

# **Sensitivitätsanalyse stabiler Gleichgewichtslagen dünnwandiger Strukturen unter Verwendung von Lösungsverfahren für Parallelrechner**

Zur Erlangung des akademischen Grades eines

**DOKTOR-INGENIEURS**

von der Fakultät für

Bauingenieur- und Vermessungswesen  
der Universität Fridericiana zu Karlsruhe (TH)

genehmigte

**DISSERTATION**

von

Dipl.-Ing. Thomas Rottner  
aus Karlsruhe

Tag der mündlichen Prüfung : 3. Mai 2000  
Hauptreferent : Prof. Dr.-Ing. K. Schweizerhof  
Korreferent : Prof. Dr. G. Alefeld  
Korreferent : Prof. Dr.-Ing. P. Vielsack

Karlsruhe 2000



## Kurzfassung

Um die Stabilität von Gleichgewichtslagen dünnwandiger, und damit beulgefährdeter Strukturen quantitativ zu beurteilen, wird in der vorliegenden Arbeit der Begriff der Sensitivität eingeführt. Sie wird definiert als der reziproke Wert der kinetischen Energie, die einem System mindestens eingeprägt werden muß, um ausgehend von einer stabilen Gleichgewichtslage eine weitere stabile Gleichgewichtslage oder unbegrenzt anwachsende Verschiebungen zu erreichen. Für konkrete numerische Untersuchungen dient die Methode der Finiten Elemente als Simulationsinstrument.

Die Untersuchung von Stabilitätsproblemen unter Einsatz der Finiten Elemente Methode erfolgt meist mittels der statischen Verfolgung des Last-Verformungspfades. Konvergenzprobleme des zur Lösung der nichtlinearen Probleme üblicherweise genutzten Newton Verfahrens und die zum Teil notwendige manuelle Steuerung des Bogenlängenverfahrens gestalten die Lösung insbesondere für Systeme mit vielen Freiheitsgraden sehr aufwendig. Eventuell vorhandene verzweigende Lösungsäste müssen ebenfalls berechnet werden. Dennoch ist nicht gewährleistet, daß eine minimale Traglast des untersuchten Systems im Nachbeulbereich gefunden wird, wie am Beispiel eines axial belasteten Stahlzylinders belegt wird. Transiente Berechnungen stellen hier eine attraktive Alternative dar. Hiermit ist die Berechnung des Beulverhaltens bis in den Nachbeulbereich mit moderatem Aufwand möglich. Die Bestimmung der Sensitivität der Gleichgewichtslagen im Vorbeulbereich bestätigt die durch die transiente Simulation erhaltene Nachbeullast der Struktur als zur Bemessung wesentliche Traglast.

Die Vielzahl der zur Sensitivitätsanalyse notwendigen nichtlinearen Finite Element Simulationen erfordert den Einsatz optimierter Algorithmen und moderner Rechnerarchitekturen, um die Rechenzeit zu minimieren. Dieses gilt insbesondere für den Algorithmus zum Auflösen der entstehenden linearen Gleichungssysteme. Oft werden hierzu direkte Lösungsverfahren eingesetzt, da sie sich insbesondere bei Versagensproblemen als robust erwiesen haben. Numerische Vergleiche von effizienten direkten Lösern mit iterativen Krylov Unterraumverfahren haben jedoch ergeben, daß auch iterative Löser für solche schlecht konditionierten Probleme robust und effizient einsetzbar sind. Zur Konvergenzbeschleunigung werden dabei algebraische, also auf der Koeffizientenmatrix basierende Vorkonditionierer eingesetzt. Der Einsatz dieser Löser ist somit für alle Problemklassen bzw. Elementtypen gewährleistet.

Die Robustheit und Effizienz der iterativen Lösungsverfahren erlaubt die Implementierung von Finite Element Programmen auf Parallelrechnern unter ausschließlicher Verwendung dieser Löser. Iterative Gleichungslöser sind für parallele Rechnerarchitekturen relativ einfach zu implementieren und zeichnen sich durch eine gute Skalierbarkeit aus. Anhand von Beispielen mit linearem und nichtlinearem Verhalten wird dies belegt.

Die durchgeführte Parallelisierung des Finiten Elemente Programmes basiert auf einem geometrischen Ansatz, d.h. das Finite Elemente Netz wird in möglichst gleich große Gebiete zerlegt und auf die einzelnen Prozessoren verteilt. Die Gebietszerlegung ist statisch, d.h. es findet während einer nichtlinearen Berechnung keine Umverteilung statt. Letzte-

res ist erst bei hier nicht berücksichtigten adaptiven Methoden erforderlich, da dann die Rechenlast auf den verschiedenen Prozessoren sehr ungleichmäßig werden kann. Großen Wert wurde auf die breite Anwendbarkeit des parallelisierten Programmes gelegt. So werden keinerlei Einschränkungen bezüglich der Elementtypen gemacht und sowohl statische als auch dynamische nichtlineare Lösungsverfahren implementiert. Der Übertragbarkeit des parallelisierten Programmes auf unterschiedliche Rechnerarchitekturen ist durch den Einsatz der standardisierten Kommunikationsbibliothek MPI Rechnung getragen.

## Abstract

To quantitatively judge the stability of equilibrium states of thin-walled structures, which are prone to buckle, the term of sensitivity is introduced in the present thesis. It is defined as the reciprocal value of the kinetic energy that is necessary to be introduced into a mechanical system in order to reach either a second stable equilibrium state or to achieve unlimited growth of displacements. For numerical investigations, the method of finite elements is used.

The investigation of stability problems using the finite element method usually is tackled by computing the static load-deflection behavior. However, convergence problems of Newton's method and the partially necessary manual control of the arc-length method make this solution strategy unfavorable, especially for systems involving many unknowns. Possible branching solution paths have to be calculated as well. Nevertheless, it is not ensured, that the minimal post-buckling load of the investigated structure is found, as is shown for the special structure of an axially loaded steel cylinder. Here, transient computations are an attractive alternative. Then, the calculation of the buckling behavior is possible with only moderate effort. The estimation of the sensitivity of equilibrium states in the pre-buckling regime confirms the post-buckling load of the transient simulation as an important design load.

The large number of nonlinear finite element simulations that is necessary to calculate the sensitivity requires the use of efficient algorithms and modern hardware architectures to minimize the computing time. This is especially true for the algorithm to solve the arising systems of linear equations. Here, often direct solvers are used, as they are known to be robust for collapse problems. However, comparisons with iterative Krylov subspace methods have shown, that iterative methods also are robust and efficient for badly conditioned systems from structural mechanics. To improve the convergence behavior, algebraic preconditioners based on the coefficient matrix are used. Therefore, the use of these methods is possible for all types of problems, regardless, what kind of element is used.

The robustness and efficiency of iterative solution methods allows the parallel implementation of finite element programs with the restriction to these solution methods. Iterative solvers are relatively easy to implement for parallel strategies and scale very well. This is shown with some linear and nonlinear examples.

The parallelization of the finite element code performed is based on a geometrical ap-

proach, e.g. the finite element mesh is partitioned in preferably equally sized parts and distributed to the different processors. The partitioning is static, no redistribution is performed within a nonlinear computation. This is sufficient, until no adaptive strategies are used, as the load can then be quite unbalanced. Much care has been taken to ensure a wide application range of the parallel program. Therefore, no restrictions to the type of elements are made, and solution procedures for static and transient nonlinear problems are available. The portability of the program is ensured by the use of the standard communication library MPI.

## Vorwort

Die vorliegende Arbeit entstand während meiner Tätigkeit als wissenschaftlicher Angestellter am Institut für Mechanik der Universität Karlsruhe. Wesentliche Teile wurden im Rahmen des vom BMBF geförderten Verbundprojektes VERSA “Verbesserung und Beschleunigung von Versagensanalysen in der Strukturmechanik” erarbeitet.

Herrn Prof. Dr.-Ing. K. Schweizerhof danke ich für die Anregung zu dieser Arbeit und die Übernahme des Hauptreferates. Seine engagierte und kompetente wissenschaftliche Betreuung hat diese Arbeit mit geprägt.

Für die sorgfältige Durchsicht der Arbeit, die Übernahme des Korreferates und die begleitende Betreuung bei dem Verbundprojekt VERSA bedanke ich mich bei Herrn Prof. Dr. G. Alefeld.

In besonderem Maße möchte ich mich bei Herrn Prof. Dr.-Ing. P. Vielsack für seine – über die Mechanik hinausreichenden – Ratschläge und Hinweise bedanken. Durch seine stete Diskussionsbereitschaft und konstruktive Kritik wird der wissenschaftliche Dialog am Institut immens belebt.

Schließlich danke ich allen Kollegen des Instituts für das konstruktive Arbeitsklima und für zahllose offene Diskussionen. Besonders hervorheben möchte ich hierbei die Herren Ralf Hauptmann und Stefan Doll, deren stete Hilfsbereitschaft und Unterstützung in allen Situationen für mich eine große Hilfe war. Gleichermäßen danke ich Frau Ingrid Lenhardt vom Institut für Angewandte Mathematik für die vertrauensvolle und angenehme, auch über das Projekt VERSA hinausreichende Zusammenarbeit.

Karlsruhe, Mai 2000

Thomas Rottner

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>Zur numerischen Lösung strukturmechanischer Probleme</b>	<b>4</b>
2.1	Grundgleichungen der Kontinuumsmechanik . . . . .	4
2.1.1	Kinematik und Verzerrung . . . . .	4
2.1.2	Bilanzgleichungen . . . . .	6
2.1.3	Spannungsmaße und Stoffgesetz . . . . .	8
2.1.4	Das Prinzip der virtuellen Verschiebungen . . . . .	9
2.1.5	Linearisierung des Prinzips der virtuellen Verschiebungen . . . . .	10
2.2	Finite Element Diskretisierung . . . . .	11
2.2.1	Das isoparametrische Konzept . . . . .	12
2.2.2	Die diskrete schwache Form . . . . .	12
2.3	Lösungsverfahren . . . . .	13
2.3.1	Das Newton-Raphson Verfahren . . . . .	13
2.3.2	Das Bogenlängenverfahren . . . . .	15
2.3.3	Das Newmark Verfahren . . . . .	17
<b>3</b>	<b>Wahl von Benchmark Problemen</b>	<b>20</b>
3.1	Zahnkrone . . . . .	20
3.2	Flache Zylinderschale . . . . .	20
3.3	Dünnwandiger Torus . . . . .	23
3.4	Rohrkreuz . . . . .	25
3.5	Gummiblock . . . . .	27
<b>4</b>	<b>Direkte und iterative Verfahren</b>	<b>30</b>
4.1	Direkte Gleichungslösung . . . . .	31
4.1.1	Einige Begriffe aus der Graphentheorie . . . . .	33
4.1.2	Permutationsstrategien und symbolische Faktorisierung . . . . .	33
4.1.3	Numerische Faktorisierung . . . . .	35

4.1.4	Rückwärts- und Vorwärtselimination . . . . .	35
4.2	Iterative Gleichungslösung . . . . .	36
4.2.1	Das Verfahren der konjugierten Gradienten . . . . .	37
4.2.2	Vorkonditionierung . . . . .	40
4.2.2.1	Jacobi-Vorkonditionierung . . . . .	42
4.2.2.2	SSOR-Vorkonditionierung . . . . .	42
4.2.2.3	Unvollständige $LDL^T$ -Zerlegungen . . . . .	43
4.2.2.4	Unvollständige $LDL^T$ -Zerlegung auf dem Speicherplatz von $A$ (MPILU) . . . . .	43
4.2.2.5	Blockweise unvollständige $LDL^T$ -Zerlegung . . . . .	44
4.2.2.6	Unvollständige $LDL^T$ -Zerlegung mit <i>fill-in</i> erster Stufe (FLILU) . . . . .	45
4.2.2.7	Unvollständige $LDL^T$ -Zerlegung mit <i>numerical drop- ping</i> (NDILU) . . . . .	46
4.2.2.8	Element-by-Element (EBE) Vorkonditionierung . . . . .	46
4.2.3	Das Lanczos Verfahren . . . . .	47
4.2.4	Das QMR-Verfahren . . . . .	52
4.2.5	Abbruchkriterium . . . . .	53
4.3	Vergleichsberechnungen für lineare Probleme . . . . .	54
4.3.1	Numerischer Vergleich direkter Lösungsstrategien . . . . .	55
4.3.2	Permutationsstrategien zur Verbesserung der Qualität von unvoll- ständigen Faktorisierungen . . . . .	58
4.3.3	CG-Verfahren mit blockweiser unvollständiger $LDL^T$ -Zerlegung	60
4.3.4	Vergleich zwischen direkten und iterativen Verfahren . . . . .	61
4.4	Vergleichsuntersuchungen nichtlinearer Probleme . . . . .	64
4.4.1	Vorkonditionierungsstrategien bei nichtlinearen Berechnungen . .	65
4.4.2	Abbruchkriterium bei nichtlinearen Berechnungen . . . . .	66
4.4.3	Vergleich zwischen iterativen und direkten Verfahren . . . . .	67
4.5	Richtlinien zur Wahl eines Löser . . . . .	69

---

5.1	Klassifikation von Parallelrechnern . . . . .	71
5.2	Programmiermodelle . . . . .	73
5.3	<i>Message Passing</i> . . . . .	73
5.4	Netzwerktopologien . . . . .	74
5.4.1	Vollständiges Netzwerk . . . . .	74
5.4.2	Ringtopologie . . . . .	75
5.4.3	Gitter- bzw. Torustopologie . . . . .	75
5.4.4	Mehrstufige <i>Crossbar Switches</i> . . . . .	76
5.5	<i>Speedup</i> , Effizienz, Amdahl's Gesetz . . . . .	77
5.6	Parallelisierung eines Finite Element Programmes . . . . .	79
5.6.1	Gebietszerlegungsverfahren . . . . .	80
5.6.2	Vom FE-Netz zum Graphen . . . . .	81
5.6.3	Geometrisch basierte Heuristiken zur Partitionierung . . . . .	82
5.6.4	Graphenorientierte Heuristiken zur Partitionierung . . . . .	83
5.7	Datenverteilung . . . . .	87
5.7.1	Vektoren mit lokalen und globalen Einträgen . . . . .	87
5.7.2	Last- bzw. Verschiebungsvorgabe . . . . .	88
5.8	Iterative Gleichungslösung . . . . .	88
5.8.1	Vektor-Aufdatierungen . . . . .	89
5.8.2	Skalarprodukte . . . . .	89
5.8.3	Matrix-Vektor Multiplikation . . . . .	90
5.8.4	Lösen des Vorkonditionierungssystems . . . . .	90
5.9	Weitere Aspekte der Parallelisierung . . . . .	91
5.9.1	Details zur Globalisierung der lokalen Steifigkeitsmatrizen . . . . .	91
5.9.2	Statische nichtlineare Finite Element Berechnungen . . . . .	93
5.10	Parallele Vergleichsberechnungen . . . . .	94
5.10.1	Lineares Problem: Zahnkrone . . . . .	94
5.10.2	Nichtlineares Problem: Rohrkreuz . . . . .	95
5.11	Wertung der Resultate . . . . .	96

<b>6</b>	<b>Einige numerische Aspekte</b>	<b>98</b>
6.1	Über die Kondensation von EAS-Parametern . . . . .	98
6.1.1	Kondensation innerer EAS-Parameter . . . . .	99
6.1.1.1	Algorithmus EAS-1 . . . . .	100
6.1.1.2	Algorithmus EAS-2 . . . . .	100
6.1.2	Vergleich der Algorithmen EAS-1 und EAS-2 . . . . .	100
6.1.2.1	Konvergenzverhalten . . . . .	103
6.1.2.2	Vergleich mit dem ANS3Dq Element . . . . .	104
6.1.3	Zusammenfassung und Diskussion . . . . .	104
6.2	Verteilte Berechnung des Skalarproduktes . . . . .	106
<b>7</b>	<b>Stabilitätsuntersuchungen mit Finiten Elementen</b>	<b>107</b>
7.1	Stabilitätsbegriff nach Ljapunov . . . . .	107
7.1.1	1. Methode nach Ljapunov . . . . .	108
7.1.2	2. oder direkte Methode nach Ljapunov . . . . .	108
7.1.3	Sonderfall: Gleichgewichtslagen . . . . .	109
7.2	Behandlung von Stabilitätsproblemen . . . . .	109
7.2.1	Trägheit der tangentiellen Steifigkeitsmatrix . . . . .	110
7.2.2	Determinantenkriterium . . . . .	111
7.2.3	Steifigkeitswert nach Bergan . . . . .	112
7.2.4	Eigenwertanalyse . . . . .	112
7.3	Prognose von singulären Punkten . . . . .	113
7.3.1	Bisektionsverfahren . . . . .	113
7.3.2	Lineare Eigenwertuntersuchungen . . . . .	114
7.3.3	Direkte Berechnung von Stabilitätspunkten . . . . .	115
7.3.4	Klassifizierung von singulären Punkten . . . . .	116
7.3.5	Behandlung von Verzweigungsproblemen . . . . .	117
7.3.6	Imperfektionsempfindlichkeit . . . . .	120
7.3.7	Zwischenbilanz zu statischen Stabilitätsanalysen . . . . .	121
7.3.8	Perfekte und imperfekte Systeme . . . . .	122

---

7.4	Transiente Analysen . . . . .	123
7.4.1	Gleichung der ersten Näherung . . . . .	124
7.4.2	Dynamische Stabilitätsuntersuchungen . . . . .	126
7.5	Weitere Beurteilung von Gleichgewichtslagen . . . . .	127
7.6	Auffinden nichteindeutiger Gleichgewichtslagen . . . . .	129
7.6.1	Symmetrisch stabiles Systemverhalten . . . . .	130
7.6.2	Symmetrisch instabiles Systemverhalten . . . . .	137
7.6.3	Durchschlagproblem . . . . .	142
7.7	Zusammenfassung und Diskussion . . . . .	145
<b>8</b>	<b>Untersuchung eines Stahlzylinders mit imperfekter Geometrie</b>	<b>147</b>
8.1	Problembeschreibung . . . . .	147
8.2	Modellierung mit Finiten Elementen . . . . .	148
8.3	Beschreibung der Versuchsergebnisse . . . . .	149
8.4	Imperfektionsempfindlichkeit . . . . .	150
8.5	Statische Stabilitätsuntersuchung . . . . .	152
8.6	Transiente Belastung . . . . .	154
8.6.1	Verschiebungsgesteuerter Prozess . . . . .	155
8.6.2	Gewichtskraftgesteuerter Prozess . . . . .	157
8.6.3	Vergleich beider Prozesse . . . . .	158
8.6.4	Rechentechnische Aspekte . . . . .	159
8.7	Sensitivität von Gleichgewichtslagen im Vorbeulbereich . . . . .	161
8.7.1	Definition der Anfangsgeschwindigkeitsverteilungen . . . . .	162
8.7.2	Gleichgewichtslage bei $120\text{ kN}$ . . . . .	163
8.7.3	Gleichgewichtslage bei $80\text{ kN}$ . . . . .	169
8.7.4	Gleichgewichtslage bei $50\text{ kN}$ . . . . .	171
8.7.5	Sensitivität . . . . .	174
8.8	Untersuchungen an symmetrischen Zylinderausschnitten . . . . .	175
8.8.1	Untersuchungen an Zylinderhälften . . . . .	175
8.8.2	Untersuchungen an einem Zylinderviertel . . . . .	178

8.8.3	Vergleich der Sensitivität . . . . .	178
8.8.4	Diskussion der Modellierung mit Zylinderausschnitten . . . . .	181
8.9	Zusammenfassung und Wertung der Resultate . . . . .	182
<b>9</b>	<b>Zusammenfassung und Ausblick</b>	<b>184</b>
<b>A</b>	<b>Mathematische Grundlagen</b>	<b>199</b>
A.1	Rechenregeln zur Divergenz . . . . .	199
A.2	Gaußscher Integralsatz . . . . .	199
<b>B</b>	<b>Finite Element Formulierung eines Scheibenelementes</b>	<b>199</b>
<b>C</b>	<b>Speicherstruktur</b>	<b>201</b>

# 1 Einleitung

Zur Verkürzung des Produktzyklus werden bei der Entwicklung technischer Produkte aus Kostengründen verstärkt numerische Simulationen in den Entwicklungsprozeß integriert. Für Großserienprodukte, wie z.B. Fahrzeuge oder Fahrzeugteile, erlaubt die Simulation eine Reduktion der Anzahl von benötigten Prototypen. Bei Einzelkonstruktionen wie z.B. Bauwerken kann eine größere Sicherheit durch die vorherige numerische Untersuchung unterschiedlicher Konstruktionsvarianten erreicht werden. Zur Strukturanalyse wird als Simulationsinstrument oft die Methode der Finiten Elemente eingesetzt. Analytische Methoden liefern meist nur für Probleme mit einfachen Geometrien und Randbedingungen Lösungen.

