w}'^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \Delta \widehat{\mathbf{W}}^T \mathbf{I}_3^T \boldsymbol{\omega} + \delta \mathbf{w}'^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \Delta \boldsymbol{\omega} \, ds_0 \\
= & \int_0^{L_0} (\Delta \mathbf{u}' \times \delta \mathbf{w})^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \boldsymbol{\varepsilon} + \left(\Delta \widehat{\mathbf{W}}^T (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}') \right)^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \boldsymbol{\varepsilon} \\
& + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}')^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \Delta \widehat{\mathbf{W}}^T (\bar{\mathbf{u}}' - \mathbf{a}_1) + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}')^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\Delta \mathbf{u}' - \Delta \mathbf{a}_1) \\
& + \left(\Delta \widehat{\mathbf{W}}^T \delta \mathbf{w}' \right)^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \boldsymbol{\kappa} + \delta \mathbf{w}'^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \Delta \widehat{\mathbf{W}}^T \boldsymbol{\omega} + \delta \mathbf{w}'^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \Delta \boldsymbol{\omega} \, ds_0 \\
= & \int_0^{L_0} (\Delta \mathbf{u}' \times \delta \mathbf{w} - \Delta \mathbf{w} \times \delta \mathbf{u}' + \Delta \mathbf{w} \times (\delta \mathbf{w} \times \bar{\mathbf{u}}'))^T \mathbf{n} \\
& + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}')^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (-\Delta \mathbf{w} \times \bar{\mathbf{u}}' + \Delta \mathbf{w} \times \mathbf{a}_1 + \Delta \mathbf{u}' - \Delta \mathbf{a}_1) \\
& + (-\Delta \mathbf{w} \times \delta \mathbf{w}')^T \mathbf{m} + \delta \mathbf{w}'^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T (-\Delta \mathbf{w} \times \boldsymbol{\omega} + \Delta \boldsymbol{\omega}) \, ds_0 \\
= & \int_0^{L_0} (\Delta \mathbf{u}' \times \delta \mathbf{w} + \delta \mathbf{u}' \times \Delta \mathbf{w} + \Delta \mathbf{w} \times (\delta \mathbf{w} \times \bar{\mathbf{u}}'))^T \mathbf{n} \\
& + (\delta \mathbf{u}' - \delta \mathbf{w} \times \bar{\mathbf{u}}')^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T (\Delta \mathbf{u}' - \Delta \mathbf{w} \times \bar{\mathbf{u}}') \\
& + (\delta \mathbf{w}' \times \Delta \mathbf{w})^T \mathbf{m} + \delta \mathbf{w}'^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \Delta \mathbf{w}' \, ds_0 \\
= & \int_0^{L_0} \delta \boldsymbol{\varepsilon}^T \bar{\Lambda} \mathbf{C}_n \bar{\Lambda}^T \Delta \boldsymbol{\varepsilon} + \delta \boldsymbol{\kappa}^T \bar{\Lambda} \mathbf{C}_m \bar{\Lambda}^T \Delta \boldsymbol{\kappa} \\
& + (\Delta \mathbf{u}' \times \delta \mathbf{w} + \delta \mathbf{u}' \times \Delta \mathbf{w} + \Delta \mathbf{w} \times (\delta \mathbf{w} \times \bar{\mathbf{u}}'))^T \mathbf{n} + (\delta \mathbf{w}' \times \Delta \mathbf{w})^T \mathbf{m} \, ds_0.
\end{aligned}$$

Es bleibt noch zu zeigen, dass $\Delta \delta \Pi_{ext} = 0$.

Wenn die vektorwertigen Funktionen \mathbf{p} und \mathbf{q} nicht von β abhängen, gilt

$$\Delta \delta \Pi_{ext}(\bar{\Phi}, \delta \Phi, \Delta \Phi) = \lim_{\beta \rightarrow 0} \frac{\partial}{\partial \beta} \delta \Pi_{ext}(\bar{\Phi}_\beta, \delta \Phi) = \lim_{\beta \rightarrow 0} \frac{\partial}{\partial \beta} \int_0^{L_0} \delta \mathbf{u}'^T \mathbf{p} + \delta \mathbf{w}'^T \mathbf{q} \, ds_0 = \lim_{\beta \rightarrow 0} \int_0^{L_0} 0 \, ds_0 = 0.$$