Für Festigkeitsberechnungen genügen häufig lineare Analysen, um Spannungsverteilungen und Verformungen von Bauteilen zu berechnen. Soll aber das gesamte Strukturverhalten unter Berücksichtigung von geometrischen und materiellen Nichtlinearitäten ermittelt werden, müssen wegen des nichtlinearen Zusammenhanges zwischen einem Steuerparameter (meist der Last) und den Variablen (z.B. Verschiebungen) sogenannte nichtlineare Analysen durchgeführt werden, wie beispielsweise bei Versagens- und Stabilitätsproblemen. Für eine ingenieurpraktische Anwendung liegt das Berechnungsziel meist im Last-Verformungsverhalten der mechanischen Struktur, wobei insbesondere Informationen über kritische Lasten, plötzliche Änderungen im Systemverhalten usw. von Interesse sind. Zusätzlich sind bei Stabilitätsproblemen Aussagen über die Qualität der berechneten Gleichgewichtslagen von Bedeutung. Ist eine Gleichgewichtslage stabil, so ist sie physikalisch realisierbar, andernfalls nicht. Für technische Zwecke sind daher nur stabile Gleichgewichtslagen geeignet. Mit dem Stabilitätsbegriff ist eine Aussage über die Qualität der Gleichgewichtslage aber nur eingeschränkt möglich. So könnte die Gleichgewichtslage zwar stabil gegenüber beliebig kleinen Störungen sein, jedoch empfindlich bzw. sensitiv gegenüber endlichen aber dennoch relativ kleinen Störungen. Ein solches Verhalten wird als praktisch instabil bezeichnet. Um die Sicherheit von stabilen Gleichgewichtslagen gegenüber solcher endlichen Störungen einzuschätzen, wird der Begriff der Sensitivität verwendet.

Zur Bestimmung der Sensitivität stabiler Gleichgewichtslagen sind im Rahmen der Methode der Finiten Elemente für komplexe realitätsnahe Probleme bislang noch keine gesicherten Methoden verfügbar. Hierzu soll im Rahmen dieser Arbeit ein Beitrag geleistet werden. Wesentlicher Bestandteil ist das Testen der vorgeschlagenen Vorgehensweisen an realen Problemen. Dabei entsteht die Notwendigkeit zahlreicher nichtlinearer kinetischer Simulationen mit hoher Auflösung der Geometrie, um die wesentlichen Eigenschaften des realen Systems möglichst gut abzubilden.

Um solche Simulationen überhaupt mit überschaubarem Aufwand durchführen zu können, ist es wichtig, effiziente Algorithmen zur Beschreibung des mechanischen Modells bereitzustellen – also die das Stoffgesetz beinhaltenden Elemente. Eine große Bedeutung hinsichtlich Effizienz kommt aber dem Algorithmus zum Auflösen der entstehenden linearen Gleichungssysteme zu. Dies ist in der Mathematik ein Gebiet langjähriger, bis heute aktueller Forschung, deren Ergebnisse bislang aber noch nicht ausreichend Eingang in die In-

genieurwissenschaften fanden, da die Computerindustrie durch leistungsstärkere Prozessoren und preiswerte Hauptspeichermodule die steigenden Anforderungen bezüglich Größe der Gleichungssysteme und benötigte Zeit zur Lösung anwenderseits oftmals befriedigen konnte.

Inzwischen scheint sich aber die Entwicklung in der Prozessortechnologie zu verlangsamen. Eine deutliche Erhöhung der Rechenleistung ist daher vorzugsweise durch eine veränderte Rechnerarchitektur zu erzielen. Insbesondere Parallelrechner sollen durch gleichzeitigen Einsatz mehrerer Prozessoren die Rechenleistung vervielfachen. Eine vollständige Überarbeitung und zum Teil eine Neuimplementierung der Simulationsprogramme ist dann meist unvermeidbar. Diese Anpassung der Programme an die parallele Rechnerarchitektur stellt die Programmentwickler vor eine große Herausforderung, da sich herkömmliche Algorithmen für die Parallelisierung teilweise als völlig ungeeignet erweisen, andere Prozeduren hingegen vorteilhaft zur parallelen Implementierung und Parallelbearbeitung sind. Die Eignung eines Algorithmus zur parallelen Bearbeitung liegt in einer gegenüber dem sequentiellen adäquat verkürzten Abarbeitung. Dies gilt insbesondere bei der bereits angesprochenen Frage der linearen Gleichungslösung: Fanden in Ingenieur Anwendungen bislang fast ausschließlich direkte, auf dem Gauß'schen Eliminationsverfahren basierende Algorithmen Verwendung, so werden nun insbesondere iterative Lösungsstrategien interessant, die sich zum einen sehr einfach parallelisieren lassen und außerdem eine höhere Effizienz auf parallelen Plattformen erreichen als direkte Verfahren. Der effiziente Einsatz dieser iterativen Verfahren ist insbesondere durch die bei Ingenieurproblemen häufig auftretenden nur sehr dünn besetzten Matrizen möglich. Bei den Programmarbeiten zur Parallelisierung stellt sich dann der Effekt ein, daß auch das ursprüngliche sequentielle Programm bezüglich Effizienz verbessert wird. Für den Parallelrechner entwickelte Algorithmen und Programmstrukturen erweisen sich häufig gleichermaßen für sequentielle Rechner geeignet.

Die Verfügbarkeit effizienter paralleler Algorithmen erlaubt die Lösung von Problemen, die vorher aufgrund rechenzeitintensiverer Algorithmen nur beschränkt lösbar waren.

## Gliederung der Arbeit

In Kapitel 2 werden diejenigen kontinuumsmechanischen Grundlagen bereitgestellt, die für die vorliegende Abhandlung benötigt werden. Die schwache Form des Gleichgewichts wird angegeben und die in der Methode der Finiten Elemente übliche Diskretisierung wird eingeführt. Für das entstehende nichtlineare algebraische Gleichungssystem werden Lösungsverfahren formuliert, die auf der Linearisierung dieses Gleichungssystems beruhen.

Da ein wesentlicher Teil dieser Arbeit sich mit der effizienten Lösung von Finiten Element Problemen befaßt, werden in Kapitel 3 einige Benchmark Probleme eingeführt, die zum Vergleich der entwickelten Lösungsverfahren genutzt werden. Diesen Problemen ist ein ausgeprägt nichtlineares Verhalten und überwiegend schlechte Konditionierung gemeinsam, sodaß zu erwarten ist, daß die damit erzielten Vergleichsergebnisse sich auf allgemeinere Aussagen für weite Problemklassen anwenden lassen.

Unterschiedliche Verfahren zur Lösung linearer Gleichungssysteme werden in Kapitel 4 erläutert. Zunächst werden effiziente direkte Verfahren diskutiert und anschließend auf iterative Verfahren eingegangen, die für die zu lösenden Problemklassen geeignet sind. Anhand der in Kapitel 3 eingeführten Probleme werden die beschriebenen Verfahren auf unterschiedlichen Rechnern verglichen, um für Anwender Hinweise zur Wahl eines effizienten Löser bereitzustellen.

Auf die Parallelisierung eines Finiten Elemente Programmes wird dann in Kapitel 5 eingegangen. Ausgehend von einer geometrischen Zerlegung des Diskretisierungsgebietes, z.B. des Ausgangsnetzes, wird die entstehende Datenverteilung und die nachfolgende Implementierung paralleler iterativer Lösungsverfahren beschrieben und Ergebnisse präsentiert.

Bei der Anwendung numerischer Verfahren treten gelegentlich auch Effekte auf, deren Ursache in numerischen Rundungsfehlern begründet ist. Dies gilt insbesondere für Algorithmen mit paralleler Ausführung der Operationen. Auf zwei solcher Probleme und deren Beseitigung wird in Kapitel 6 eingegangen.

Die Behandlung von Stabilitätsproblemen mit Hilfe der Finiten Element Methode ist Thema des Kapitels 7. Basierend auf dem Stabilitätsbegriff von Ljapunov werden die bekannten Kriterien angegeben und deren effiziente Ermittlung aufgrund der Verwendung von unterschiedlichen Verfahren zur linearen Gleichungslösung aus Kapitel 4 diskutiert. Ein Verfahren zum Auffinden nichteindeutiger Gleichgewichtslagen, die bei statischen Analysen nicht bestimmbar sind, wird vorgeschlagen. Damit einhergehend wird mittels auf das System aufgebrachtener endlicher Störungen die Qualität stabiler Gleichgewichtslagen bezüglich Sensitivität beurteilt.

Diese Verfahren werden im Kapitel 8 exemplarisch auf das Stabilitätsproblem eines axial belasteten, imperfekten Stahlzylinder angewandt und ausführlich diskutiert. Einflüsse der Modellierung, wie z.B. die Behandlung von symmetrischen Teilsystemen werden ebenfalls besprochen.

Kapitel 9 gibt eine Zusammenfassung und einen Ausblick auf weiterführende Entwicklungen.

## 2 Zur numerischen Lösung strukturmechanischer Probleme

Der Zustand eines Kontinuums wird in der Mechanik durch ein partielles Randwertproblem beschrieben. Die Differentialgleichungen enthalten die Kinematik, das Stoffgesetz und das Gleichgewicht. Die Lösung solcher Probleme kann im allgemeinen Fall nicht mehr analytisch erfolgen, sodaß häufig Diskretisierungsstrategien angewandt werden, die das partielle Randwertproblem in ein im allgemeinen Fall nichtlineares algebraisches Gleichungssystem überführen.

In diesem Abschnitt soll nun das allgemeine Vorgehen dargestellt werden, indem zunächst die kontinuumsmechanischen Grundgleichungen formuliert werden. Für eine detailliertere Darstellung diesbezüglich wird auf das Schrifttum, z.B. [141, 26, 9] oder [11] verwiesen. Nach der Einführung einer Diskretisierung mittels Finiter Elemente wird dann das nichtlineare algebraische Gleichungssystem aufgestellt, und es werden Verfahren zur Lösung dieser Probleme beschrieben. Siehe hierzu z.B. [156, 7, 25] oder [66].

### 2.1 Grundgleichungen der Kontinuumsmechanik

#### 2.1.1 Kinematik und Verzerrung

Die Lage eines materiellen Punktes des Körpers  $B$  in der unverformten Ausgangskonfiguration bzw. Referenzkonfiguration, siehe Abbildung 2.1, läßt sich in einer festen kartesischen Basis  $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$  durch den Ortsvektor

$$\mathbf{X} = [X_1, X_2, X_3], \quad (2.1)$$

die einer verformten Konfiguration bzw. Momentankonfiguration durch den Vektor

$$\mathbf{x} = [x_1, x_2, x_3] \quad (2.2)$$

darstellen. Die Bewegung dieses Punktes von der Ausgangskonfiguration  $\mathbf{X}$  ( $t = 0$ ) zur Momentankonfiguration  $\mathbf{x}$  ( $t > 0$ ) kann mathematisch als Abbildung

$$\mathbf{x} = \chi(\mathbf{X}, t) \quad (2.3)$$

beschrieben werden. Diese Abbildung muß bijektiv (da der Körper sich nicht selbst durchdringen darf) und stetig (benachbarte materielle Punkte bleiben immer benachbart) sein. Die Zeit  $t$  in Gleichung (2.3) soll eine zeitliche Abfolge der Bewegung des materiellen Punktes ermöglichen, auch wenn es sich um einen statischen, zeitinvarianten Vorgang handelt. Die Verschiebung  $\mathbf{u}$  ist gegeben durch (Abbildung 2.1)

$$\mathbf{u} = \mathbf{x} - \mathbf{X}. \quad (2.4)$$

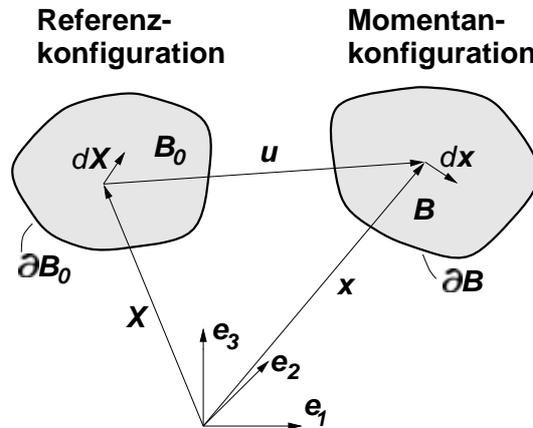


Abbildung 2.1: Kinematik des Kontinuums

Die Beziehung zwischen dem Linienelement  $d\mathbf{X}$  der Referenzkonfiguration und dem entsprechenden Linienelement  $d\mathbf{x}$  in der Momentankonfiguration lautet

$$d\mathbf{x} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} \cdot d\mathbf{X} = \text{Grad } \mathbf{x} \cdot d\mathbf{X} = \mathbf{F} \cdot d\mathbf{X}. \quad (2.5)$$

Diese Beziehung liefert mit dem Verschiebungsgradienten

$$\mathbf{H} = \text{Grad } \mathbf{u} = \frac{\partial \mathbf{u}}{\partial \mathbf{X}} \quad (2.6)$$

und mit (2.4) den Deformationsgradienten

$$\mathbf{F} = \mathbf{I} + \text{Grad } \mathbf{u} = \mathbf{I} + \mathbf{H}. \quad (2.7)$$

Für eine Starrkörperverschiebung mit  $\text{Grad } \mathbf{u} = \mathbf{0}$  nimmt der Deformationsgradient den Wert  $\mathbf{F} = \mathbf{I}$  an, und ist damit nicht als Maß für Verzerrungen geeignet. Der Green-Lagrange Verzerrungstensor

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T \mathbf{F} - \mathbf{I}) \quad (2.8)$$

hingegen, den man aus der Betrachtung der Differenz der Linienelement-Quadrate (Abstand zweier Körperpunkte)  $d\mathbf{x} \cdot d\mathbf{x} - d\mathbf{X} \cdot d\mathbf{X}$  erhält, nimmt für eine reine Starrkörperbewegung wie gewünscht den Wert  $\mathbf{0}$  (keine Verzerrung) an. Im Schrifttum existieren weitere alternative Definitionen von Verzerrungsmaßen wie z.B. der Euler-Almansi Verzerrungstensor, der sich nicht auf die Referenzkonfiguration, sondern auf die Momentankonfiguration bezieht.

### 2.1.2 Bilanzgleichungen

#### Massenbilanz

Die Masse eines Körpers berechnet sich als Volumenintegral über die stetige Massendichte  $\rho$ . Wenn im Innern des Körpers kein Zuwachs oder Verlust von Masse während der Deformation stattfindet und über die Oberfläche keine Masse ausgetauscht werden kann, gilt somit

$$m = \int_B \rho \, dv = \int_{B_0} \rho_0 \, dV. \quad (2.9)$$

Kleine Buchstaben kennzeichnen in (2.9) die Momentankonfiguration, große Buchstaben bzw. ein Index "0" die Referenzkonfiguration. Durch Anwendung der Transformationsvorschrift der Volumenelemente

$$dv = \det \mathbf{F} \, dV \quad (2.10)$$

erhält man die lokale Form der Massenerhaltung

$$\rho_0 = \det \mathbf{F} \, \rho. \quad (2.11)$$

Aus Gründen der Vollständigkeit wird hier ebenfalls die Transformationsvorschrift für Flächenelemente

$$da = \det \mathbf{F} \, \mathbf{F}^{-T} \cdot dA \quad (2.12)$$

angegeben.

#### Impulsbilanz

Der Impuls  $\mathbf{i}$  ist definiert als

$$\mathbf{i} = \int_B \rho \mathbf{\dot{x}} \, dv. \quad (2.13)$$

Die Summe der am Körper  $B$  angreifenden Kräfte  $\mathbf{f}$  besteht aus Oberflächenkräften und Volumenkräften

$$\mathbf{f} = \int_B \rho \mathbf{b} \, dv + \int_{\partial B} \mathbf{t} \, da. \quad (2.14)$$

Hierbei bezeichnet  $\mathbf{b}$  die eingepprägten Volumenkräfte und  $\mathbf{t}$  den auf die Flächeneinheit  $da$  bezogenen Spannungsvektor. Die Spannungen  $\mathbf{t}$  können mittels des Cauchy Theorems  $\mathbf{t} =$

$\mathbf{T}\mathbf{n}$  aus dem Cauchy Spannungstensor  $\mathbf{T}$  und dem Normalenvektor  $\mathbf{n}$  berechnet werden. Der Impulssatz lautet

$$\dot{\mathbf{i}} = \mathbf{f}. \quad (2.15)$$

Durch Anwendung des Gaußschen Integralsatzes (A.4) auf das Oberflächenintegral in (2.14) und unter Beachtung von (2.11) und (2.12) ergibt sich

$$\int_B \rho \ddot{\mathbf{x}} \, dv = \int_{\partial B} \mathbf{T}\mathbf{n} \, da + \int_B \rho \mathbf{b} \, dv = \int_B (\text{Div } \mathbf{P} + \rho_0 \mathbf{b}) \, dV = \mathbf{f} \quad (2.16)$$

mit dem unsymmetrischen 1. Piola Kirchhoff Spannungstensor

$$\mathbf{P} = \det \mathbf{F} \mathbf{T} \mathbf{F}^{-T}. \quad (2.17)$$

Die lokale Formulierung des Impulssatzes in der Momentankonfiguration lautet schließlich

$$\rho_0 \ddot{\mathbf{x}} = \text{Div } \mathbf{P} + \rho_0 \mathbf{b}. \quad (2.18)$$

### Drehimpulsbilanz

Der Drehimpuls bzw. Drall  $\mathbf{L}$  eines Körpers ist das Moment des Impulses eines Körpers bezüglich eines beliebigen raumfesten Punktes, der ohne Einschränkung der Allgemeinheit als Ursprung des raumfesten Koordinatensystems gewählt werden kann:

$$\mathbf{L} = \int_B \rho \mathbf{x} \times \dot{\mathbf{x}} \, dv = \int_{B_0} \rho_0 \mathbf{x} \times \dot{\mathbf{x}} \, dV. \quad (2.19)$$

Der Satz von der Erhaltung des Drehimpulses besagt, daß die Zeitableitung des Dralls dem aus Volumen- und Oberflächenkräften resultierenden Moment  $\mathbf{m}$  entspricht:

$$\dot{\mathbf{L}} = \int_B \rho \mathbf{x} \times \ddot{\mathbf{x}} \, dv = \int_{\partial B} \mathbf{x} \times \mathbf{T} \cdot \mathbf{n} \, da + \int_B \rho \mathbf{x} \times \mathbf{b} \, dv = \mathbf{m}. \quad (2.20)$$

Die lokale Form von (2.20) erhält man wiederum durch Anwendung des Gaußschen Integralsatzes (A.4), und mit dem Bezug auf die Referenzkonfiguration erhält man

$$\rho_0 \mathbf{x} \times \ddot{\mathbf{x}} = \text{Div}(\mathbf{x} \times \mathbf{P}) + \rho_0 \mathbf{x} \times \mathbf{b}. \quad (2.21)$$

Mit der Rechenregel (A.2) für die Berechnung der Divergenz und mit Gleichung (2.18) kann Gleichung (2.21) zu

$$\begin{aligned} \mathbf{x} \times (\rho_0 \ddot{\mathbf{x}} - \rho_0 \mathbf{b} - \text{Div } \mathbf{P}) - \text{Grad } \mathbf{x} \times \mathbf{P} &= -\mathbf{F} \times \mathbf{P} \\ &= -\mathbf{I} \times (\mathbf{F} \cdot \mathbf{P}^T) = \mathbf{0} \end{aligned} \quad (2.22)$$

umgeformt werden (siehe z.B. [26]). Hieraus folgt die Symmetrie des Tensorproduktes  $\mathbf{F}\mathbf{P}^T = \mathbf{P}\mathbf{F}^T$  und damit dann auch die Symmetrie des nachfolgend beschriebenen Spannungstensors.

### 2.1.3 Spannungsmaße und Stoffgesetz

In Abschnitt 2.1.2 wurde bereits der Cauchy Spannungstensor und der unsymmetrische 1. Piola Kirchhoff Spannungstensor eingeführt. Der symmetrische 2. Piola Kirchhoff Spannungstensor

$$\mathbf{S} = \mathbf{F}^{-1} \mathbf{P} = \det \mathbf{F} \mathbf{F}^{-1} \mathbf{T} \mathbf{F}^{-T} \quad (2.23)$$

ist ein zum Green Lagrange Verzerrungstensor  $\mathbf{E}$  äquivalentes Spannungsmaß. Anschaulich ist er aber nicht interpretierbar.

Das Stoffgesetz stellt einen konstitutiven Zusammenhang zwischen den Verzerrungen und den Spannungen (z.B. zwischen den Green Lagrange Verzerrungen  $\mathbf{E}$  und den 2. Piola Kirchhoff Spannungen  $\mathbf{S}$ ) her. Für ein elastisches Material gilt

$$\mathbf{S} = f(\mathbf{E}), \quad (2.24)$$

der Spannungstensor ist eine Funktion ausschließlich des Verzerrungstensors. Ein solches Material wird als "Cauchy-elastisch" bezeichnet. Wird weiterhin die Wegunabhängigkeit der durch die Spannungen verrichteten Arbeit gefordert (d.h. konservative Systeme), läßt sich der Zusammenhang zwischen den Spannungen und den Verzerrungen aus einem Potential ableiten. Das Material wird dann als hyperelastisch oder "Green-elastisch" bezeichnet. Es existiert dann eine Formänderungsenergiefunktion  $W(\mathbf{E})$ , sodaß

$$\mathbf{P} = \frac{\partial}{\partial \mathbf{E}} W(\mathbf{E}) \quad (2.25)$$

gilt.