□

Anhang B

Materialparameter

elmt33

2 Knoten, Materialparameter $d(6)$

$$\begin{array}{ccc} EA & GA_2 & GA_3 \\ GI_t & EI_2 & EI_3 \end{array}$$

Bedeutung der einzelnen Parameter:

- $EA = d(1)$, Dehnsteifigkeit
- $GA_2 = d(2)$, Schubsteifigkeit in \mathbf{a}_2 Richtung
- $GA_3 = d(3)$, Schubsteifigkeit in \mathbf{a}_3 Richtung
- $GI_t = d(4)$, Torsionssteifigkeit
- $EI_2 = d(5)$, Biegesteifigkeit in \mathbf{a}_2 Richtung
- $EI_3 = d(6)$, Biegesteifigkeit in \mathbf{a}_3 Richtung

elmt82 und *elmt86*

2 Knoten, Materialparameter $d(8)$, bei *elmt86* zusätzlich 2 Knoten zur Speicherverwaltung

$$\begin{array}{ccc} EA & GA_2 & GA_3 \\ GI_t & EI_2 & EI_3 \\ T_{mode} & R_{mode} & \end{array}$$

Bedeutung der zusätzlichen Parameter:

- $T_{mode} = d(7)$, Modus zur Berechnung der tangentialen Steifigkeitsmatrix
 - $T_{mode} = 0$, nur materielle tangentialer Steifigkeitsmatrix
 - $T_{mode} = 1$, mit geometrischer tangentialer Steifigkeitsmatrix nach obiger Theorie
 - $T_{mode} = 2$, mit geometrischer tangentialer Steifigkeitsmatrix nach Ibrahimbegovic
 - $T_{mode} = 3$, mit geometrischer tangentialer Steifigkeitsmatrix nach Simo
 - $T_{mode} = 4$, mit symmetrischer geometrischer tangentialer Steifigkeitsmatrix
- $R_{mode} = d(8)$, Modus zur Berechnung des Residuums
 - $R_{mode} = 0$, Berechnung des Residuums über die tangentialer Steifigkeitsmatrix
 - $R_{mode} = 1$, Berechnung des Residuums über die erste Potentialvariation

elmt83 und *elmt87*

3 Knoten, Materialparameter $d(16)$, bei *elmt87* zusätzlich 3 Knoten zur Speicherverwaltung

EA	GA_2	GA_3		
GI_t	EI_2	EI_3		
T_{mode}	R_{mode}			
DIS_{mode}	w_1	w_2	w_3	
REF_{mode}	w_1	w_2	w_3	

Bedeutungen der zusätzlichen Parameter:

- $DIS_{mode} = d(9)$, Wahl der Ansatzfunktionen zur Approximation der Deformationen
 $DIS_{mode} = 0$, polynomialer hierarchischer Ansatz, dann sind w_1 bis w_3 ohne Bedeutung
 $DIS_{mode} = 1$, polynomialer Lagrange Ansatz, dann sind w_1 bis w_3 ohne Bedeutung
 $DIS_{mode} = 2$, rationaler Ansatz, dann sind w_1 bis w_3 die Gewichte der rationalen Funktionen
- $REF_{mode} = d(13)$, Wahl der Ansatzfunktionen zur Approximation der Referenzgeometrie
 $REF_{mode} = 0$, polynomiale hierarchische Geometrie, dann sind w_1 bis w_3 ohne Bedeutung
 $REF_{mode} = 1$, polynomiale Lagrange Geometrie, dann sind w_1 bis w_3 ohne Bedeutung
 $REF_{mode} = 2$, rationale Geometrie, dann sind w_1 bis w_3 die Gewichte der rat. Funktionen
 $REF_{mode} = 3$, Kreisbogen Geometrie, dann sind w_1 bis w_3 ohne Bedeutung

elmt84 und *elmt88*

4 Knoten, Materialparameter $d(18)$, bei *elmt88* zusätzlich 4 Knoten zur Speicherverwaltung

EA	GA_2	GA_3		
GI_t	EI_2	EI_3		
T_{mode}	R_{mode}			
DIS_{mode}	w_1	w_2	w_3	w_4
REF_{mode}	w_1	w_2	w_3	w_4

Bedeutungen der zusätzlichen Parameter:

- $DIS_{mode} = d(9)$, Wahl der Ansatzfunktionen zur Approximation der Deformationen
 $DIS_{mode} = 0$, polynomialer hierarchischer Ansatz, dann sind w_1 bis w_4 ohne Bedeutung
 $DIS_{mode} = 1$, polynomialer Lagrange Ansatz, dann sind w_1 bis w_4 ohne Bedeutung
 $DIS_{mode} = 2$, rationaler Ansatz, dann sind w_1 bis w_4 die Gewichte der rationalen Funktionen
- $REF_{mode} = d(13)$, Wahl der Ansatzfunktionen zur Approximation der Referenzgeometrie
 $REF_{mode} = 0$, polynomiale hierarchische Geometrie, dann sind w_1 bis w_4 ohne Bedeutung
 $REF_{mode} = 1$, polynomiale Lagrange Geometrie, dann sind w_1 bis w_4 ohne Bedeutung
 $REF_{mode} = 2$, rationale Geometrie, dann sind w_1 bis w_4 die Gewichte der Funktionen
 $REF_{mode} = 3$, Spline Ansatz, dann geben $w_1 = c_1$ und $w_2 = c_2$ Anfang und Ende an
 $w_1 = w_2 = 0$, inneres Element
 $w_1 = 1, w_2 = 0$, Anfangselement
 $w_1 = 0, w_2 = 1$, Endelement
 w_3 gibt die Wahl des Parameters χ an
 $w_3 = 0$, dann ist $\chi = 1$
 $w_3 = 1$, dann wird χ entsprechend Kapitel 2.5.3 berechnet
 w_4 ist ohne Bedeutung

elmt90

4 Knoten, Materialparameter $d(7)$

$$\begin{array}{cccc} \epsilon_N & r_1 & r_2 & \\ \epsilon_{T1} & \mu_1 & \epsilon_{T2} & \mu_2 \end{array}$$

Bedeutung der einzelnen Parameter:

- $\epsilon_N = d(1)$, Strafparameter für Kontakt, normal zu den Balkenachsen
- $r_1 = d(2)$, Radius des ersten Balkens, d.h. Knoten 1 und 2
- $r_2 = d(3)$, Radius des zweiten Balkens, d.h. Knoten 3 und 4
- $\epsilon_{T1} = d(4)$, Strafparameter für Kontakt, tangential zum ersten Balken
- $\mu_1 = d(5)$, Reibungskoeffizient am ersten Balken
- $\epsilon_{T2} = d(6)$, Strafparameter für Kontakt, tangential zum zweiten Balken
- $\mu_2 = d(7)$, Reibungskoeffizient am zweiten Balken

Die Parameter in der zweiten Zeile werden bei der aktuellen Implementierung nicht benötigt, da keine Reibung berücksichtigt wird.

elmt91

6 Knoten, Materialparameter $d(24)$

$$\begin{array}{cccc} \epsilon_N & r_1 & r_2 & \\ \epsilon_{T1} & \mu_1 & \epsilon_{T2} & \mu_2 \\ DIS_{mode}^{balken1} & w_1 & w_2 & w_3 \\ REF_{mode}^{balken1} & w_1 & w_2 & w_3 \\ DIS_{mode}^{balken2} & w_1 & w_2 & w_3 \\ REF_{mode}^{balken2} & w_1 & w_2 & w_3 \end{array}$$

Die Bedeutung der Parameter entspricht *elmt90* bzw. *elmt83*.

Die Einträge in der 3. und 4. Zeile stehen für den ersten Balken, d.h. Knoten 1 bis 3.

Die Einträge in der 5. und 6. Zeile stehen für den zweiten Balken, d.h. Knoten 4 bis 6.

Die Wahl des rationalen Ansatzes, $DIS_{mode}^{balken1} = REF_{mode}^{balken1} = 2$, steht hier nicht zur Verfügung.

elmt92

8 Knoten, Materialparameter $d(28)$

$$\begin{array}{cccc} \epsilon_N & r_1 & r_2 & \\ \epsilon_{T1} & \mu_1 & \epsilon_{T2} & \mu_2 \\ DIS_{mode}^{balken1} & w_1 & w_2 & w_3 & w_4 \\ REF_{mode}^{balken1} & w_1 & w_2 & w_3 & w_4 \\ DIS_{mode}^{balken2} & w_1 & w_2 & w_3 & w_4 \\ REF_{mode}^{balken2} & w_1 & w_2 & w_3 & w_4 \end{array}$$

Die Bedeutung der Parameter entspricht *elmt90* bzw. *elmt84*.