Für ein St. Venant-Kirchhoff-Material ist

$$W(\mathbf{E}) = \frac{1}{2} \mathbf{E} \mathbf{C} \mathbf{E} \quad (2.26)$$

mit dem vierstufigen Materialtensor

$$\mathbf{C} = \lambda \mathbf{1} \otimes \mathbf{1} + 2\mu \mathbf{I}, \quad (2.27)$$

wobei  $\lambda$  und  $\mu$  die beiden Lamé Konstanten darstellen, die mit dem Elastizitätsmodul  $E$  und der Querkontraktionszahl  $\nu$  über

$$\mu = G = \frac{E}{2(1 + \nu)} \quad \text{und} \quad \lambda = \frac{E\nu}{(1 + \nu)(1 - 2\nu)} \quad (2.28)$$

verknüpft sind. Der 2. Piola Kirchhoff Spannungstensor wird dann mit

$$\mathbf{S} = \frac{\partial W}{\partial \mathbf{E}} = \mathbf{C} \mathbf{E} \quad (2.29)$$

berechnet.

Neben dem hier beschriebenen elastischen Materialgesetz sind auch weitere, nichtlineare Stoffgesetze wie z.B. Mooney-Rivlin, Ogden, Neo-Hooke oder Elastoplastizität möglich. Diese sollen im Rahmen der vorliegenden Arbeit nicht weiter berücksichtigt werden.

### 2.1.4 Das Prinzip der virtuellen Verschiebungen

Mit der Impulsbilanz (2.18), der konstitutiven Beziehung (2.29), der Kinematik (2.8) und den dazugehörigen Spannungs- und Verschiebungsrandbedingungen

$$\begin{aligned} \mathbf{t}_0 &= \mathbf{P}\mathbf{n}_0 \quad \text{auf } \partial B_\sigma \\ \mathbf{u} &= \hat{\mathbf{u}} \quad \text{auf } \partial B_u \end{aligned} \quad (2.30)$$

können nun kontinuumsmechanische Probleme mathematisch formuliert werden. Hierin bezeichnet  $\mathbf{n}_0$  die Normalenrichtung des Randes. Eine analytische Lösung dieser Probleme ist aber wegen der schwierigen Erfüllung beliebiger Randbedingungen auf wenige Sonderfälle beschränkt.

Mit dem Prinzip der virtuellen Verschiebungen bzw. gewichteter Residuen geht man daher auf die schwache Form der Differentialgleichung über, die die Gleichgewichtsbeziehung nur noch im integralen Mittel, die Randbedingungen hingegen exakt erfüllt. Dabei entsteht aus der lokalen Impulsbilanz (2.18) durch Multiplikation mit Testfunktionen  $\delta\mathbf{u}$ , den virtuellen Verschiebungen, die den geometrischen Randbedingungen genügen, d.h.  $\delta\mathbf{u} = \mathbf{0}$  auf  $\partial B$ , und anschließender Integration die schwache Form des Gleichgewichts

$$\int_{B_0} \delta\mathbf{u}(\text{Div}\mathbf{P} + \rho_0\mathbf{b} - \rho_0\ddot{\mathbf{x}}) dV = \mathbf{0}. \quad (2.31)$$

Unter Beachtung von (A.1) und dem Gaußschen Integralsatz (A.4) wird daraus

$$\int_{B_0} (-\mathbf{P}\text{Grad}\delta\mathbf{u} + \rho_0\mathbf{b}\delta\mathbf{u}) dV + \int_{\partial B_0} \mathbf{P}^T \delta\mathbf{u}\mathbf{n}_0 dA - \int_{B_0} \rho_0\ddot{\mathbf{x}}\delta\mathbf{u} dV = \mathbf{0} \quad (2.32)$$

und weiter mit (2.7) und der Beziehung

$$\text{Grad}\delta\mathbf{u} = \delta\text{Grad}\mathbf{u} = \delta(\mathbf{F} - \mathbf{I}) = \delta\mathbf{F} \quad (2.33)$$

erhält man

$$-\int_{B_0} \mathbf{P}\delta\mathbf{F} dV + \int_{B_0} \rho_0\mathbf{b}\delta\mathbf{u} dV + \int_{\partial B_0} \mathbf{t}_0\delta\mathbf{u} dA - \int_{B_0} \rho_0\ddot{\mathbf{x}}\delta\mathbf{u} dV = \mathbf{0}. \quad (2.34)$$

Wegen

$$\mathbf{P}\delta\mathbf{F} = \mathbf{F}\mathbf{S}\delta\mathbf{F} = \delta\mathbf{F}^T \mathbf{F}\mathbf{S} = \frac{1}{2}(\delta\mathbf{F}^T \mathbf{F} + \mathbf{F}^T \delta\mathbf{F}) \cdot \mathbf{S} \quad (2.35)$$

und

$$\delta \mathbf{E} = \frac{1}{2} \delta (\mathbf{F}^T \mathbf{F} - \mathbf{I}) = \frac{1}{2} (\delta \mathbf{F}^T \mathbf{F} + \mathbf{F}^T \delta \mathbf{F}) \quad (2.36)$$

gilt

$$G(\mathbf{u}, \delta \mathbf{u}) = \int_{B_0} \mathbf{S} \delta \mathbf{E} dV - \int_{B_0} \rho_0 \mathbf{b} \delta \mathbf{u} dV - \int_{\partial B_\sigma} \mathbf{t}_0 \delta \mathbf{u} dA + \int_{B_0} \rho_0 \ddot{\mathbf{x}} \delta \mathbf{u} dV = 0. \quad (2.37)$$

Für statische Probleme sei an dieser Stelle ein weiterer Zugang über das Potential  $\Pi$  angegeben, sofern eine Energiedichtefunktion  $W(\mathbf{E})$  existiert und nur konservative Lasten zugelassen werden:

$$\Pi(\mathbf{u}) = \int_{B_0} W(\mathbf{E}) dV - \int_{B_0} \rho_0 \mathbf{b} \mathbf{u} dV - \int_{\partial B_\sigma} \mathbf{t}_0 \mathbf{u} dA. \quad (2.38)$$

Mit dem Prinzip vom stationären Wert der potentiellen Energie

$$\delta \Pi(\mathbf{u}, \delta \mathbf{u}) = 0 \quad (2.39)$$

folgt

$$\delta \Pi(\mathbf{u}, \delta \mathbf{u}) = \int_{B_0} \frac{\partial W(\mathbf{E})}{\partial \mathbf{E}} \delta \mathbf{E} dV - \int_{B_0} \rho_0 \mathbf{b} \delta \mathbf{u} dV - \int_{\partial B_\sigma} \mathbf{t}_0 \delta \mathbf{u} dA = 0 \quad (2.40)$$

und mit einem elastischen Material

$$\frac{\partial W(\mathbf{E})}{\partial \mathbf{E}} = \mathbf{S} \quad (2.41)$$

ergibt sich schließlich analog zu (2.37)

$$\delta \Pi(\mathbf{u}, \delta \mathbf{u}) = \int_{B_0} \mathbf{S} \delta \mathbf{E} dV - \int_{B_0} \rho_0 \mathbf{b} \delta \mathbf{u} dV - \int_{\partial B_\sigma} \mathbf{t}_0 \delta \mathbf{u} dA = 0. \quad (2.42)$$

D.h. für statische Probleme entspricht die schwache Form der Variation des Potentials.

### 2.1.5 Linearisierung des Prinzips der virtuellen Verschiebungen

Um die obigen im Allgemeinen nichtlinearen Gleichungen einer numerischen Lösungsprozedur, wie z.B. dem Newton-Raphson Verfahren (siehe Abschnitt 2.3.1) zugänglich zu

machen, muß Gleichung (2.37) linearisiert werden (siehe z.B. [68, 152, 54]). Die Linearisierung erfolgt mittels einer Reihenentwicklung an einer Stelle  $\bar{\mathbf{u}}$ , wobei nach dem linearen Glied abgebrochen wird:

$$G(\mathbf{u}, \delta \mathbf{u}) = G(\bar{\mathbf{u}}, \delta \mathbf{u}) + DG(\bar{\mathbf{u}}, \delta \mathbf{u})\Delta \mathbf{u} + R(\Delta \mathbf{u}). \quad (2.43)$$

$R(\Delta \mathbf{u})$  stellt Terme höherer Ordnung dar, die vernachlässigt werden. Der Term  $DG(\bar{\mathbf{u}}, \delta \mathbf{u})$  bezeichnet die Anwendung der Gateaux Ableitung

$$DG(\bar{\mathbf{u}}, \delta \mathbf{u})\Delta \mathbf{u} = \frac{d}{d\varepsilon}[G(\bar{\mathbf{u}} + \varepsilon\Delta \mathbf{u})]_{\varepsilon=0} = \frac{\partial G(\bar{\mathbf{u}})}{\partial \mathbf{u}} \cdot \Delta \mathbf{u}. \quad (2.44)$$

Unter ausschließlicher Berücksichtigung richtungstreuer Belastungen [128] beschränkt sich die Linearisierung auf den ersten Term in (2.37), siehe [68]. Der Beschleunigungsterm ist bereits linear. Bei Anwendung der Produktregel erhält man

$$DG(\bar{\mathbf{u}}, \delta \mathbf{u}) = \int_{B_0} \left[ \delta \mathbf{E} \frac{\partial \mathbf{S}}{\partial \mathbf{E}} D\mathbf{E} + \mathbf{S} D\delta \mathbf{E} \right] dV, \quad (2.45)$$

der erste Term im Integral wird dabei als materieller, der zweite als geometrischer Anteil bezeichnet.

## 2.2 Finite Element Diskretisierung

Um das in den vorigen Abschnitten formulierte, kontinuierliche, allgemein räumliche Problem einer numerischen Lösung zugänglich zu machen, wird sowohl die Geometrie des Körpers  $B$  in der Ausgangskonfiguration, als auch das kontinuierliche Verschiebungsfeld  $\mathbf{u}$  diskretisiert. Dabei wird die Lösung nur an diskreten Stellen, den sogenannten Knoten, mit den Ortsvektoren

$$\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_K, \dots, \mathbf{X}_N \quad (2.46)$$

berechnet und das Problem dadurch in ein algebraischen Gleichungssystem überführt.

Das Gebiet  $B$  wird hierbei in nichtüberlappende Teilgebiete unterteilt, den ‘‘Finiten Elementen’’:

$$B_0 \approx B_0^h = \bigcup_{e=1}^{n_{el}} B_{0_e}^h \quad \text{mit} \quad B_{0_{e1}}^h \cap B_{0_{e2}}^h = \emptyset \quad \text{für} \quad e1 \neq e2. \quad (2.47)$$

Mit dieser Diskretisierung, dem Übergang von einer kontinuierlichen zu einer diskreten Beschreibung gehen Fehler, wie z.B. in der Beschreibung der Geometrie oder den Randbedingungen, einher, weshalb die diskreten Größen mit einem hochgestellten  $h$  gekennzeichnet werden sollen. Die aktuellen Ortsvektoren  $\mathbf{x}_K$  der Knoten werden mit (2.4) aus

$$\mathbf{x}_K = \mathbf{X}_K + \mathbf{u}_K \quad (2.48)$$

bestimmt.

### 2.2.1 Das isoparametrische Konzept

Eine häufig angewandte Technik zur Diskretisierung stellt das isoparametrische Konzept dar. Bei einem beliebigen Beanspruchungszustand werden die Orte im Elementinnern mit denselben Ansatzfunktionen interpoliert wie das Verschiebungsfeld. Dies gibt

$$\mathbf{X}_e^h = \sum_{I=1}^{I_e} N_I(\boldsymbol{\xi}) \mathbf{X}_I \quad \text{und} \quad \mathbf{u}_e^h = \sum_{I=1}^{I_e} N_I(\boldsymbol{\xi}) \mathbf{u}_I. \quad (2.49)$$

Die Ansatzfunktionen  $N(\boldsymbol{\xi})$  sind dabei im Einheitsgebiet  $B_\square$  mit den orthogonalen Koordinaten  $\boldsymbol{\xi}$  definiert mit  $-1 \leq \xi_j \leq +1$ . Für die Interpolationsfunktionen  $N(\boldsymbol{\xi})$  können sowohl Funktionen niedriger Ordnung wie z.B. lineare Funktionen, als auch Funktionen höherer Ordnung gewählt werden, siehe hierzu das Standardschrifttum [7, 156, 127]. Wesentlich ist jedoch, daß die Funktionen  $N_I$  beim Knoten  $I$  den Wert 1, an allen anderen Knoten des Elements jedoch den Wert 0 besitzen:

$$N_I(\boldsymbol{\xi} = \boldsymbol{\xi}_J) = \delta_{IJ}, \quad (2.50)$$

sodaß an den diskreten Knoten ausschließlich die diskreten Knotenwerte vorliegen. Dann gilt

$$\begin{aligned} \mathbf{x}_e^h &= \sum_{I=1}^{I_e} N_I(\boldsymbol{\xi}) \mathbf{x}_I = \sum_{I=1}^{I_e} N_I(\boldsymbol{\xi}) [\mathbf{X}_I + \mathbf{u}_I] = \sum_{I=1}^{I_e} N_I(\boldsymbol{\xi}) \mathbf{X}_I + \sum_{I=1}^{I_e} N_I(\boldsymbol{\xi}) \mathbf{u}_I \\ &= \mathbf{X}_e^h + \mathbf{u}_e^h. \end{aligned} \quad (2.51)$$

### 2.2.2 Die diskrete schwache Form

Mit der oben beschriebenen Diskretisierung wird anschließend in eine Matrixdarstellung übergegangen. Das diskrete Verschiebungsfeld  $\mathbf{u}_e^h$  wird im Folgenden durch den Vektor der Knotenverschiebungen  $\mathbf{d}$  dargestellt, die virtuellen Verschiebungen  $\delta \mathbf{u}_e^h$  durch den Vektor  $\mathbf{v}$ . Dann liefert die Variation des Verzerrungstensors  $\delta \mathbf{E}$

$$\delta \mathbf{E}^h = \sum_{I=1}^{I_e} \mathbf{B}(\mathbf{d}_I) \mathbf{v}_I \quad \text{mit} \quad \mathbf{B}(\mathbf{d}_I) = \frac{\partial N_I(\mathbf{d}_I)}{\partial \mathbf{x}}. \quad (2.52)$$

Für ein konkretes Beispiel siehe hierzu Anhang B. Führt man (2.52) in Gleichung (2.42) ein, so folgt

$$\begin{aligned} \bigcup_{e=1}^{n_{el}} \sum_{I=1}^{I_e} \left[ \int_{B_0} \mathbf{B}^T(\mathbf{d}_I) \mathbf{S} \, dV - \int_{B_0} N_I^T \rho_0 \mathbf{b} \, dV - \int_{\partial B_0} N_I^T \mathbf{t}_0 \, dA \right. \\ \left. + \sum_{K=1}^{I_e} \int_{B_0} \rho_0 \ddot{\mathbf{d}}_I N_I^T N_K \, dV \right] = 0. \end{aligned} \quad (2.53)$$

Nach Einführung der Abkürzungen

$$\mathbf{r}(\mathbf{d}) = \bigcup_{e=1}^{n_{el}} \sum_{I=1}^{I_e} \int_{B_0} \mathbf{B}^T(\mathbf{d}_I) \mathbf{S} dV, \quad (2.54)$$

$$\bar{\mathbf{p}} = \bigcup_{e=1}^{n_{el}} \sum_{I=1}^{I_e} \left[ \int_{B_0} N_I^T \rho_0 \mathbf{b} dV + \int_{\partial B_0} N_I^T \mathbf{t}_0 dA \right] \quad (2.55)$$

und der konsistenten Massenmatrix

$$\mathbf{M} = \bigcup_{e=1}^{n_{el}} \left[ \sum_{I=1}^{I_e} \sum_{K=1}^{I_e} \int_{B_0} \rho_0 \mathbf{N}_I^T \mathbf{N}_K dV \right], \quad (2.56)$$

die analog zur Steifigkeitsmatrix erhalten wird, liefert das Fundamentallemma der Variationsrechnung das nichtlineare, algebraische Gleichungssystem

$$\mathbf{G}(\mathbf{d}) = \mathbf{M} \ddot{\mathbf{d}} + \mathbf{r}(\mathbf{d}) - \bar{\mathbf{p}} = \mathbf{0}. \quad (2.57)$$

Dabei stellt  $\mathbf{r}(\mathbf{d})$  die inneren Kräfte in Abhängigkeit der Knotenverschiebungen  $\mathbf{d}$  und  $\bar{\mathbf{p}}$  die äußeren Lasten dar. Für statische Berechnungen und unter der Annahme richtungstreu eingepprägter äußerer Kräfte kann in (2.57) der Lastvektor  $\bar{\mathbf{p}}$  auch als Grundlastvektor  $\mathbf{p}$  multipliziert mit einem skalaren Parameter  $\lambda$ , dem sogenannten Lastfaktor, formuliert werden:

$$\mathbf{G}(\mathbf{d}) = \mathbf{r}(\mathbf{d}) - \lambda \mathbf{p} = \mathbf{0} \quad \text{mit} \quad \bar{\mathbf{p}} = \lambda \mathbf{p}. \quad (2.58)$$

Somit herrscht statisches Gleichgewicht, wenn bei vorgegebenem  $\lambda$ , dem Steuerparameter des Problems, die inneren und die eingepprägten äußeren Kräfte gleich groß sind.

## 2.3 Lösungsverfahren

### 2.3.1 Das Newton-Raphson Verfahren

Zur Lösung der aus statischen Problemen resultierenden nichtlinearen Gleichungssysteme (2.58) werden meist Newton-artige Verfahren angewandt. Dabei wird das diskrete nichtlineare Gleichungssystem (2.58) bei festgehaltenem Steuerparameter  $\lambda$  an einer Stelle  $\bar{\mathbf{d}}$  in eine Taylorreihe entwickelt, die nach dem linearen Term abgebrochen wird:

$$\mathbf{G}(\bar{\mathbf{d}} + \Delta \mathbf{d}) = \mathbf{G}(\bar{\mathbf{d}}) + \left. \frac{\partial \mathbf{G}(\bar{\mathbf{d}})}{\partial \mathbf{d}} \right|_{\bar{\mathbf{d}}} \cdot \Delta \mathbf{d} = \mathbf{0}. \quad (2.59)$$

Man erhält also das lineare Gleichungssystem

$$D\mathbf{G}(\bar{\mathbf{d}})\Delta\mathbf{d} = -\mathbf{G}(\bar{\mathbf{d}}). \quad (2.60)$$

Die Berechnung der Linearisierung erfolgt gemäß (2.44), da sie formal auf die gleiche Art gebildet werden kann, wie die Variation. Die in Gleichung (2.45) angegebene Linearisierung soll fortan in ihrer diskretisierten Form als Matrix  $\mathbf{K}_T$ , die sogenannte tangentielle Steifigkeitsmatrix, bezeichnet werden. Nun kann (2.60) als

$$\mathbf{K}_T \Delta\mathbf{d} = -\mathbf{G}(\bar{\mathbf{d}}) \quad (2.61)$$

geschrieben werden. Die bei der Linearisierung vernachlässigten Terme höherer Ordnung müssen iterativ berücksichtigt werden. Mit dem Newton-Raphson Verfahren wird in jeder Iteration  $i$  das lineare Gleichungssystem

$$\mathbf{K}_T(\mathbf{d}_i) \Delta\mathbf{d}_i = -\mathbf{G}(\mathbf{d}_i), \quad i = 0, 1, 2, \dots \quad (2.62)$$

gelöst, und der Verschiebungsvektor  $\mathbf{d}_i$  mit dem Inkrement  $\Delta\mathbf{d}_i$  aufdatiert. Die Vorschrift lautet

$$\mathbf{d}_{i+1} = \mathbf{d}_i + \Delta\mathbf{d}_i. \quad (2.63)$$

Dieser Vorgang wird solange wiederholt, bis ein geeignetes Abbruchkriterium erfüllt ist. Dann wird der Steuerparameter geändert, z.B. die äußere Last mittels des skalaren Lastparameters  $\lambda$  erhöht und die nächste Gleichgewichtslage berechnet.

Eine besondere Eigenschaft des Newton Verfahrens ist seine quadratische Konvergenz zur Lösung im Falle einer konsistenten Linearisierung. Allerdings ist die Konvergenz nicht global, sodaß nicht beliebig große Lastschritte erlaubt sind. Erwähnenswert ist in diesem Zusammenhang die Erschwernis, daß eine konkrete Angabe des Parameters  $\lambda$  zur Vorgabe einer äußeren Last zu Beginn der nichtlinearen Berechnung ohne Kenntnis des Konvergenzradius erfolgen muß. Alternativ ist es möglich, andere Größen, wie z.B. Verschiebungen vorzugeben, die allerdings durch einen Block-Gauß Schritt auf die rechte Seite des linearen Gleichungssystems gebracht werden können, sodaß implizit auch hier eine Lastvorgabe stattfindet.

Als Abbruchkriterien des iterativen Algorithmus werden in der Strukturmechanik üblicherweise Verschiebungsnormen, Normen der Residuen oder Energienormen herangezogen. Im Rahmen der in dieser Abhandlung durchgeführten Untersuchungen wird als Abbruchkriterium eine Energienorm verwendet, siehe [8]:

$$\frac{\Delta\mathbf{d}_i^T (\mathbf{r} - \lambda_{i-1}\mathbf{p})}{\Delta\mathbf{d}_1^T (\lambda_1\mathbf{r})} < \varepsilon. \quad (2.64)$$

Es wird ausnahmslos  $\varepsilon = 1.0 \cdot 10^{-12}$  gewählt, wodurch eine hohe Genauigkeit der ausiterierten Gleichgewichtslage erreicht wird.

Um die numerische Effizienz zu steigern, werden oft auch sogenannte modifizierte Newton Verfahren eingesetzt, bei denen die tangentielle Steifigkeitsmatrix  $\mathbf{K}_T$  nicht in jeder Newton Iteration neu berechnet wird, sondern z.B. nur bei der 1. Iteration. Beim Einsatz direkter Verfahren zur Lösung des linearen Gleichungssystems (2.62) hat dies den Vorteil, daß die Steifigkeitsmatrix während der Newton Iteration nur einmal faktorisiert werden muß. Das quadratische Konvergenzverhalten ist allerdings bei den modifizierten Verfahren nicht gegeben.

Auch weitere Varianten, bei denen die Steifigkeitsmatrix alle  $j$  Iterationen neu berechnet wird sind möglich. In [74] werden solche Verfahren als Shamanskii Methode bezeichnet. Interessant ist hierbei, daß nach [74] eine höhere Konvergenzordnung, nämlich  $j + 1$  erreicht wird. Da aber nur die echten Newton Schritte gezählt werden, ist mit dieser Methode nur dann eine effizientere Lösung erreichbar, wenn die modifizierten Newton Schritte mit wenig Aufwand durchgeführt werden können, was im Allgemeinen nur unter Verwendung direkter Löser gegeben ist.

Eine weitere Möglichkeit zur Lösung nichtlinearer algebraischer Gleichungen sind die sogenannten Quasi-Newton Verfahren. Zu einer effizienten Lösung werden hierzu direkte Lösungsverfahren für lineare Gleichungssysteme eingesetzt. Zu Beginn eines Lastschrittes wird die tangentielle Steifigkeitsmatrix aufgestellt und faktorisiert. Mittels der Sherman-Morrison Formel wird dann die inverse Steifigkeitsmatrix aufdatiert und so ein erneutes Aufstellen und Faktorisieren der tangentiellen Steifigkeitsmatrix vermieden, siehe z.B. [129]. Das Konvergenzverhalten ist dann aber nicht mehr quadratisch, wie beim ursprünglichen Newton Verfahren.

### 2.3.2 Das Bogenlängenverfahren

Wie in Abschnitt 2.3.1 beschrieben, muß zur Anwendung des Newton-Raphson Verfahrens ein Steuerparameter, d.h. ein Last- oder ein Verschiebungsparameter konkret vorgegeben werden. Bei Problemen mit singulären Stellen, siehe Abbildung 2.2, ist dies nicht immer möglich. Hier ist sowohl die ausschließliche Last- als auch die Verschiebungsvorgabe nur begrenzt möglich; mit der Lastvorgabe kann der nichtlineare Gleichgewichtspfad bis zum Punkt "1", mit der Verschiebungsvorgabe nur bis Punkt "2" verfolgt werden. Wird an der Gleichgewichtslage "1" der Lastparameter weiter erhöht, so wird aufgrund der lokalen Konvergenzeigenschaften des Newton-Raphson Verfahrens keine Konvergenz mehr erzielt, analog verhält es sich bei einer weiteren Steigerung der Verschiebungsvorgabe am Punkt "2".

Eine Berechnung des gesamten Last Verformungsverhaltens des in Abbildung 2.2 gezeigten (und in Abschnitt 3.2 genauer charakterisierten) Problems wird erst durch den Einsatz sogenannter Bogenlängenverfahren möglich, siehe z.B. [23, 114, 118] oder [129]. Dabei wird der skalare Steuerparameter als zusätzliche Unbekannte betrachtet. Um nun diese zusätzliche Unbekannte zu bestimmen, muß eine Nebenbedingung

$$f(\mathbf{d}, \lambda) = 0 \tag{2.65}$$

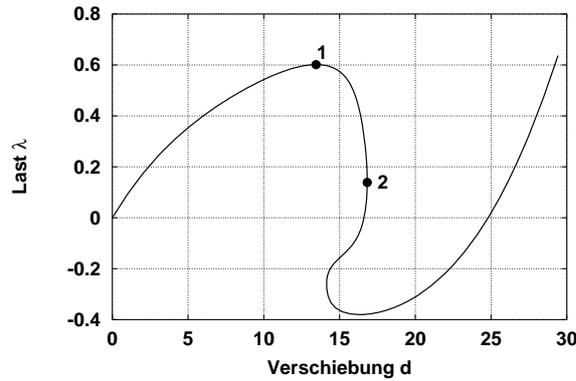


Abbildung 2.2: Nichtlineare Last-Verformungskurve mit “snap-back”

eingeführt werden. Die feste Vorgabe von  $\lambda$  wird durch Gleichung (2.65) ersetzt. Da diese Gleichung durchaus nichtlinear sein kann, kann es sinnvoll sein, auch sie im Rahmen der nichtlinearen Lösungsprozedur durch

$$\mathbf{v}_i^T \Delta \mathbf{d}_i + \alpha_i \Delta \lambda_i = -f(\mathbf{d}_i, \lambda_i) \quad (2.66)$$

zu linearisieren [132]. Dabei ist

$$\mathbf{v}_i^T = \frac{d}{d\varepsilon} [f(\mathbf{d}_i + \varepsilon \Delta \mathbf{d}_i, \lambda_i)]_{\varepsilon=0} \quad \text{und} \quad (2.67)$$

$$\alpha_i = \frac{d}{d\varepsilon} [f(\mathbf{d}_i, \lambda_i + \varepsilon \Delta \lambda_i)]_{\varepsilon=0}. \quad (2.68)$$

Unterschiedliche Formulierungen der Nebenbedingung (2.65) und auch der Lösung in Kombination mit einem Newton Verfahren sind möglich. Eine ausführliche Übersicht befindet sich z.B. in [129].

Insgesamt entsteht in jeder Newton Iteration  $i$  das unsymmetrische lineare Gleichungssystem

$$\begin{bmatrix} \mathbf{K} & -\mathbf{p} \\ \mathbf{v}_i^T & \alpha_i \end{bmatrix} \begin{bmatrix} \Delta \mathbf{d} \\ \Delta \lambda_i \end{bmatrix} = - \begin{bmatrix} \mathbf{r}(\mathbf{d}_i) - (\lambda_m + \lambda_i) \mathbf{p} \\ f(\mathbf{d}_i, \lambda_i) \end{bmatrix} = - \begin{bmatrix} \mathbf{f}_i \\ f \end{bmatrix}, \quad (2.69)$$

wenn der skalare Steuerparameter  $\lambda$  in die Anteile  $\lambda = \lambda_m + \lambda_i + \Delta \lambda$  aufgeteilt wird.  $\lambda_m$  stellt darin den Parameter des letzten auskonvergierten Lastniveaus,  $\lambda_i$  den inkrementellen Lastfaktor und  $\Delta \lambda$  den iterativen Anteil dar, siehe [129]. Üblicherweise wird Gleichung (2.69) durch

$$\Delta \lambda_i = - \frac{f + \mathbf{v}_i^T \Delta \mathbf{d}^{II}}{\alpha_i + \mathbf{v}_i^T \Delta \mathbf{d}^I} \quad \text{und} \quad \Delta \mathbf{d}_i = \Delta \mathbf{d}^{II} + \Delta \lambda_i \Delta \mathbf{d}^I \quad (2.70)$$

gelöst, wobei die Vektoren  $\mathbf{d}^I$  und  $\mathbf{d}^{II}$  aus

$$\mathbf{K}_T \Delta \mathbf{d}^I = \mathbf{p} \quad \text{und} \quad \mathbf{K}_T \Delta \mathbf{d}^{II} = -\mathbf{f} \quad (2.71)$$

ermittelt werden. Es müssen mit dem Bogenlängenverfahren also in jeder Newton Iteration entweder zwei symmetrische lineare Gleichungssysteme (2.71) mit der tangentiellen Steifigkeitsmatrix  $\mathbf{K}_T$  als Koeffizientenmatrix oder alternativ ein unsymmetrisches lineares Gleichungssystem (2.69) gelöst werden.

### 2.3.3 Das Newmark Verfahren

In Abschnitt 2.2.2 wurden nur statische Vorgänge berücksichtigt, wie aus dem fehlenden Beschleunigungsterm in (2.58) ersichtlich. Für kinetische Prozesse wird die konsistente Massenmatrix (2.56) benötigt. Sie ist positiv definit. Häufig wird die Massenmatrix aus Effizienz- und Speicherplatzgründen diagonalisiert (*mass-lumping*). In Kombination mit dem zentralen Differenzenverfahren bietet die Diagonalisierung noch weitere Vorteile. Verschiedene Verfahren hierzu findet man z.B. in [156].

Die das Systemverhalten beschreibende nichtlineare Bewegungsgleichung (2.57) lautet nach Neuordnung der Terme und zusätzlicher Berücksichtigung von Dämpfung

$$\mathbf{M} \ddot{\mathbf{d}} + \mathbf{C} \dot{\mathbf{d}} + \mathbf{r}(\mathbf{d}) = \bar{\mathbf{p}}. \quad (2.72)$$

Durch die räumliche Diskretisierung wird das partielle Anfangsrandwertproblem auf ein diskretes Problem mit endlich vielen Freiheitsgraden zurückgeführt. Die Wahl einer Dämpfungsmatrix  $\mathbf{C}$ , die die innere Energiedissipation der mechanischen Struktur realistisch beschreibt, ist eine weitgehend offene Frage. Meistens wird sie heuristisch als konstant und geschwindigkeitsproportional (viskos) gewählt. Eine weitere Vereinfachung ist die Rayleigh-Dämpfung

$$\mathbf{C}_T = \alpha_D \mathbf{M} + \beta_D \mathbf{K}, \quad (2.73)$$

bei der die Dämpfungsmatrix als Linearkombination der Massen- und der Steifigkeitsmatrix angesetzt wird. Als Grenzfälle sind die alleinige massen- bzw. steifigkeitsproportionale Dämpfung  $\beta_D = 0$  oder  $\alpha_D = 0$  möglich. Die massenproportionale Dämpfung führt zu einer vorwiegenden Dämpfung der niedrigen, die steifigkeitsproportionale Dämpfung zu einer vorwiegenden Dämpfung der hohen Frequenzen. Zur Wahl der beiden Parameter  $\alpha_D$  und  $\beta_D$  wird an dieser Stelle z.B. auf [21] verwiesen.

Ausgehend von der bekannten Lösung des Anfangsrandwertproblems (2.72) zur Zeit  $t_n$  wird die Lösung zur Zeit  $t_{n+1} = t_n + \Delta t$  gesucht:

$$\mathbf{M} \ddot{\mathbf{d}}_{n+1} + \mathbf{r}(\mathbf{d}_{n+1}) = \mathbf{p}_{n+1}. \quad (2.74)$$

Die Dämpfung soll bei der Darstellung des Newmark Verfahrens aus Gründen der Übersichtlichkeit nicht weiter berücksichtigt werden, sie bereitet bei der Einarbeitung in das Lösungsverfahren aber keine grundsätzlichen Probleme, siehe z.B. [66].

Beim Newmark Verfahren [102] wird ein Ansatz für die gesuchten Verschiebungen  $\mathbf{d}_{n+1}$  und Geschwindigkeiten  $\dot{\mathbf{d}}_{n+1}$  zum Zeitpunkt  $t_{n+1}$  in der Form

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t \dot{\mathbf{d}}_n + \Delta t^2 \left[ \left( \frac{1}{2} - \beta \right) \ddot{\mathbf{d}}_n + \beta \ddot{\mathbf{d}}_{n+1} \right] \quad \text{und} \quad (2.75)$$

$$\dot{\mathbf{d}}_{n+1} = \dot{\mathbf{d}}_n + \Delta t \left[ (1 - \gamma) \ddot{\mathbf{d}}_n + \gamma \ddot{\mathbf{d}}_{n+1} \right], \quad (2.76)$$

gewählt. Die Abhängigkeit vom Zeitschritt wird für den Geschwindigkeitsverlauf linear, und für den Verschiebungsverlauf quadratisch angenommen. Da die Zustandsgrößen zum Zeitpunkt  $t_n$  bekannt sind, können die entsprechenden Terme in (2.75) und (2.76) zu Prädiktoren

$$\mathbf{d}_{n+1} = \mathbf{d}_{n+1}^P + \Delta t^2 \beta \ddot{\mathbf{d}}_{n+1} \quad \text{und} \quad (2.77)$$

$$\dot{\mathbf{d}}_{n+1} = \dot{\mathbf{d}}_{n+1}^P + \Delta t \gamma \ddot{\mathbf{d}}_{n+1} \quad (2.78)$$

zusammengefaßt werden. Gleichung (2.77) kann wiederum nach den gesuchten Beschleunigungen

$$\ddot{\mathbf{d}}_{n+1} = \frac{1}{\Delta t^2 \beta} [\mathbf{d}_{n+1} - \mathbf{d}_{n+1}^P] \quad (2.79)$$

aufgelöst werden, womit sich das nichtlineare Gleichungssystem (2.74) zur Zeit  $t_{n+1}$  zu

$$\frac{1}{\Delta t^2 \beta} \mathbf{M} \mathbf{d}_{n+1} + \mathbf{r}(\mathbf{d}_{n+1}) = \mathbf{p}_{n+1} + \frac{1}{\Delta t^2 \beta} \mathbf{M} \mathbf{d}_{n+1}^P \quad (2.80)$$

ergibt. Innerhalb des Newton Verfahrens wird dieses nichtlineare Gleichungssystem linearisiert, und man erhält in jeder Newton Iteration  $i$ :

$$\underbrace{\left[ \frac{1}{\Delta t^2 \beta} \mathbf{M} + \mathbf{K}_T \right]}_{\mathbf{K}^*} \Delta \mathbf{d} = -\mathbf{r}(\mathbf{d}) + \mathbf{p}_{n+1} + \frac{1}{\Delta t^2 \beta} \mathbf{M} (\mathbf{d}_{n+1}^P - \mathbf{d}_i) \quad (2.81)$$

mit der effektiven Steifigkeitsmatrix

$$\mathbf{K}^* = \frac{1}{\Delta t^2 \beta} \mathbf{M} + \mathbf{K}_T. \quad (2.82)$$

Die Verschiebungsinkremente werden in jeder Iteration zu  $\mathbf{d}_{i+1} = \mathbf{d}_i + \Delta \mathbf{d}$  aufaddiert.

Nach der Konvergenz des Newton Verfahrens gilt  $\mathbf{d}_{n+1} = \mathbf{d}_{i \text{ konv}}$  und damit lassen sich dann wiederum die Geschwindigkeiten  $\dot{\mathbf{d}}_{n+1}$  und die Beschleunigungen  $\ddot{\mathbf{d}}_{n+1}$  mit Hilfe der Gleichungen (2.79) und (2.76) berechnen.

### Wahl der Parameter $\beta$ und $\gamma$

Die bislang noch nicht diskutierten Parameter  $\beta$  und  $\gamma$  bestimmen die Genauigkeit und Stabilität der berechneten Lösung, siehe hierzu z.B. [66].

Eine Genauigkeit zweiter Ordnung wird nur für  $\gamma = 0.5$  erreicht. Für größere Werte von  $\gamma$  stellt sich eine künstliche numerische Dämpfung ein, bei der die hohen Frequenzen stärker gedämpft werden als die niedrigen. Dies kann erwünscht sein, da es dem tatsächlichen Verhalten von realen Strukturen entspricht. Nachteilig ist dabei aber, daß hierdurch die Dämpfung nicht in ihrem Einfluß auf das Ergebnis bewertet werden kann. Zudem verschlechtert sich die erzielbare numerische Genauigkeit.

Wählt man  $\beta = 0.25$ , so erhält man aus (2.75)

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \Delta t \dot{\mathbf{d}}_n + \frac{\Delta t^2}{2} \frac{\ddot{\mathbf{d}}_n + \ddot{\mathbf{d}}_{n+1}}{2}, \quad (2.83)$$

das Newmark Verfahren der konstanten mittleren Beschleunigung. Dabei stellt  $\frac{1}{2}(\ddot{\mathbf{d}}_n + \ddot{\mathbf{d}}_{n+1})$  die mittlere Beschleunigung im Zeitintervall  $\Delta t$  dar. Für diese Wahl der Parameter ( $\gamma = 0.5$ ,  $\beta = 0.25$ ) ist das Newmark Verfahren unbedingt stabil (siehe [66]) und stellt gleichzeitig die Energieerhaltung sicher. Für andere Werte von  $\beta$  ergeben sich Stabilitätsbedingungen für die Zeitschrittgröße.

In der vorliegenden Arbeit werden, wo nicht anders angegeben, die Werte  $\gamma = 0.5$  und  $\beta = 0.25$  verwendet. Wird numerische Dämpfung über das Newmark Verfahren eingebracht, geschieht dies mit den Werten  $\gamma = 0.9$  und  $\beta = 0.49$ .

## 3 Wahl von Benchmark Problemen

Nachfolgend werden in diesem Abschnitt einige lineare und nichtlineare Aufgaben vorgestellt, die als Benchmark Probleme für die Verwendung der FE Methode dienen sollen. Diese Probleme sollen im darauffolgenden Kapitel, das von der numerischen Lösung linearer Gleichungssysteme handelt, zu numerischen Vergleichszwecken herangezogen werden. Darum wird vor allem darauf geachtet, die diesbezüglichen Eigenschaften der mechanischen Probleme herauszustellen, wie z.B. die Kondition und die Struktur der Steifigkeitsmatrizen. Um aus den Effizienzvergleichen, die mit diesen Benchmark Problemen durchgeführt werden sollen, möglichst allgemeine Schlüsse ziehen zu können, müssen sie eine größtmögliche Vielfalt aufweisen. Diesem Anspruch wird durch die Auswahl der Probleme von linearen 3D Kontinuumsproblemen bis zu stark nichtlinearen Durchschlagproblemen dünner Zylinderschalen Rechnung getragen.

Wenn nicht anders angegeben, werden die Beispiele an späterer Stelle mit der hier vorgestellten Diskretisierung und Inkrementierung, d.h. gleicher Steuerung untersucht.

### 3.1 Zahnkrone

Beim ersten Problem geht es um die Berechnung einer Zahnkrone, die mit einem Stift im natürlichen Zahnstumpf befestigt ist, der seinerseits wiederum im Kiefer verankert ist. Zwischen dem Zahnstumpf und dem Kiefer befindet sich eine weiche Membran, das sogenannte Parodontium. In Abbildung 3.1 ist links das reale System und rechts die Finite Element Diskretisierung dargestellt. Diese konnte dankenswerterweise von [138] übernommen werden.

Untersuchungen an 2D Modellen in [138] ergaben im Vergleich zu dem hier betrachteten 3D Modell eine gleichartige Spannungsverteilung im Zahnstumpf. Jedoch wurden erwartungsgemäß signifikante Unterschiede in den von Mises Vergleichsspannungen zwischen den beiden Modellen festgestellt. Aus Gründen der Hardwareverfügbarkeit und Rechenzeiten wurden die Parameterstudien in [138] allerdings nur an den 2D Modellen durchgeführt.

Das im Rahmen der vorliegenden Arbeit nur für Rechenzeitvergleiche genutzte 3D-Modell ist mit 91 678 linearen 10-Knoten Tetraederelementen und 127 852 Knoten diskretisiert. Dies ergibt ein lineares Gleichungssystem mit 380 629 Unbekannten.

### 3.2 Flache Zylinderschale

Bei diesem Beispiel mit starker geometrischer Nichtlinearität handelt es sich um eine flache Zylinderschale, die durch eine mittig angreifende Einzellast belastet wird (siehe Abbildung 3.2). Zur Berechnung werden 4-knotige Schalenelemente mit bilinearen Ansatzfunktionen eingesetzt, siehe [46]. In Abbildung 3.3 sind die Verformungsfiguren wiedergegeben, dabei sind sowohl die Koordinaten als auch die Verschiebungen in vertikaler Richtung mit dem Faktor drei überhöht dargestellt.