Die Einträge in der 3. und 4. Zeile stehen für den ersten Balken, d.h. Knoten 1 bis 4.

Die Einträge in der 5. und 6. Zeile stehen für den zweiten Balken, d.h. Knoten 5 bis 8.

Die Wahl des rationalen Ansatzes, $DIS_{mode}^{balken1} = REF_{mode}^{balken1} = 2$, steht hier nicht zur Verfügung.

Symbolverzeichnis

$(\cdot)'$	Ableitung nach s_0 , 20	Testfunktion, 27
$\widehat{(\cdot)}$	schiefsymmetrische Matrix, 3	Variation bzw. Ableitung nach der Konfiguration $\bar{\Phi}$, 24
$\mathbf{0}$	Nullvektor und Nullmatrix, 3	$\Delta\delta(\cdot)$
A	Ansatzfunktionen, 29	zweite kovariante Ableitung, 8
	Querschnittsfläche, 21	zweite Variation bzw. Ableitung nach der Konfiguration $\bar{\Phi}$, 25
\mathbf{A}	Matrix der Ansatzfunktionen, 29	ds
\mathbf{a}	lokales Koordinatensystem in der aktuellen Konfiguration, 18	differentielles Balkenelement in der aktuellen Konfiguration, 19
A_{ref}	Ansatzfunktionen der Referenzgeometrie, 29	ds_0
A_s	Schubquerschnittsfläche, 21	differentielles Balkenelement in der Referenzkonfiguration, 19
β	Binormale zu \boldsymbol{q} , 55	E
B	Bernstein-Polynom, 12	Elastizitätsmodul, 21
C	Bézier-Kurve, 13	$(\cdot)^e$
	C^1 -Spline, 15	Knotenwerte, 29
	Materialmatrix in lokalen Koordinaten, 22	\mathbf{e}
D	Materialmatrix in globalen Koordinaten, 35	globales Koordinatensystem, 17
d	Balkenabstand, 55	$\tilde{\mathbf{e}}$
$\Delta(\cdot)$	Inkrement der Newton-Iteration, 39	Einheitsvektor am Punkt des minimalen Abstandes, 55
	Richtung der kovarianten Ableitung, 8	$\boldsymbol{\varepsilon}$
$\delta(\cdot)$	Gâteaux-Variation, 6	Normal- und Schubverzerrung, 21
	kovariante Ableitung, 7	ϵ_N
	Richtung der kovarianten Ableitung, 7	Strafparameter in Normalenrichtung, 57
		F
		Einzelkraft, 22
		G
		Menge der Gauß'schen Integrationspunkte, 35
		Schubmodul, 21
		\mathbf{g}
		lokales Koordinatensystem in der Referenzkonfiguration, 17
		$\tilde{\mathbf{g}}$
		Ableitung von $\tilde{\mathbf{e}}$, 55
		Γ
		Abbildung auf Referenzelement, 28
		gip

- Gauß'scher Integrationspunkt, 28
- \mathbb{H}
Raum der Quaternionen, 9
- I
Flächenträgheitsmoment, 21
- $(\cdot)^{(i)}$
Index der Ansatzfunktionen, 29
- $\text{Im } \mathfrak{q}$
Vektorteil einer Quaternion, 9
- I_n
Einheitsmatrix, 3
- J
Jacobimatrix, 30
- $(\cdot)_j$
Index des Balkens bei Kontakt, 54
- $(\cdot)_k$
Iterationsschritt der Newton-Iteration, 39
- κ
Torsions- und Biegeverzerrung, 21
- κ
Krümmung, 55
- L
Balkenlänge in der aktuellen Konfiguration, 17
- L_0
Balkenlänge in der Referenzkonfiguration, 17
- Λ
Rotationsmatrix, 4
Verdrehung, 18
- $\overline{\Lambda}$
Verdrehung in der aktuellen Konfiguration, 17
- Λ_0
Verdrehung in der Referenzkonfiguration, 17
- L_e
Finites Element, 28
- M
Einzelmoment, 22
Mannigfaltigkeit, 6
- M
Ableitung von C , 15
- m
Schnittmoment, 19
- N
Kontaktkraft in Normalenrichtung, 57
- n
Schnittkraft, 19
- \mathbb{N}^+
Menge der natürlichen Zahlen ohne Null, 3
- \mathbb{N}_0
Menge der natürlichen Zahlen mit Null, 3
- nen
Zahl der Knoten je Element, 29
- ν
Normale zu ϱ , 55
- ν
Querkontraktion, 21
- ω
Torsions- und Biegeverzerrung κ , 21
- p
Streckenlast, 19
- p
Durchdringung, 57
- $\partial(\cdot)$
Ableitung nach ξ , 29
- Φ
Deformation, 19
- $\overline{\Phi}$
aktuelle Konfiguration, 17
- Φ_0
Referenzkonfiguration, 17
- Π
Potential, 24
- q
Streckenmoment, 19
- \bar{q}
konjugierte Quaternion, 9
- q
Quaternion, 9
- u_Λ
Quaternion zur Rotation Λ , 11
- \mathbb{R}
Menge der reellen Zahlen, 3
- r
allgemeine Kontaktkraft, 58
- r
Querschnittsradius, 57
- R_0
Bogenradius in der Referenzkonfiguration, 69