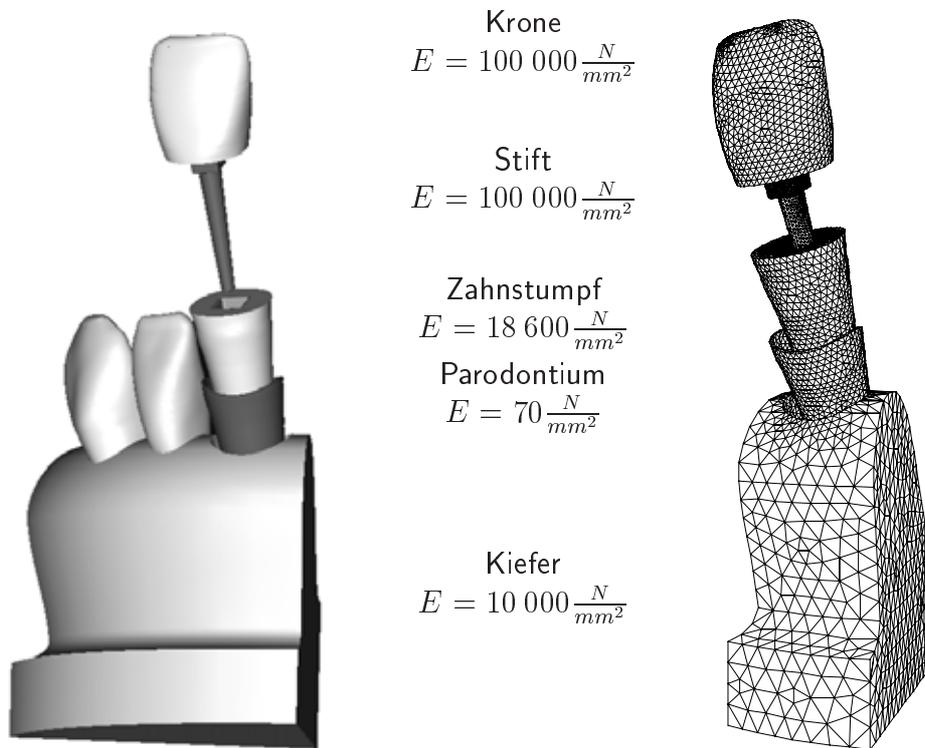


Abbildung 3.1: Zahnkrone – Geometrie und Materialdaten

Die Struktur weist ein sogenanntes *snap-through* und *snap-back* Verhalten auf (siehe Abbildung 3.4 links, in der die Last über der Verschiebung des Kraftangriffspunktes aufgetragen ist). Eine Untersuchung kann nur mit Hilfe des dem Bogenlängenverfahrens vorgenommen werden. Nach Erreichen einer Last von ca.  $0.6\text{ kN}$  schlägt die Struktur durch (*snap-through*), die Tangentensteifigkeitsmatrix wird (fast) singular, im instabilen Bereich ergeben sich indefinite Matrizen. In diesem Bereich verringern sich die Verschiebungen wieder; dieses Verhalten wird als *snap-back* bezeichnet. Zur Berechnung des gesamten Last-Verformungspfades dieses Systems wird der Einsatz des Bogenlängenverfahrens notwendig (siehe Abschnitt 2.3.2), da hier sowohl die ausschließliche Last- als auch die Ver-

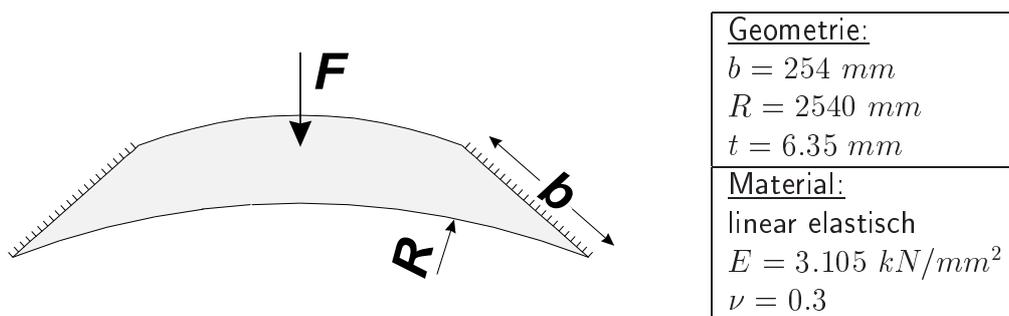


Abbildung 3.2: Dünne Zylinderschale – Geometrie und Materialdaten

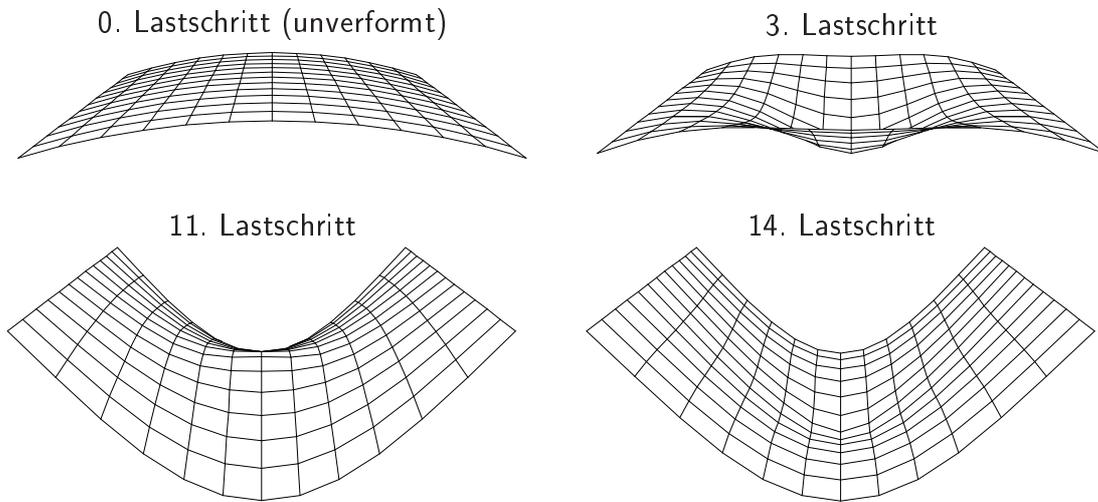


Abbildung 3.3: Flache Zylinderschale – Verformungsfiguren

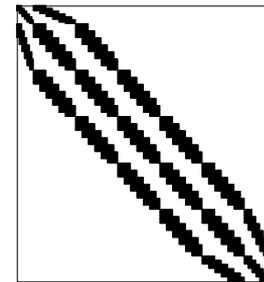
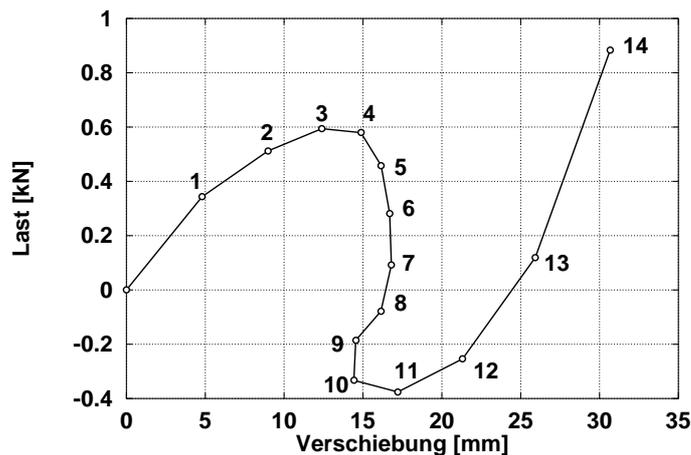


Abbildung 3.4: Flache Zylinderschale – Last-Verformungskurve und Struktur der Steifigkeitsmatrix, Lastschritte markiert

schiebungssteuerung versagt.

Aus Symmetriegründen wird nicht das gesamte System diskretisiert, sondern nur ein Viertel, unter Formulierung entsprechender Symmetriestandbedingungen. Dabei wird allerdings das Systemverhalten eingeschränkt und z.B. eine Verzweigung in unsymmetrische Formen damit ausgeschlossen. Für die Viertelstruktur werden 4 900 Knoten und 3 600 Elemente verwendet, was zu 24 011 Gleichungen führt. Durch die regelmäßige Struktur des Netzes ergibt sich auch eine regelmäßige Struktur der Steifigkeitsmatrix (siehe Abbildung 3.4 rechts). Jede Zeile der Matrix enthält im Durchschnitt 22 Einträge.

In Tabelle 3.1 sind die Konditionszahlen für die auskonvergierten Gleichgewichtslagen angegeben. Bei diesem Beispiel ist im Bereich der indefiniten Tangentensteifigkeitsmatrizen (Gleichgewichtslagen 4-11), der betragskleinste gleichzeitig der kleinste Eigenwert.

Lastschritt	kleinster Eigenwert	größter Eigenwert	Konditionszahl
0	$0.3286 \cdot 10^{-4}$	$0.2894 \cdot 10^3$	$0.8805 \cdot 10^7$
1	$0.1691 \cdot 10^{-4}$	$0.2883 \cdot 10^3$	$0.1704 \cdot 10^8$
2	$0.7858 \cdot 10^{-5}$	$0.2881 \cdot 10^3$	$0.3666 \cdot 10^8$
3	$0.2213 \cdot 10^{-5}$	$0.2879 \cdot 10^3$	$0.1301 \cdot 10^9$
4	$-0.2916 \cdot 10^{-5}$	$0.2878 \cdot 10^3$	$0.9871 \cdot 10^8$
5	$-0.6436 \cdot 10^{-5}$	$0.2878 \cdot 10^3$	$0.4472 \cdot 10^8$
6	$-0.7816 \cdot 10^{-5}$	$0.2878 \cdot 10^3$	$0.3682 \cdot 10^8$
7	$-0.8370 \cdot 10^{-5}$	$0.2878 \cdot 10^3$	$0.3439 \cdot 10^8$
8	$-0.8728 \cdot 10^{-5}$	$0.2877 \cdot 10^3$	$0.3297 \cdot 10^8$
9	$-0.8504 \cdot 10^{-5}$	$0.2875 \cdot 10^3$	$0.3381 \cdot 10^8$
10	$-0.8537 \cdot 10^{-5}$	$0.2873 \cdot 10^3$	$0.3365 \cdot 10^8$
11	$-0.2417 \cdot 10^{-5}$	$0.2875 \cdot 10^3$	$0.1189 \cdot 10^9$
12	$0.6839 \cdot 10^{-5}$	$0.2878 \cdot 10^3$	$0.4209 \cdot 10^8$
13	$0.1900 \cdot 10^{-4}$	$0.2882 \cdot 10^3$	$0.1516 \cdot 10^8$

Tabelle 3.1: Flache Zylinderschale – Konditionszahlen

Deshalb wurde das Vorzeichen hier mit angegeben. Man kann deutlich ein Ansteigen der Konditionszahl an den fast singulären Stellen, d.h. den Gleichgewichtslagen 3 bzw. 11 erkennen. Während der größte Eigenwert im Verlauf der nichtlinearen Berechnung nahezu konstant bleibt, sinkt der kleinste Eigenwert im Bereich der singulären Stellen auf einen Wert nahe Null und im instabilen Bereich unter Null ab. Der betragskleinste Eigenwert ist aber – im Vergleich mit anderen Beispielen – nicht nur nahe der singulären Stellen sehr klein sondern an allen Punkten der Last-Verformungskurve. Er erhöht sich erst beim im Bereich größerer Verschiebungen, wenn sich die Struktur nach dem Durchschlagen wieder versteift.

### 3.3 Dünnwandiger Torus

Bei diesem Beispiel handelt es sich um einen dünnwandigen Torus, der durch vier Einzel-lasten belastet wird, siehe Abbildung 3.5.

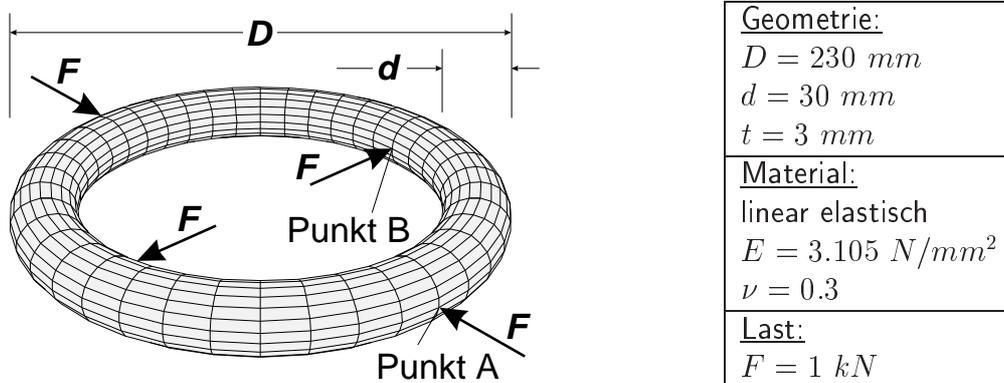


Abbildung 3.5: Torus – Geometrie und Materialdaten

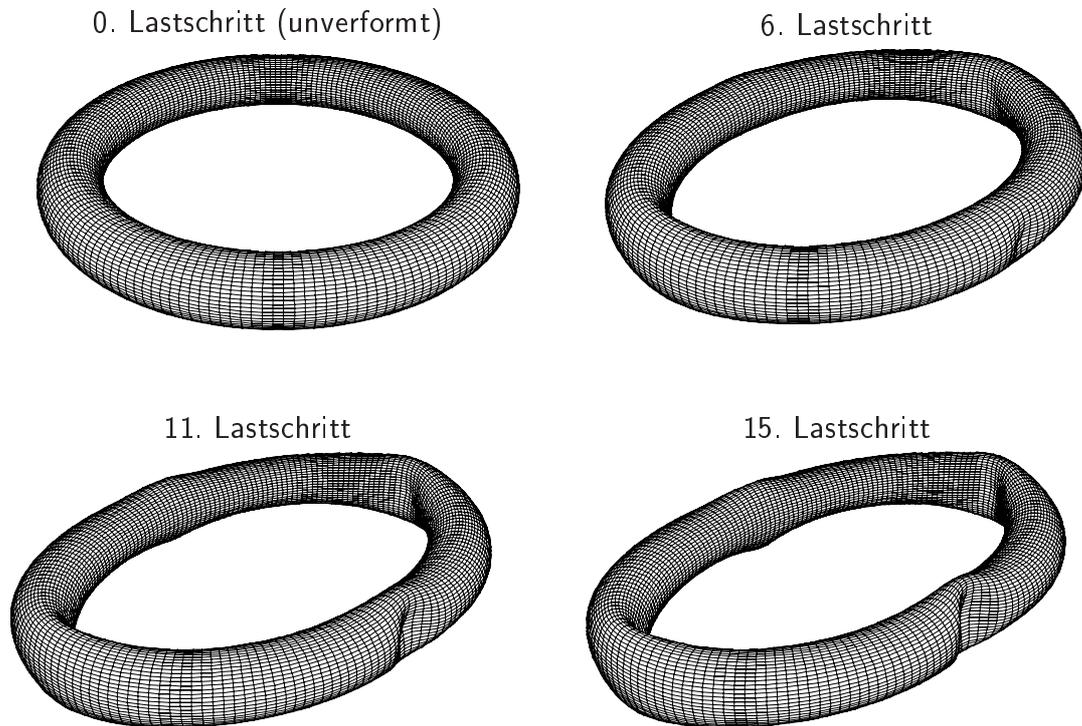


Abbildung 3.6: Dünnwandiger Torus – Verformungsfiguren

Hier ist eine sehr feine Diskretisierung erforderlich, um Knicke abzubilden, die während der Lastaufbringung entstehen (siehe Abbildung 3.6). Für die Berechnungen wird zum einen ein regelmäßiges FE-Netz mit 10 800 Elementen und 11 041 Knoten verwendet, was zu 53 400 Gleichungen führt. Zum anderen wird mit einem aus einer adaptiven Berechnung stammenden Netz gerechnet. Bei 11 552 Knoten und 11 552 Elementen ergeben sich hierfür 68 679 Gleichungen.

Die Last-Verformungskurve (siehe Abbildung 3.7 links) endet mit einer horizontalen Tangente; hier führen kleine Laststeigerungen zu großen Verschiebungsänderungen. Nach dem Bilden von Knicken reagiert der Torus auf geringe Laststeigerung mit großen Verschiebungen. Die mit den beiden unterschiedlichen Diskretisierungen berechneten Last-Verformungskurven unterscheiden sich kaum; das Netz ist auskonvergiert.

Die Struktur der Steifigkeitsmatrix für die gleichmäßige Diskretisierung ist sehr regelmäßig, jedoch existieren Matrixeinträge, die sehr weit von der Diagonale entfernt sind. Dies ist eine Folge der Torusgeometrie. Es ist unvermeidbar, daß Knoten mit hoher Nummer mit Knoten niedriger Nummer in einem Element zusammen auftreten. Durchschnittlich befinden sich in jeder Matrixzeile 27 Einträge, da im Gegensatz zum Beispiel in Abschnitt 3.2 hier sogenannte “Solid Shell” Elemente aus [60] verwendet werden.

In Tabelle 3.2 sind die Konditionszahlen für den dünnwandigen Torus sowohl für das regelmäßige, als auch für das adaptive Netz angegeben. Bei beiden verwendeten Netzen sind die Konditionszahlen von gleicher Größenordnung. Diejenigen des adaptiven Netzes sind

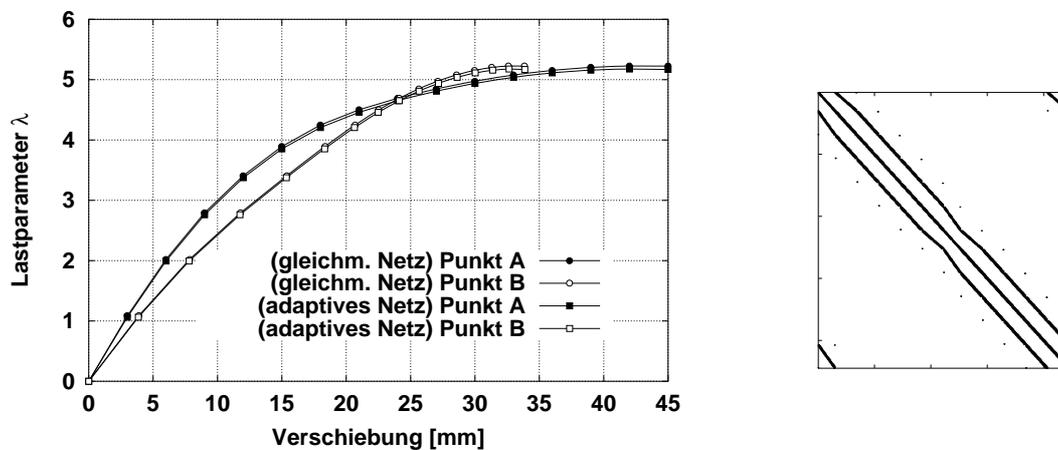


Abbildung 3.7: Torus – Last-Verformungskurve und Matrixstruktur

regelmäßiges Netz				adaptives Netz			
Lastschritt	kleinster Eigenwert	größter Eigenwert	Konditionszahl	Lastschritt	kleinster Eigenwert	größter Eigenwert	Konditionszahl
0	$0.8215 \cdot 10^{-3}$	$0.6161 \cdot 10^2$	$0.2556 \cdot 10^6$	0	$0.5939 \cdot 10^{-3}$	$0.1368 \cdot 10^3$	$0.7552 \cdot 10^6$
1	$0.1363 \cdot 10^{-3}$	$0.6125 \cdot 10^2$	$0.4493 \cdot 10^6$	1	$0.1032 \cdot 10^{-3}$	$0.1318 \cdot 10^3$	$0.1277 \cdot 10^7$
2	$0.1274 \cdot 10^{-3}$	$0.6145 \cdot 10^2$	$0.4821 \cdot 10^6$	2	$0.9662 \cdot 10^{-4}$	$0.1322 \cdot 10^3$	$0.1368 \cdot 10^7$
3	$0.1150 \cdot 10^{-3}$	$0.6170 \cdot 10^2$	$0.5363 \cdot 10^6$	3	$0.8770 \cdot 10^{-4}$	$0.1325 \cdot 10^3$	$0.1511 \cdot 10^7$
4	$0.1026 \cdot 10^{-3}$	$0.6195 \cdot 10^2$	$0.6037 \cdot 10^6$	4	$0.7918 \cdot 10^{-4}$	$0.1328 \cdot 10^3$	$0.1677 \cdot 10^7$
5	$0.9208 \cdot 10^{-4}$	$0.6217 \cdot 10^2$	$0.6751 \cdot 10^6$	5	$0.7189 \cdot 10^{-4}$	$0.1331 \cdot 10^3$	$0.1852 \cdot 10^7$
6	$0.8211 \cdot 10^{-4}$	$0.6236 \cdot 10^2$	$0.7595 \cdot 10^6$	6	$0.4049 \cdot 10^{-3}$	$0.1333 \cdot 10^3$	$0.1198 \cdot 10^7$
7	$0.7148 \cdot 10^{-4}$	$0.6252 \cdot 10^2$	$0.8746 \cdot 10^6$	7	$0.3444 \cdot 10^{-3}$	$0.1335 \cdot 10^3$	$0.1374 \cdot 10^7$
8	$0.4449 \cdot 10^{-3}$	$0.6264 \cdot 10^2$	$0.5864 \cdot 10^6$	8	$0.2899 \cdot 10^{-3}$	$0.1337 \cdot 10^3$	$0.1608 \cdot 10^7$
9	$0.3779 \cdot 10^{-3}$	$0.6271 \cdot 10^2$	$0.7069 \cdot 10^6$	9	$0.2448 \cdot 10^{-3}$	$0.1339 \cdot 10^3$	$0.1912 \cdot 10^7$
10	$0.3202 \cdot 10^{-3}$	$0.6274 \cdot 10^2$	$0.8743 \cdot 10^6$	10	$0.2069 \cdot 10^{-3}$	$0.1341 \cdot 10^3$	$0.2322 \cdot 10^7$
11	$0.2699 \cdot 10^{-3}$	$0.6275 \cdot 10^2$	$0.1122 \cdot 10^7$	11	$0.1744 \cdot 10^{-3}$	$0.1342 \cdot 10^3$	$0.2902 \cdot 10^7$
12	$0.2262 \cdot 10^{-3}$	$0.6271 \cdot 10^2$	$0.1521 \cdot 10^7$	12	$0.1464 \cdot 10^{-3}$	$0.1344 \cdot 10^3$	$0.3771 \cdot 10^7$
13	$0.1884 \cdot 10^{-3}$	$0.6267 \cdot 10^2$	$0.2251 \cdot 10^7$	13	$0.1224 \cdot 10^{-3}$	$0.1345 \cdot 10^3$	$0.5178 \cdot 10^7$
14	$0.1562 \cdot 10^{-3}$	$0.6260 \cdot 10^2$	$0.3942 \cdot 10^7$	14	$0.1021 \cdot 10^{-3}$	$0.1347 \cdot 10^3$	$0.7737 \cdot 10^7$

Tabelle 3.2: Dünnwandiger Torus – Konditionszahlen

jedoch stets größer, da dieses Netz zum Teil sehr kleine, aber auch größere Elemente verglichen mit dem gleichmäßigen Netz enthält.

### 3.4 Rohrkreuz

Bei diesem Schalenproblem handelt es sich um ein Rohrkreuz, das an seiner Ober- und Unterseite fest eingespannt ist und an den beiden gegenüberliegenden Seiten durch Schubkräfte belastet wird (siehe Abbildung 3.8). An der Ober- und Unterseite ist das Rohrkreuz unverschieblich gelagert. Die Schalendicke der Zylinderflächen beträgt  $3 \text{ mm}$ , die Anschlüsse sind  $6 \text{ mm}$  dick. Zur Berechnung werden 4-knotige Schalenelemente aus [46]

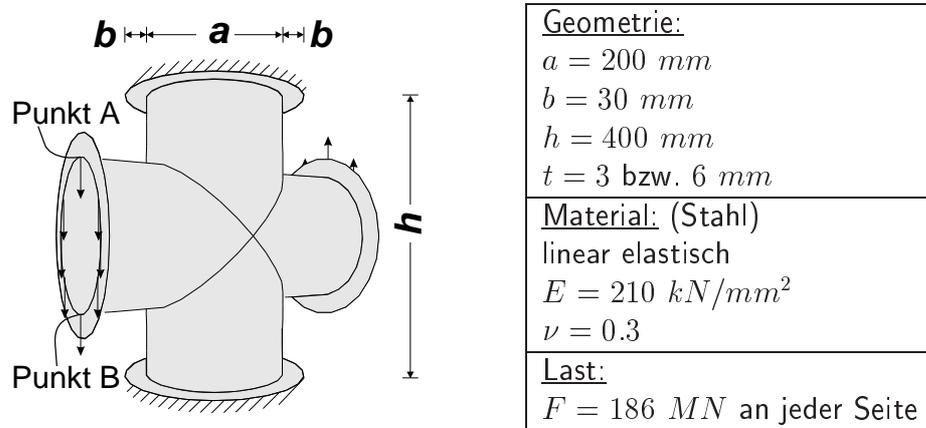


Abbildung 3.8: Rohrkreuz – Geometrie und Materialdaten

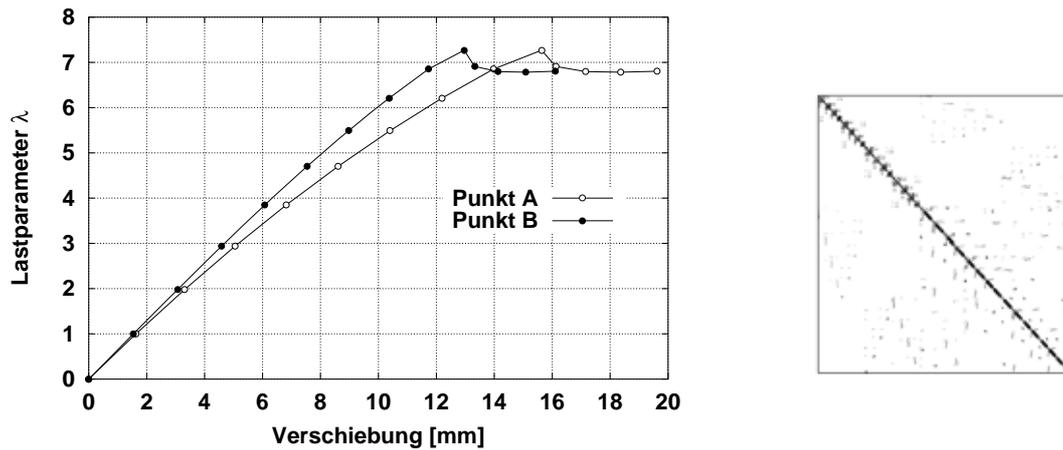


Abbildung 3.9: Rohrkreuz – Last-Verformungskurve und Matrixstruktur

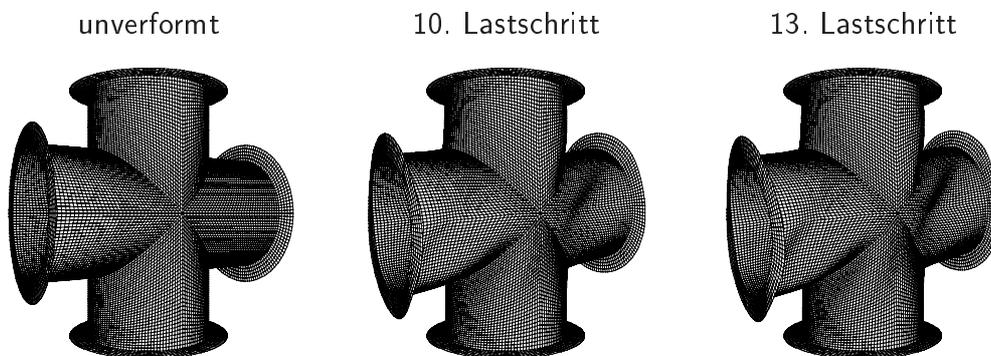


Abbildung 3.10: Rohrkreuz – Verformungsfiguren

Lastschritt	kleinster Eigenwert	größter Eigenwert	Konditionszahl
0	$0.2218 \cdot 10^5$	$0.1849 \cdot 10^{11}$	$0.9138 \cdot 10^6$
1	$0.2189 \cdot 10^5$	$0.1862 \cdot 10^{11}$	$0.9277 \cdot 10^6$
2	$0.2110 \cdot 10^5$	$0.1861 \cdot 10^{11}$	$0.8832 \cdot 10^6$
3	$0.1998 \cdot 10^5$	$0.1862 \cdot 10^{11}$	$0.9336 \cdot 10^6$
4	$0.1870 \cdot 10^5$	$0.1863 \cdot 10^{11}$	$0.9985 \cdot 10^6$
5	$0.1744 \cdot 10^5$	$0.1866 \cdot 10^{11}$	$0.1072 \cdot 10^7$
6	$0.1624 \cdot 10^5$	$0.1868 \cdot 10^{11}$	$0.1152 \cdot 10^7$
7	$0.1490 \cdot 10^5$	$0.1871 \cdot 10^{11}$	$0.1258 \cdot 10^7$
8	$0.1265 \cdot 10^5$	$0.1874 \cdot 10^{11}$	$0.1484 \cdot 10^7$
9	$-0.1375 \cdot 10^4$	$0.1877 \cdot 10^{11}$	$0.1368 \cdot 10^8$
10	$-0.3397 \cdot 10^3$	$0.1881 \cdot 10^{11}$	$0.5550 \cdot 10^8$
11	$-0.7933 \cdot 10^3$	$0.1883 \cdot 10^{11}$	$0.2379 \cdot 10^8$
12	$0.3333 \cdot 10^3$	$0.1884 \cdot 10^{11}$	$0.1088 \cdot 10^9$
13	$0.7566 \cdot 10^3$	$0.1884 \cdot 10^{11}$	$0.3336 \cdot 10^8$
14	$0.8702 \cdot 10^3$	$0.1882 \cdot 10^{11}$	$0.2948 \cdot 10^8$

Tabelle 3.3: Rohrkreuz – Konditionszahlen

verwendet.

Das Netz wurde mit einem kommerziellen Netzgenerierer erzeugt; als Folge der automatischen Generierung ist die Knotennumerierung nicht optimal. Die Matrixstruktur ist daher nicht ausgeprägt bandförmig (siehe Abbildung 3.9). Pro Zeile enthält die Steifigkeitsmatrix durchschnittlich 22 Einträge. Die zur Berechnung verwendete Diskretisierung erfolgt mit 22 678 Knoten und 22 400 Elementen. Dies führt zu 112 550 Gleichungen.

In Abbildung 3.10 kann man im 13. Lastschritt das Einklappen der Flanken der horizontalen Anschlüsse der Schale erkennen. Hier verliert das Rohrkreuz infolge dieser Beulen schlagartig an Tragfähigkeit. Dies ist auch in der Last-Verformungskurve erkennbar (siehe Abbildung 3.9). Anschließend wird das Rohrkreuz sehr weich. Ohne nennenswerte Steigerung der Last treten große Verformungen auf.

Bei diesem Beispiel sind sowohl die betragsgrößten als auch die betragskleinsten Eigenwerte sehr hoch (verglichen mit den anderen Beispielen), siehe Tabelle 3.3. Der Grund hierfür liegt in der beiderseitigen festen Einspannung. Die sich daraus ergebenden Konditionszahlen liegen aber in einer ähnlichen Größenordnung wie beim Beispiel in Abschnitt 3.2. Beim Erreichen der maximalen Traglast (9. Lastschritt) werden die Steifigkeitsmatrizen indefinit, das Strukturverhalten ist instabil. Auch hier ist der betragskleinste Eigenwert negativ.

### 3.5 Gummiblock

Beim letzten Beispiel handelt es sich um ein 3D-Kontinuumsproblem aus Gummimaterial, für das ein Mooney-Rivlin Materialgesetz angenommen wird. Ein Quader aus Gummi wird zwischen zwei steifen Platten zusammengedrückt. Die verwendeten Mooney-Rivlin Parameter und die Geometriedaten sind in Abbildung 3.11 angegeben, die Verformungsfiguren

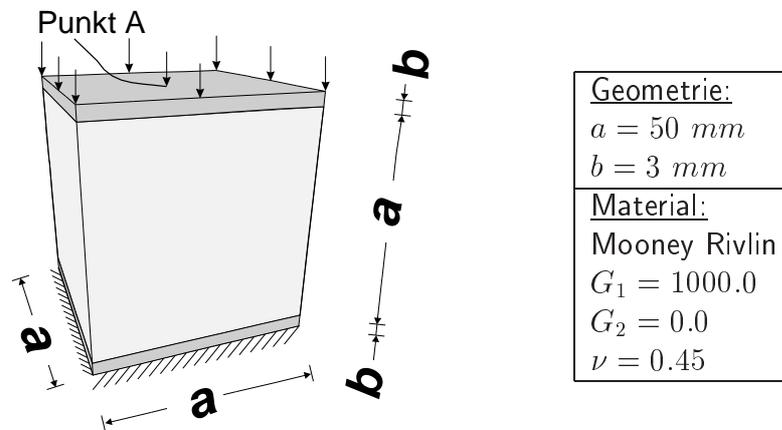


Abbildung 3.11: Gummiblock – Geometrie und Materialdaten

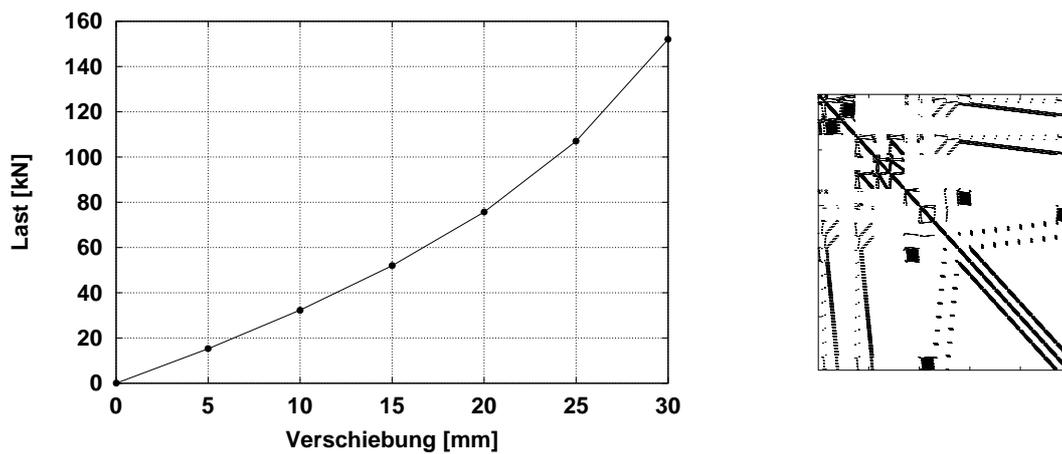


Abbildung 3.12: Gummiblock – Last-Verformungskurve und Matrixstruktur

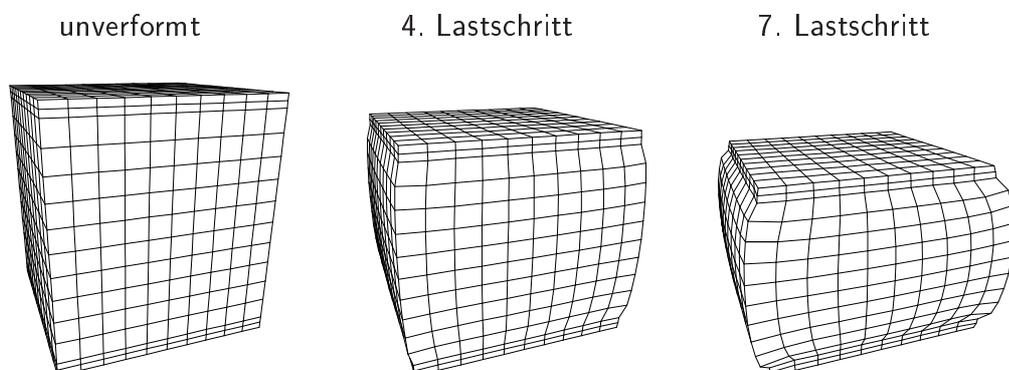


Abbildung 3.13: Gummiblock – Verformungsfiguren

in Abbildung 3.13 dargestellt.

Bei wachsender Verformung versteift der Gummiblock. Die Last-Verformungskurve ist überlinear (siehe Abbildung 3.12 links, in der die Last über der Verschiebung der Oberseite aufgetragen ist).

Für die Berechnungen wird ein regelmäßiges Netz mit 21 632 Knoten und 19 375 Elementen verwendet, was zu linearen Gleichungssystemen mit 62 192 Unbekannten führt. Auch dieses Netz wurde automatisch erzeugt. Die Steifigkeitsmatrix ist trotz des regelmäßigen FE-Netzes sehr unstrukturiert (siehe Abbildung 3.12 rechts). Bei diesem Volumenproblem ist sie aber deutlich dichter besetzt als bei den vorangegangenen Schalenproblemen. Im Durchschnitt enthält hier jede Matrixzeile 34 Einträge. Dies resultiert aus der Verknüpfung der Elemente in allen drei Dimensionen. Jeder Knoten im Innern des Blockes hat 26 Nachbarknoten.

## 4 Direkte und iterative Verfahren zur linearen Lösung dünn besetzter Gleichungssysteme

Bei vielen numerischen Simulationen naturwissenschaftlicher und technischer Probleme, die durch partielle Randwert- oder Anfangsrandwertprobleme beschrieben werden, ist der rechentechnisch aufwendigste Schritt das Lösen von linearen Gleichungssystemen. Ein Beispiel hierfür sind die in Abschnitt 2 eingeführten strukturmechanischen Probleme. Ähnlich verhält es sich bei Problemen aus der Strömungsmechanik oder bei anderen Disziplinen, wie z.B. in der Quantenphysik oder in der Chemie. Unabhängig vom Anwendungsbereich besitzen die auftretenden linearen Gleichungssysteme gemeinsame Eigenschaften. Die auftretende Koeffizientenmatrix ist oft *symmetrisch*, *dünn besetzt* und *positiv definit*, wobei gelegentlich einzelne dieser Eigenschaften verlorengehen können. Beispielsweise werden die hier betrachteten Koeffizientenmatrizen aus FE-Simulationen nichtlinearer Probleme der Strukturmechanik bei instabilem Strukturverhalten indefinit, unter Einbeziehung z.B. von Reibung oder speziellen Materialmodellen können sie auch unsymmetrisch werden. Letzteres soll aber im Rahmen der vorliegenden Untersuchung nicht betrachtet werden.

Um lineare Gleichungssysteme der Art

$$\mathbf{Ax} = \mathbf{b} \tag{4.1}$$

mit der Koeffizientenmatrix  $\mathbf{A}$ , der rechten Seite  $\mathbf{b}$  und dem unbekanntem Lösungsvektor  $\mathbf{x}$  zu lösen, existieren zwei grundsätzlich verschiedene Lösungsstrategien: die *direkte* und die *iterative* Lösung. Die direkte Lösung wird vielfach von Ingenieuren bevorzugt und dabei häufig als exakte Lösung<sup>1</sup> bezeichnet. Grundlage der direkten Löser ist die Gauß-Elimination. Im Unterschied dazu wird bei der iterativen Lösung versucht, eine Näherungslösung zu verbessern, bis sie einer vorgegebenen Genauigkeitsanforderung genügt. Ein Nachteil der iterativen Strategie im Gegensatz zur direkten ist, daß a priori nicht bekannt ist, ob der Algorithmus konvergiert, und wieviel Rechenzeit im Falle der Konvergenz zur Lösung benötigt wird. Der Vorteil im Vergleich mit dem direkten Löser hingegen ist, daß sehr wenig Speicherplatz erforderlich ist. Ein weiterer Grund für die wachsende Popularität der iterativen Verfahren in jüngerer Zeit ist ihre hervorragende Eignung zur Berechnung der Lösung von linearen Gleichungssystemen auf Parallelrechnern.

Da im Rahmen der vorliegenden Abhandlung die Parallelisierung eines Finiten Elemente Programmes angestrebt wird, muß zunächst untersucht werden, inwieweit diese Parallelisierung unter Beschränkung auf iterative Löser möglich ist. Ein ausschließlicher Einsatz iterativer Verfahren ist dann möglich, wenn die angestrebten Problemklassen damit robust gelöst werden können. Es soll jedoch kein nennenswerter Verlust an Rechenzeit auftreten, der dann erst durch den Einsatz mehrerer Prozessoren kompensiert werden müßte. Ein aussagekräftiger Effizienzvergleich zwischen iterativen und direkten Verfahren setzt allerdings voraus, daß in beiden Fällen optimierte Algorithmen eingesetzt werden.

---

<sup>1</sup>Im Sinne der Maschinengenauigkeit.

Dazu werden in den nächsten Abschnitten zunächst bekannte direkte und iterative Lösungsstrategien beschrieben und die Algorithmen in das Finite Elemente Programm FEAP [131] integriert. Zur iterativen Lösung notwendige Vorkonditionierungsstrategien – bekannte und auch eigene Varianten – werden vorgestellt. Es wird versucht, deren Eigenschaften anschließend anhand der im Abschnitt 3 definierten Benchmark Probleme zu vergleichen und zu bewerten. Für den Anwender des Programmes auf sequentiellen Rechnern sollen letztlich einige Richtlinien zur Wahl eines geeigneten Löser für ein Problem formuliert werden.