- $\text{Re } \mathfrak{q}$
 Skalarmeil einer Quaternion, 9
- RES***
 Residuum, 37
- RES_{kont}***
 Kontaktanteil des Residuums, 61
- ϱ
 Balkenschwerachse bei Kontakt, 54
- s_0
 Bogenlängenkoordinate in der Referenz-
 konfiguration, 17
- SO_3
 Menge der Rotationen, 5
- s^c
 Bogenlängenkoordinate der Kontaktstelle,
 58
- so_3
 Menge der schiefssymmetrischen Matrizen,
 5
- TANG***
 tangentielle Steifigkeitsmatrix, 36
- TANG_{geo}***
 geometrischer Teil der Steifigkeitsmatrix,
 36
- TANG_{kont}***
 Kontaktanteil der Steifigkeitsmatrix, 61
- TANG_{mat}***
 materieller Teil der Steifigkeitsmatrix, 36
- τ
 Tangentialvektor zu ϱ , 55
- τ
 Torsion, 57
- θ
 Norm des Rotationsvektors, 5
 Rotationswinkel, 4
- $T_x M$
 Tangentialraum einer Mannigfaltigkeit, 6
- $T_\Lambda SO_3$
 Tangentialraum zu SO_3 , 5
- u
 Verschiebung, 18
- \bar{u}
 Balkenschwerachse in der aktuellen
 Konfiguration, 17
- u_0
 Balkenschwerachse in der Referenz-
 konfiguration, 17
- w
 infinitesimale Rotation, 4
 Rotationsachse, 4
 Rotationsvektor, 4
- \tilde{w}
 Summe der inkrementellen Verdrehungen,
 31
- ξ
 Koordinate im Referenzelement, 28
- ξ^c
 Koordinate der Kontaktstelle, 60
- \mathbb{Z}
 Menge der ganzen Zahlen, 3

Literaturverzeichnis

- [Arg82] ARGYRIS, John: An excursion into large rotations. In: *Computer methods in applied mechanics and engineering* 32 (1982), S. 85–155
- [BB79] BATHE, Klaus-Jürgen ; BOLOURCHI, Said: Large displacement analysis of three-dimensional beam structures. In: *International journal for numerical methods in engineering* 14 (1979), S. 961–986
- [Bet04] BETTEN, Josef: *Finite Elemente für Ingenieure 2*. Springer, 2004
- [BLL06] BRAUNE, Klaus ; LAMMARSCH, Joachim ; LAMMARSCH, Marion: *Latex*. Springer, 2006
- [Bra03] BRAESS, Dietrich: *Finite Elemente*. Springer, 2003
- [DB02] DEUFLHARD, Peter ; BORNEMANN, Folkmar: *Numerische Mathematik I*. de Gruyter, 2002
- [Flegt] FLEISCHER, Michael: *Verifikation der virtuellen Prozesskette und Umformverhalten*, TU München, Fakultät für Bauingenieur- und Vermessungswesen, Lehrstuhl für Statik, Diss., (noch nicht verteidigt)
- [GR08] GHOSH, S. ; ROY, D.: Consistent quaternion interpolation for objective finite element approximation of geometrically exact beam. In: *Computer methods in applied mechanics and engineering* 198 (2008), S. 555–571
- [Ibr95] IBRAHIMBEGOVIC, Adnan: On finite element implementation of geometrically nonlinear Reissner's beam theory: three-dimensional curved beam elements. In: *Computer methods in applied mechanics and engineering* 122 (1995), S. 11–26
- [Ibr97] IBRAHIMBEGOVIC, Adnan: On the choice of finite rotation parameters. In: *Computer methods in applied mechanics and engineering* 149 (1997), S. 49–71
- [IF94] IBRAHIMBEGOVIC, Adnan ; FREY, François: Stress resultant geometrically nonlinear shell theory with drilling rotations - Part II. Computational aspects. In: *Computer methods in applied mechanics and engineering* 118 (1994), S. 285 – 308
- [IFK95] IBRAHIMBEGOVIC, Adnan ; FREY, François ; KOZAR, Ivica: Computational aspects of vector-like parametrization of three-dimensional finite rotations. In: *International journal for numerical methods in engineering* 38 (1995), S. 3653–3673
- [IT02] IBRAHIMBEGOVIC, Adnan ; TAYLOR, Robert L.: On the role of frame-invariance in structural mechanics models at finite rotations. In: *Computer methods in applied mechanics and engineering* 191 (2002), S. 5159–5176
- [Knö96] KNÖRRER, Horst: *Geometrie*. Vieweg, 1996