## 4.1 Direkte Gleichungslösung

Formal kann das lineare Gleichungssystem (4.1) durch linksseitiges Multiplizieren mit der Kehrmatrix  $A^{-1}$  gelöst werden:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}. \quad (4.2)$$

Die numerische Berechnung von  $A^{-1}$  ist jedoch sehr aufwendig, die Kehrmatrix ist außerdem im allgemeinen Falle dicht besetzt, sodaß zur direkten Lösung immer dem Gauß-Verfahren verwandte Algorithmen verwendet werden. Damit das Gleichungssystem unter Umständen für mehrere rechte Seiten gelöst werden kann, beruhen diese auf einer Faktorisierung der Koeffizientenmatrix in zwei Dreiecksmatrizen

$$\mathbf{A} = \hat{\mathbf{L}}\hat{\mathbf{U}} \quad (4.3)$$

und anschließendem sogenannten Rückwärts- und Vorwärtseinsetzen:

$$\mathbf{Ax} = \hat{\mathbf{L}}\underbrace{\hat{\mathbf{U}}\mathbf{x}}_{\mathbf{y}} = \mathbf{b}; \quad (4.4)$$

$$\hat{\mathbf{L}}\mathbf{y} = \mathbf{b} \quad \text{Vorwärtseinsetzen} \quad (4.5)$$

$$\hat{\mathbf{U}}\mathbf{x} = \mathbf{y} \quad \text{Rückwärtseinsetzen.} \quad (4.6)$$

Die Faktoren  $\hat{\mathbf{L}}$  und  $\hat{\mathbf{U}}$  besitzen Dreiecksgestalt.  $\hat{\mathbf{L}}$  ist eine untere Dreiecksmatrix, deren Diagonaleinträge sämtlich Eins sind.  $\hat{\mathbf{U}}$  ist eine obere Dreiecksmatrix. Durch die Diagonalgestalt von  $\hat{\mathbf{L}}$  und  $\hat{\mathbf{U}}$  sind die Gleichungssysteme (4.5) und (4.6) sehr einfach lösbar. Der hauptsächliche numerische Aufwand besteht in der Bestimmung der Faktoren  $\hat{\mathbf{L}}$  und  $\hat{\mathbf{U}}$ . Hierzu sind im Schrifttum unterschiedliche Strategien für Koeffizientenmatrizen unterschiedlicher Eigenschaften angegeben, siehe z.B. in [50, 53]. Mit direktem Bezug auf strukturmechanische Probleme wird dies ebenfalls in [24, 127] diskutiert. Einige wesentliche Aspekte werden in den folgenden Abschnitten nochmals kurz erläutert.

Für symmetrische, positiv definite Matrizen  $\mathbf{A}$  können mittels der Cholesky-Zerlegung symmetrische Faktoren

$$\mathbf{A} = \bar{\mathbf{L}}\bar{\mathbf{L}}^T \quad (4.7)$$

bestimmt werden. Damit wird der Speicherbedarf für die Faktoren halbiert. Bei dichtbesetzten Matrizen wird die Anzahl der erforderlichen Rechenoperationen ebenfalls halbiert. Zur Berechnung der Diagonalelemente von  $\bar{L}$  müssen aber Quadratwurzeln berechnet werden. Negative Diagonalelemente, wie sie bei der Faktorisierung indefiniter Matrizen auftreten, sind daher hier nicht zulässig. Für symmetrische, aber auch indefinite Matrizen, wie sie in der vorliegenden Arbeit betrachtet werden, wird deshalb zur direkten Lösung eine Faktorisierung der Form

$$A = LDL^T \quad (4.8)$$

gewählt, der benötigte Speicherplatz entspricht dem der Cholesky-Zerlegung (4.7). Durch Überwachen der Elemente der Diagonalmatrix  $D$  kann als Nebenprodukt die Anzahl der negativen Eigenwerte der Koeffizientenmatrix  $A$  festgestellt werden, vgl. hierzu den Satz von Sylvester z.B. in [53]. Dies ist eine bei nichtlinearen Finite Element Berechnungen wichtige Information, die zu Stabilitätsaussagen für die betrachtete mechanische Struktur verwendet werden kann. Die Diagonalelemente von  $L$  sind sämtlich Eins; es handelt sich um eine sogenannte untere Dreiecksmatrix.

Wichtig bezüglich Effizienz bei diesen Zerlegungen ist die Entstehung von sogenanntem *fill-in*, d.h. von Matrixelementen, die in der Koeffizientenmatrix  $A$  Null waren, die aber in  $L$  ungleich Null sind. Bei dichtbesetzten Matrizen spielt dies keine Rolle; werden bei den betrachteten sehr dünn besetzten ursprünglichen Matrizen jedoch ausschließlich Nichtnull-elemente gespeichert (siehe Anhang C), so entsteht durch den *fill-in* ein zusätzlicher Speicheraufwand, der erheblich größer sein kann als für die Matrix selbst. Durch eine geeignete Permutation von  $A$  kann allerdings der entstehende *fill-in* reduziert werden. Es wird dann ein permutiertes Gleichungssystem

$$PAP^T Px = Pb \quad (4.9)$$

gelöst, sodaß weniger *fill-in* entsteht. Dabei wird der Aspekt der numerischen Stabilität nicht berücksichtigt. Die Faktorisierung erfolgt also ohne numerische Pivotsuche, da dadurch die dünne Besetzungsstruktur verlorenginge und der Speicherbedarf und damit einhergehend die Rechenzeit anstiege.

Zusammenfassend läßt sich die direkte Gleichungslösung für dünnbesetzte Matrizen in die folgenden Schritte untergliedern:

1. Bestimmung einer *fill-in* minimierenden Permutation  $P$ .
2. Symbolische Faktorisierung, d.h. Ermittlung der Besetzungsstruktur des Faktors  $L$ .
3. Numerische Faktorisierung, d.h. Berechnung von  $L$  und  $D$ .
4. (Möglicherweise wiederholtes) Vorwärts- und Rückwärtseinsetzen.

Diese Punkte werden in den anschließenden Abschnitten noch ausführlicher behandelt, zuvor werden jedoch noch einige dazu notwendige Begriffe aus der Graphentheorie eingeführt.

### 4.1.1 Einige Begriffe aus der Graphentheorie

Ein Graph  $G(V, E)$  besteht aus einer Menge von Knoten  $V$  (*vertex*), die durch eine Menge von Kanten  $E$  (*edges*) miteinander verbunden sind, siehe Abbildung 4.1 (rechts).

Zwei Knoten  $v$  und  $w$  sind benachbart, wenn die Kante  $(v, w)$  in der Menge  $E$  der Kanten enthalten ist. Der Knotengrad  $\deg(v)$  ist die Anzahl der Nachbarn eines Knotens, was der Anzahl der vom Knoten  $v$  ausgehenden Kanten entspricht. Der Knotengrad des Knoten 5 in Abbildung 4.1 (rechts) ist also 5, die benachbarten Knoten sind 1, 3, 7, 9 und 11. Mit  $|V|$  wird die Anzahl der Knoten des Graphen  $G$  bezeichnet, in Abbildung 4.1 (rechts) 12. Ein Untergraph  $G'(V', E')$  ist ein Graph mit  $V' \subset V$  und  $E' \subset E$ .

Für eine ausführlichere Darstellung sei an dieser Stelle auf [49] verwiesen.

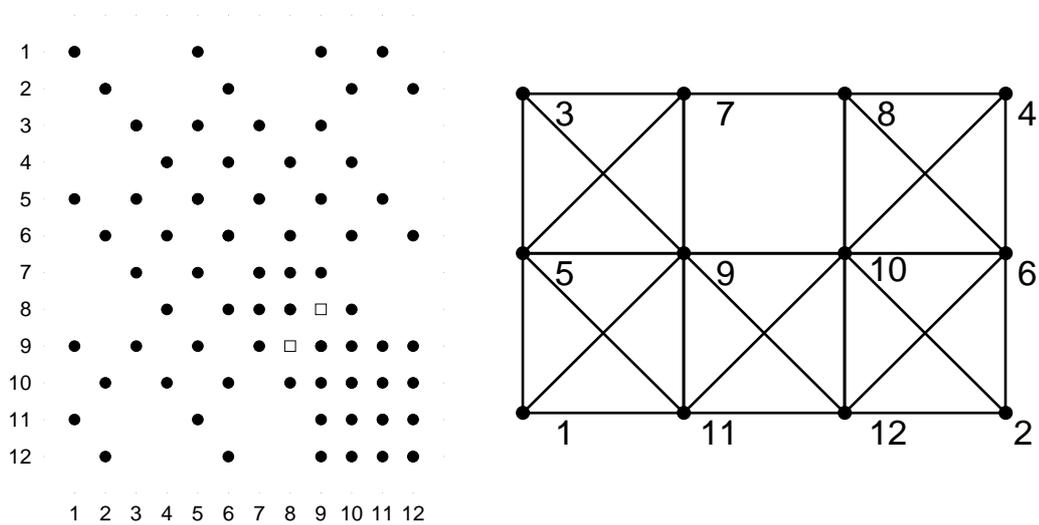


Abbildung 4.1: Beispielmatrix und resultierender Matrixgraph

Der Graph einer symmetrischen Matrix wird so aufgebaut, daß jede der  $n$  Zeilen der Matrix einem Knoten des Graphen entspricht. Eine Kante zwischen den Knoten  $v$  und  $w$  entsteht genau dann, wenn in der Matrix der Eintrag  $a_{vw} \neq 0$  ist. Der Graph repräsentiert also die Besetzungsstruktur der Matrix. In Abbildung 4.1 (links) ist die Besetzungsstruktur einer Matrix (die besetzten Stellen sind durch Punkte gekennzeichnet – die Bedeutung der beiden Quadrate wird an späterer Stelle erläutert) und der dazugehörige Graph (rechts) abgebildet.

### 4.1.2 Permutationsstrategien zur *fill-in* Minimierung und symbolische Faktorisierung

Die ersten beiden Schritte zur direkten Lösung linearer Gleichungssysteme sollen in diesem Abschnitt gemeinsam diskutiert werden, obwohl sie tatsächlich voneinander unabhängig berechnet werden können. Der Grund hierfür liegt darin, daß die Kenntnis über die symbolische Faktorisierung – also der Bestimmung des Auftretens von *fill-in* – Heuristiken zur dessen Vermeidung nahelegt.

Zunächst werden die Knoten des Matrixgraphen mit der ihnen entsprechenden Zeilennummer der Koeffizientenmatrix numeriert, und somit die Eliminationsreihenfolge festgelegt. Die symbolische Faktorisierung kann am Matrixgraphen nun folgendermaßen veranschaulicht werden: Für den Knoten mit der kleinsten Nummer werden zuerst alle Nachbarknoten miteinander durch Kanten verbunden, sofern diese Kanten nicht bereits existierten. Anschließend streicht man den Knoten aus dem Graphen. Jede hinzugefügte Kante entspricht einem zusätzlich entstehenden Matrixeintrag im Faktor, also *fill-in*.

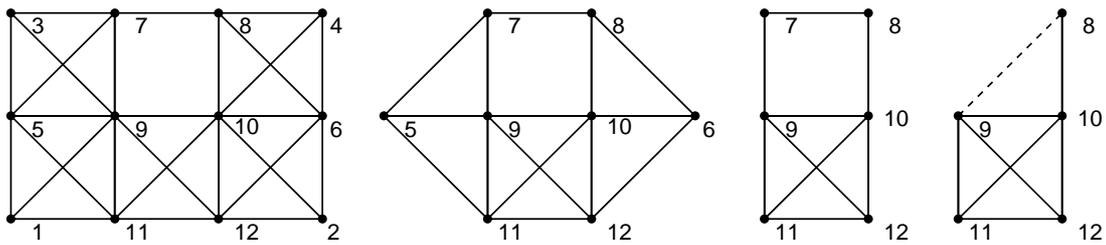


Abbildung 4.2: Veranschaulichung des Eliminationsvorganges am Graphen

In Abbildung 4.2 ist dieses Vorgehen für die Matrix aus Abbildung 4.1 dargestellt. In mehreren Schritten werden die Knoten des Graphen aus Abbildung 4.1 entfernt und dabei vorher die Nachbarknoten miteinander verbunden, was nur in einem Schritt das Einfügen einer zusätzlichen Kante zwischen den Knoten 8 und 9 erfordert (gestrichelt eingezeichnet). Beim Faktorisieren entsteht also nur ein zusätzliches Matrixelement. Dieses ist in der Matrix in Abbildung 4.1 (links) durch ein  $\square$  gekennzeichnet.

### *Minimum degree* Permutation

Die *minimum degree* Heuristik (siehe z.B. [51]) bestimmt die Eliminationsreihenfolge – und damit die Permutation der Matrix – so, daß als nächstes immer der Knoten kleinsten Grades eliminiert wird. Verglichen mit Knoten hohen Grades, bei denen mit größerer Wahrscheinlichkeit Nachbarknoten existieren, die noch nicht durch Kanten miteinander verbunden sind, wird dadurch die Wahrscheinlichkeit gesenkt, daß eine zusätzliche Kante eingefügt werden muß. Existieren mehrere Knoten gleichen Grades, so sind sogenannte *tie-breaking* Strategien notwendig, die dann zu Permutationen unterschiedlicher Qualität führen können. Das in Abbildung 4.2 gewählte Beispiel wurde mit dieser Heuristik numeriert und so der *fill-in* minimiert. Für weitergehende Aspekte und zur Implementierung sei auf [51] verwiesen.

### *Nested dissection* Permutation

Bei der *nested dissection* Strategie (siehe z.B. [48] oder [49]) wird der Graph rekursiv so in gleich große Teile geteilt, daß sich möglichst wenig Knoten zwischen den Teilgebieten befinden. Diese Knoten bezeichnet man als Separatoren. Es werden zuerst die Teilgebiete numeriert und anschließend die Separatoren, wodurch eine rekursive Block-Pfeil Struk-

tur der Matrix entsteht. Der *fill-in* minimierende Effekt beruht auf den nun voneinander unabhängigen Teilgebieten.

### 4.1.3 Numerische Faktorisierung

Der Algorithmus zur  $LDL^T$  Faktorisierung (4.8) ist in Tabelle 4.1 angegeben. Die  $LDL^T$  Faktorisierung stellt bezüglich des numerischen Aufwandes den rechenintensivsten Lösungsschritt dar, bei dichtbesetzten Matrizen ist die Anzahl der durchzuführenden Operationen<sup>2</sup> in der Größenordnung von  $\frac{1}{6}n^3$ , siehe z.B. [49]. Bei dünn besetzten Matrizen kann eine solche Abhängigkeit nicht allgemein angegeben werden, sie variiert stark mit der Besetzungsstruktur, der gewählten Permutation und nicht zuletzt mit der verwendeten Speicherstruktur. Für den Sonderfall einer Tridiagonalmatrix beträgt der numerische Aufwand nur noch  $2(n - 1)$  Operationen, was die Abhängigkeit von der Besetzungsstruktur verdeutlicht.

---


$$\begin{aligned}
 &\text{für } i = 1, 2, \dots, n \\
 &\quad d_{ii} = a_{ii} \\
 &\quad l_{ii} = 1 \\
 &\quad \text{für } j = i + 1, \dots, n \\
 &\quad \quad \text{für } k = j, \dots, n \\
 &\quad \quad \quad a_{kj} = a_{kj} - \frac{a_{ki}a_{ji}}{a_{ii}} \\
 &\quad \quad \quad l_{ji} = \frac{a_{ji}}{a_{ii}}
 \end{aligned}$$


---

Tabelle 4.1: Algorithmus zur numerischen  $LDL^T$ -Faktorisierung

### 4.1.4 Rückwärts- und Vorwärtselimination

Wird eine  $LDL^T$  Faktorisierung (4.8) bei der Gleichungslösung vorgenommen, so muß die Rückwärts- bzw. Vorwärtselimination aus den Gleichungen (4.5) und (4.6) entsprechend modifiziert werden:

$$\mathbf{Ax} = \mathbf{L} \underbrace{\mathbf{D} \underbrace{\mathbf{L}^T \mathbf{x}}_{\mathbf{y}}}_{\bar{\mathbf{y}}} = \mathbf{b}, \tag{4.10}$$

und dann kann die weitere Lösung in drei Schritte aufgeteilt werden:

$$\mathbf{L}\bar{\mathbf{y}} = \mathbf{b} \tag{4.11}$$

---

<sup>2</sup>In diesem Zusammenhang sollen unter Operationen ausschließlich Multiplikationen und Divisionen verstanden werden, da diese deutlich aufwendiger sind als Additionen und Subtraktionen.

$$\mathbf{y} = \mathbf{D}^{-1}\bar{\mathbf{y}} \quad (4.12)$$

$$\mathbf{L}^T \mathbf{x} = \mathbf{y}. \quad (4.13)$$

Der Algorithmus hierzu ist in Tabelle 4.2 angegeben. Aus Effizienzgründen empfiehlt es sich, den zweiten Schritt (4.12) in den ersten Schritt (4.11) zu integrieren und damit einen Schleifendurchlauf zu vermeiden. Ebenso kann statt der Diagonalen  $\mathbf{D}$  deren Inverse  $\mathbf{D}^{-1}$  gespeichert werden, um so bei der Lösung mehrerer rechter Seiten Divisionen zu vermeiden, da diese mehr Rechenzeit benötigen als Multiplikationen, die mit der Inversen gebildet werden.

---

für $i = 1, \dots, n$ $\bar{y}_i = b_i$ für $j = 1, \dots, i - 1$ $\bar{y}_i = \bar{y}_i - l_{ij}y_j$ für $i = 1, \dots, n$ $y_i = \frac{\bar{y}_i}{d_{ii}}$	für $i = n, \dots, 1$ $x_i = y_i$ für $j = i + 1, \dots, n$ $x_i = x_i - l_{ij}x_j$
---	--

---

Tabelle 4.2: Algorithmus zur Vorwärts- und Rückwärtssubstitution

Der numerische Aufwand für (4.11) bis (4.13) ist klein verglichen mit der Faktorisierung. Bei dicht besetzten Matrizen ist er proportional zu  $n^2$ ; bei dünn besetzten Matrizen entsprechend niedriger. Bei direkten Lösungen spielt es demnach eine untergeordnete Rolle, ob die Lösung nur für eine oder aber für mehrere rechte Seiten berechnet werden muß – der numerische Aufwand ist nahezu konstant, da die numerische Faktorisierung dominiert. Insbesondere bei Bogenlängenverfahren nach Abschnitt 2.3.2 erweist sich diese Eigenschaft als vorteilhaft, da die Gleichungen (2.71) so deutlich effizienter gelöst werden können als das unsymmetrische Gleichungssystem (2.69).

## 4.2 Iterative Gleichungslösung

Bei der im vorigen Abschnitt beschriebenen direkten Gleichungslösung ist stets eine Faktorisierung der Koeffizientenmatrix  $\mathbf{A}$  notwendig. Ist  $\mathbf{A}$  sehr groß und dünn besetzt – wie bei den betrachteten Steifigkeitsmatrizen – so kann der Aufwand sehr groß werden, insbesondere da der entstehende Faktor vergleichsweise dicht besetzt sein kann. Eine grundsätzlich andere Vorgehensweise liegt darin, schrittweise eine Sequenz von Näherungsvektoren  $\mathbf{x}_j$ ,  $j = 0, 1, 2, \dots$  zu generieren, die – iterativ – gegen die wahre Lösung  $\mathbf{x}^*$  konvergiert. Dabei geht die Koeffizientenmatrix  $\mathbf{A}$  üblicherweise multiplikativ ein, d.h. die dünne Besetzung kann voll ausgenutzt werden. Entscheidend für die Effizienz bei diesem sogenannten iterativen Vorgehen ist die Konvergenzgeschwindigkeit. Dies kann auch als Nachteil iterativer Verfahren betrachtet werden, da vor der Ausführung der Aufwand bis zum Abschluß des Algorithmus nicht bekannt ist, was bei direkten Methoden hingegen der Fall ist.

### 4.2.1 Das Verfahren der konjugierten Gradienten

Eine von HESTENES und STIEFEL entwickelte nichtstationäre Methode zur Lösung eines linearen Gleichungssystems (4.1) ist das Verfahren der konjugierten Gradienten [63], abgekürzt mit cg-Verfahren (*conjugate gradient*). Im Gegensatz zu stationären Methoden, wie z.B. dem Jacobi- oder dem Gauß-Seidel Verfahren, ändern sich bei nichtstationären Verfahren die Iterationsparameter von Iteration zu Iteration. Die herausragende Eigenschaft des cg-Verfahrens ist, daß bei  $n$  Unbekannten Konvergenz zur wahren Lösung  $\mathbf{x}^*$  bei rundungsfreier Rechnung, d.h. exakter Mitnahme aller Dezimalstellen einer Zahl, in maximal  $n$  Schritten, erreicht wird.

Dem cg-Verfahren liegt folgende Überlegung zugrunde: Die Lösung des Gleichungssystems (4.1) wird aus der Minimierung des Potentials

$$\Phi(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} - \mathbf{x}^T \mathbf{b} \quad (4.14)$$

hergeleitet. Die Koeffizientenmatrix  $\mathbf{A}$  muß dabei symmetrisch und positiv definit sein. Das Minimum von  $\Phi$  wird für  $\mathbf{x}^* = \mathbf{A}^{-1} \mathbf{b}$  erreicht, d.h. für die wahre Lösung von (4.1).