- [KS05] KONYUKHOV, A. ; SCHWEIZERHOF, K.: Covariant description for frictional contact problems. In: *Computational Mechanics* 35 (2005), S. 190–213
- [KSht] KONYUKHOV, Alexander ; SCHWEIZERHOF, Karl: Geometrical covariant approach for contact between curves representing beam and cable type structures. (noch nicht veröffentlicht)
- [Kui99] KUIPERS, Jack B.: *Quaternions and rotation sequences*. Princeton University Press, 1999
- [MV03] MEYBERG, Kurt ; VACHENAUER, Peter: *Höhere Mathematik 2*. Springer, 2003
- [PT97] PIEGL, Les A. ; TILLER, Wayne: *The NURBS book*. Springer, 1997
- [Rei72] REISSNER, E.: On One-Dimensional Finite-Strain Beam Theory: The Plane Problem. In: *Journal of Applied Mathematics and Physics* 23 (1972), S. 795–804
- [Rei81] REISSNER, E.: On finite deformations of space-curved beams. In: *Journal of Applied Mathematics and Physics* 32 (1981), S. 734–744
- [RT00] REHM, H.P. ; TRINKS, W.: *Lineare Algebra und analytische Geometrie*. Universität Karlsruhe, Fakultät für Mathematik, 2000
- [Sch92] SCHMITT, Günther: *Fortran-Kurs technisch orientiert*. Oldenbourg, 1992
- [Sch95] SCHOTTENLOHER, Martin: *Geometrie und Symmetrie in der Physik*. Vieweg, 1995
- [SGH02] SCHNELL, Walter ; GROSS, Dietmar ; HAUGER, Werner: *Technische Mechanik 2*. Springer, 2002
- [Sim85] SIMO, J.C.: A finite strain beam formulation. The three-dimensional dynamic problem. Part I. In: *Computer methods in applied mechanics and engineering* 49 (1985), S. 55–70
- [Sim86] SIMO, L. J.C. Vu-Quoc: A three-dimensional finite-strain rod model. Part II: Computational aspects. In: *Computer methods in applied mechanics and engineering* 58 (1986), S. 79–116
- [Sim92] SIMO, J.C.: The (symmetric) Hessian for geometrically nonlinear models in solid mechanics: Intrinsic definition and geometric interpretation. In: *Computer methods in applied mechanics and engineering* 96 (1992), S. 189–200
- [SSD03] SOLIN, Pavel ; SEGETH, Karel ; DOLEZEL, Ivo: *Higher-Order Finite Element Methods*. Chapman & Hall/CRC, 2003
- [Tay08] TAYLOR, Robert L.: *FEAP - A finite element analysis program*. Version 8.2. <http://www.ce.berkeley.edu/projects/feap/manual.pdf>, March 2008
- [Wag07] WAGNER ; INSTITUT FÜR BAUSTATIK, UNIVERSITÄT KARLSRUHE (Hrsg.): *FEAP, A Finite Element Analysis Programm*. Version 02/2007. www.bs.uni-karlsruhe.de/download/feap-man.pdf: Institut für Baustatik, Universität Karlsruhe, 2007
- [Wri06] WRIGGERS, Peter: *Computational Contact Mechanics*. Springer, 2006