Eine der einfachsten Methoden zur Minimierung einer Funktion ist die Methode des steilsten Abstiegs. In der Richtung ihres negativen Gradienten nimmt eine Funktion am schnellsten ab:

$$-\text{grad}\Phi(\mathbf{x}_k) = \mathbf{b} - \mathbf{A}\mathbf{x}_k = \mathbf{r}_k. \quad (4.15)$$

Der Vektor  $\mathbf{r}$  wird dabei als Residuum bezeichnet. Dann existiert ein Faktor  $\alpha$  für  $\mathbf{r}_k \neq \mathbf{0}$  so, daß folgendes gilt:

$$\Phi(\mathbf{x}_k + \alpha_k \mathbf{r}_k) < \Phi(\mathbf{x}_k). \quad (4.16)$$

Nun soll  $\alpha$  so gewählt werden, daß  $\Phi(\mathbf{x}_k + \alpha_k \mathbf{r}_k)$  minimiert wird. Aus

$$\frac{\partial \Phi(\mathbf{x}_k + \alpha_k \mathbf{r}_k)}{\partial \alpha_k} = 0 \quad (4.17)$$

folgt direkt

$$\alpha_k = \frac{\mathbf{r}_k^T \mathbf{r}_k}{\mathbf{r}_k^T \mathbf{A} \mathbf{r}_k}. \quad (4.18)$$

Die neue Näherung ist dann  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{r}_k$ .

Beim Verfahren der konjugierten Gradienten wird nun nicht in Richtung des Residuums minimiert, sondern in eine zunächst noch nicht weiter spezifizierte Suchrichtung  $\mathbf{p}$ . Das Minimum in  $\mathbf{p}$ -Richtung erhält man dann analog wie oben zu:

$$\frac{\partial \Phi(\mathbf{x}_k + \alpha_k \mathbf{p}_k)}{\partial \alpha} = 0 \quad \longrightarrow \quad \alpha_k = \frac{\mathbf{p}_k^T \mathbf{r}_k}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k}. \quad (4.19)$$

Die Richtungen  $\mathbf{p}_k$  werden nun so gewählt, daß sie  $\mathbf{A}$ -konjugiert sind, d.h. sie genügen der Bedingung

$$\mathbf{P}_{k-1}^T \mathbf{A} \mathbf{p}_k = 0. \quad (4.20)$$

Dabei ist  $\mathbf{P}_k$  eine Matrix, die als Spalten die Suchrichtungen  $\mathbf{p}_k$  enthält:

$$\mathbf{P}_k = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_k]. \quad (4.21)$$

Diese  $\mathbf{A}$ -Konjugiertheit hat den Vorteil, daß bei der Minimierung des Potentials  $\Phi$  ein Koppelterm zwischen dem Skalar  $\alpha$  und den Suchrichtungen  $\mathbf{p}$  verschwindet (siehe z.B. [53]). Bevor auf die Berechnung der Vektoren  $\mathbf{p}_k$  eingegangen wird, werden noch einige Beziehungen angegeben, die später benötigt werden. Mit  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$  gilt

$$\mathbf{r}_{k+1} = \mathbf{b} - \mathbf{A} \mathbf{x}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k, \quad (4.22)$$

d.h. die Residuen können rekursiv berechnet werden. Wegen der  $\mathbf{A}$ -Konjugiertheit der Suchrichtungen gilt außerdem

$$\mathbf{p}_j^T \mathbf{r}_{k+1} = 0 \quad \text{für } j \leq k. \quad (4.23)$$

Für  $j = k$  läßt sich dies mit Gleichung (4.22) und (4.19) und für  $j < k$  durch sukzessive Anwendung der Gleichungen (4.22) und (4.23) zeigen.

Die Suchrichtungen  $\mathbf{p}_k$  können nun ohne Schwierigkeit im Algorithmus aus den vorherigen Suchrichtungen erzeugt werden. Wählt man  $\mathbf{p}_0 = \mathbf{r}_0$  und

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k \quad \text{mit} \quad \beta_k = -\frac{\mathbf{p}_k^T \mathbf{A} \mathbf{r}_{k+1}}{\mathbf{p}_k^T \mathbf{A} \mathbf{p}_k}, \quad k = 0, 1, 2, \dots, \quad (4.24)$$

so sind die rekursiv erzeugten Suchrichtungen paarweise  $\mathbf{A}$ -konjugiert, d.h.  $\mathbf{r}_k^T \mathbf{r}_{k+1} = 0$ . Ebenso sind die Residuen paarweise orthogonal:  $\mathbf{p}_j^T \mathbf{A} \mathbf{p}_{k+1} = 0$ .

Dies soll in dem folgenden Abschnitt genutzt werden. Für  $k = 0$  ist die erste Beziehung durch Einsetzen von  $\mathbf{r}_1$  in (4.22), mit  $\mathbf{p}_0 = \mathbf{r}_0$  und mit Gleichung (4.19) einfach zu verifizieren. Es gilt

$$\mathbf{r}_0^T \mathbf{r}_1 = \mathbf{r}_0^T (\mathbf{r}_0 - \alpha_0 \mathbf{A} \mathbf{r}_0) = 0. \quad (4.25)$$

Die paarweise Orthogonalität für  $k = 0$  kann durch Einsetzen der Beziehung nach Gleichung (4.24) nachgewiesen werden. Hier ist

$$\mathbf{p}_0^T \mathbf{A} \mathbf{p}_1 = \mathbf{p}_0^T \mathbf{A} (\mathbf{r}_1 + \beta_0 \mathbf{p}_0) = 0. \quad (4.26)$$

<i>Phase I (Initialisierung)</i>	<i>Phase II (Iterationen)</i>
wähle erste Näherung $\mathbf{x}_0$	für $k = 0, 1, 2, \dots$
$\mathbf{p}_0 = \mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$	$\mathbf{q}_k = \mathbf{A}\mathbf{p}_k$
$\gamma_0 = \mathbf{r}_0^T \mathbf{r}_0$	$\alpha_k = \frac{\gamma_k}{\mathbf{p}_k^T \mathbf{q}_k}$
	$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$
	$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{q}_k$
	falls $\frac{\gamma_{k+1}}{\ \mathbf{b}\ } = \frac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\ \mathbf{b}\ } \leq \varepsilon$ Ende
	$\beta_k = \frac{\gamma_{k+1}}{\gamma_k}$
	$\mathbf{p}_{k+1} = \mathbf{p}_k + \beta_k \mathbf{p}_k$

Tabelle 4.3: cg-Algorithmus

Für  $k > 0$  kann der Nachweis mittels vollständiger Induktion und den Ergebnissen für  $k = 0$  für beide Beziehungen dann allgemein erfolgen. Mit Gleichung (4.23) und (4.24) gilt

$$\mathbf{r}_j^T \mathbf{r}_{k+1} = (\mathbf{p}_j - \beta_{j-1} \mathbf{p}_{j-1})^T \mathbf{r}_{k+1} = 0 \quad \text{für } j < k + 1, \quad (4.27)$$

und ebenso zunächst für  $j = k$  unter Beachtung der Definition von  $\beta_k$  in Gleichung (4.24)

$$\mathbf{p}_k^T \mathbf{A}\mathbf{p}_{k+1} = \mathbf{p}_k^T \mathbf{A}(\mathbf{r}_{k+1} + \beta_k \mathbf{p}_k) = 0. \quad (4.28)$$

Damit gilt nun für  $j < k$  und Gleichung (4.22) wegen der  $\mathbf{A}$ -Konjugiertheit der Suchrichtungen und (4.27)

$$\mathbf{p}_j^T \mathbf{A}\mathbf{p}_{k+1} = \mathbf{p}_j^T \mathbf{A}(\mathbf{r}_{k+1} + \beta_k \mathbf{p}_k) = \frac{1}{\alpha_j} (\mathbf{r}_j - \mathbf{r}_{j+1})^T \mathbf{r}_{k+1} + \beta_k \mathbf{p}_j^T \mathbf{A}\mathbf{p}_k = 0. \quad (4.29)$$

Mit den angegebenen Gleichungen ist nun nachgewiesen, daß die mit Gleichung (4.24) angegebenen Suchrichtungen  $\mathbf{A}$ -konjugiert und rekursiv aus den Residuen berechenbar sind, die sich ihrerseits rekursiv ermitteln lassen. Damit kann der cg-Algorithmus vollständig angegeben werden, siehe Tabelle 4.3. Als wesentliche numerische Operationen treten dabei eine Matrix-Vektor Multiplikation, drei Vektoraufaddierungen und zwei Skalarprodukte auf. Diese numerischen Operationen können auf einem Computer nicht rundungsfehlerfrei ausgeführt werden. Die entstehenden Fehler bewirken, daß die berechneten Suchrichtungen nicht exakt  $\mathbf{A}$ -konjugiert sind. Von diesen Rundungsfehlern kann also die Konvergenz des cg-Verfahrens beeinträchtigt werden.

### Konvergenzverhalten

Es ist bekannt, daß das Verfahren der konjugierten Gradienten bei rundungsfehlerfreier Rechnung in maximal  $n$  Schritten gegen die wahre Lösung  $\mathbf{x}^*$  konvergiert, siehe z.B. [96]

oder [53]. Das Konvergenzverhalten hängt aber von der Konditionszahl  $\kappa$  der Koeffizientenmatrix  $\mathbf{A}$  ab, die als Verhältnis des betragsgrößten zum betragskleinsten Eigenwert definiert ist:

$$\kappa(\mathbf{A}) = \frac{|\lambda_{max}|}{|\lambda_{min}|}. \quad (4.30)$$

Es kann gezeigt werden (siehe z.B. [4, 53, 74] oder [57]), daß für den Lösungsfehler im Schritt  $k$

$$\|\mathbf{x}_k - \mathbf{x}^*\|_{\mathbf{A}} \leq 2\alpha^k \|\mathbf{x}_0 - \mathbf{x}^*\|_{\mathbf{A}}, \quad \text{mit} \quad \alpha = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \quad (4.31)$$

gilt. D.h. je enger die Eigenwerte beieinanderliegen, desto kleiner wird  $\alpha$  in Gleichung (4.31) und der Fehler wird schnell reduziert.

#### 4.2.2 Vorkonditionierung

Wie bereits erwähnt, ist die Anzahl der Iterationen unter der Voraussetzung rundungsfreier Berechnung auf  $n$  beschränkt. Für sehr große Koeffizientenmatrizen ist aber auch dies eine unerwünscht hohe Anzahl. Bei der praktischen Verwendung des Verfahrens hat sich jedoch gezeigt, daß entsprechend Gleichung (4.31) häufig eine sehr viel geringere Anzahl von Iterationen für eine ausreichende Konvergenz genügt. Für schlecht konditionierte Matrizen ( $\kappa \gg 1$ ) ist allerdings die Konvergenzgeschwindigkeit klein und viele Iterationen sind notwendig. Da außerdem durch Rundungsfehler die Orthogonalität der Suchrichtungen verloren geht, können auch weit mehr als  $n$  Iterationen zur Lösung notwendig sein.

Um dieses Problem zu umgehen, wird die Koeffizientenmatrix  $\mathbf{A}$  *vorkonditioniert*, d.h. es wird eine (symmetrische) Matrix  $\mathbf{W}$  gesucht, sodaß

$$\tilde{\mathbf{A}} = \mathbf{W}^{-1} \mathbf{A} \mathbf{W}^{-1} \quad (4.32)$$

eine bessere Konditionszahl besitzt als  $\mathbf{A}$  und somit das vorkonditionierte Gleichungssystem

$$\tilde{\mathbf{A}} \tilde{\mathbf{x}} = \tilde{\mathbf{b}} \quad \text{mit} \quad \tilde{\mathbf{x}} = \mathbf{W} \mathbf{x} \quad \text{und} \quad \tilde{\mathbf{b}} = \mathbf{W}^{-1} \mathbf{b} \quad (4.33)$$

gelöst werden kann. Dann ist nach Gleichung (4.31) auch eine schnellere Konvergenz zu erwarten. In den cg-Algorithmus nach Tabelle 4.3 eingesetzt, berechnen sich nun die folgenden Größen zu

$$\tilde{\mathbf{q}}_k = \mathbf{W}^{-1} \mathbf{A} \mathbf{W}^{-1} \tilde{\mathbf{p}}_k, \quad (4.34)$$

$$\alpha_k = \frac{\gamma_k}{\tilde{\mathbf{p}}_k^T \tilde{\mathbf{q}}_k}, \quad (4.35)$$

$$\tilde{\mathbf{x}}_{k+1} = \tilde{\mathbf{x}}_k + \alpha_k \tilde{\mathbf{p}}_k, \quad (4.36)$$

$$\tilde{\mathbf{r}}_{k+1} = \tilde{\mathbf{r}}_k - \alpha_k \tilde{\mathbf{q}}_k, \quad (4.37)$$

$$\gamma_{k+1} = \tilde{\mathbf{r}}_k^T \tilde{\mathbf{r}}_k \quad \text{und} \quad (4.38)$$

$$\tilde{\mathbf{p}}_{k+1} = \tilde{\mathbf{p}}_k + \beta_k \tilde{\mathbf{p}}_k. \quad (4.39)$$

Wählt man nun

$$\mathbf{M} = \mathbf{W}\mathbf{W}^T, \quad (4.40)$$

$$\mathbf{p}_k = \mathbf{W}^{-1} \tilde{\mathbf{p}}_k, \quad (4.41)$$

$$\mathbf{x}_k = \mathbf{W}^{-1} \tilde{\mathbf{x}}_k, \quad (4.42)$$

$$\mathbf{z}_k = \mathbf{W}^{-1} \tilde{\mathbf{r}}_k \quad \text{und} \quad (4.43)$$

$$\mathbf{r}_k = \mathbf{W} \tilde{\mathbf{r}}_k = \mathbf{b} - \mathbf{A}\mathbf{x}_k, \quad (4.44)$$

so erhält man den vorkonditionierten cg-Algorithmus in Tabelle 4.4. Darin muß die vorkonditionierte Koeffizientenmatrix  $\tilde{\mathbf{A}}$ , die nicht notwendigerweise dünn besetzt ist, nicht explizit berechnet werden. Im Vergleich zum cg-Algorithmus nach Tabelle 4.3 kommt im Wesentlichen der Schritt der Lösung des linearen Gleichungssystems  $\mathbf{M}\mathbf{z}_{k+1} = \mathbf{r}_{k+1}$  hinzu.

Damit die Vorkonditionierung das cg-Verfahren auch bezüglich der Zahl der numerischen Operationen beschleunigt, muß das Vorkonditionierungssystem sehr viel einfacher und schneller lösbar sein als das ursprüngliche Gleichungssystem (4.1). Andererseits soll die Vorkonditionierungsmatrix  $\mathbf{M}$  die Koeffizientenmatrix  $\mathbf{A}$  möglichst gut approximieren, damit sich die spektralen Eigenschaften des vorkonditionierten Systems verbessern. Wird z.B.  $\mathbf{M} = \mathbf{A}$  gewählt, so wird die Lösung in einem Schritt erzielt, da dann  $\tilde{\mathbf{A}} = \mathbf{I}$  gilt und alle Eigenwerte den Wert Eins annehmen. Allerdings wird nun die Lösung des Gleichungssystems auf das Lösen des Vorkonditionierungsproblems verschoben, das iterative Verfahren wird so zur direkten Lösung.

<i>Phase I (Initialisierung)</i>	<i>Phase II (Iterationen)</i>
wähle erste Näherung $\mathbf{x}_0$	für $k = 0, 1, 2, \dots$
$\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$	$\mathbf{q}_k = \mathbf{A}\mathbf{p}_k$
löse $\mathbf{M}\mathbf{z}_0 = \mathbf{r}_0$	$\alpha_k = \frac{\gamma_k}{\mathbf{p}_k^T \mathbf{q}_k}$
$\mathbf{p}_0 = \mathbf{z}_0, \gamma_0 = \mathbf{z}_0^T \mathbf{r}_0$	$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$
	$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k \mathbf{q}_k$
	löse $\mathbf{M}\mathbf{z}_{k+1} = \mathbf{r}_{k+1}$
	falls $\frac{\gamma_{k+1}}{\ \mathbf{b}\ } = \frac{\mathbf{z}_{k+1}^T \mathbf{r}_{k+1}}{\ \mathbf{b}\ } \leq \varepsilon$ Ende
	$\beta_k = \frac{\gamma_{k+1}}{\gamma_k}$
	$\mathbf{p}_{k+1} = \mathbf{z}_{k+1} + \beta_k \mathbf{p}_k$

Tabelle 4.4: Vorkonditionierter cg-Algorithmus

In den folgenden Abschnitten wird auf einige Möglichkeiten der Vorkonditionierung eingegangen, die weitgehend auch im Schrifttum (siehe z.B. [4, 15] oder auch [127]) zu finden sind. Die meisten dieser Vorkonditionierungsmatrizen lassen sich direkt aus der Koeffizientenmatrix  $A$  ableiten und benutzen keinerlei Information über die Aufstellung dieser Matrix. Deshalb können sie für beliebige lineare Gleichungssysteme gleichermaßen verwendet werden. Bei zwei Vorkonditionierungsstrategien wird allerdings die Herkunft der Koeffizientenmatrizen aus der Methode der Finiten Elemente ausgenutzt, ohne daß jedoch besondere Voraussetzungen für die Elemente erfüllt werden müssen.

Vorkonditionierungsstrategien, die auf Mehrgitter-Techniken bzw. auf hierarchischen Basen beruhen (siehe z.B. [77, 12, 42, 126] oder [5]) werden in dieser Arbeit aus Gründen der Beschränkung des Umfangs nicht untersucht. Diese Verfahren hängen zum Teil stark von den Elementtypen ab und der Aufwand zur Umsetzung in ein FE-Programm ist deutlich größer als für die hier betrachteten cg-Verfahren. Insbesondere die Restriktions- bzw. Prolongationsoperationen müssen auf die jeweiligen Elemente abgestimmt werden. Treten gar innere Elementvariable auf, wie z.B. EAS-Parameter oder geschichtsabhängige Materialgrößen wie bei Plastizität, so müssen auch diese auf die Netzhierarchien übertragen werden. Es sei aber bemerkt, daß Mehrgitter-Techniken ein enormes Potential besitzen [6, 76, 147].

#### 4.2.2.1 Jacobi-Vorkonditionierung

Eine der einfachsten Strategien zur Vorkonditionierung besteht in der Skalierung mit der Diagonalen der Koeffizientenmatrix:

$$M = \text{diag}(A), \quad (4.45)$$

die als Jacobi-Vorkonditionierung oder als Diagonalskalierung bezeichnet wird. Theoretisch läßt sich diese Vorkonditionierung ohne jeden zusätzlichen Speicheraufwand realisieren. Da jedoch Divisionen numerisch aufwendig sind, wird meist  $M^{-1}$  berechnet und abgespeichert, d.h. es wird ein zusätzlicher Vektor der Länge  $n$  benötigt.

Diese Vorkonditionierung ist sehr einfach zu implementieren und reduziert die Anzahl der benötigten Iterationen bei einigen Problemklassen erheblich, wie numerische Ergebnisse in nachfolgenden Abschnitten belegen. Für schlecht konditionierte Probleme sind jedoch andere Verfahren erforderlich.

#### 4.2.2.2 SSOR-Vorkonditionierung

Wie schon bei der Jacobi-Vorkonditionierung muß auch für die SSOR-Vorkonditionierung die Matrix  $M$  nicht vorab berechnet werden, sondern wird unmittelbar aus Teilen von  $A$  zusammengesetzt:

$$M(\omega) = \frac{1}{2 - \omega} \left( \frac{1}{\omega} \tilde{D} + \tilde{L} \right) \left( \frac{1}{\omega} \tilde{D}^{-1} \right) \left( \frac{1}{\omega} \tilde{D} + \tilde{L} \right)^T. \quad (4.46)$$

$A = \tilde{L} + \tilde{D} + \tilde{L}^T$  wird dazu additiv in den unteren Dreiecksanteil  $\tilde{L}$ , den Diagonalanteil  $\tilde{D} = \text{diag}(A)$  und den oberen Dreiecksanteil  $\tilde{L}^T$  aufgespalten. Die SSOR-Vorkonditionierung ist wie in Gleichung (4.46) angegeben von dem Parameter  $0 < \omega < 2$  abhängig. Durch eine günstige Wahl kann die Iterationszahl deutlich gesenkt werden, wie numerische Studien [93] zeigen. Für allgemeine Problemklassen ist aber a priori kein vorteilhafter Wert bekannt und es wird  $\omega = 1$  benutzt.

Wie bereits bei der Jacobi-Vorkonditionierung ist es auch hier aus denselben Gründen vorteilhaft, die Werte für  $1/\omega \tilde{D}^{-1}$  vorab zu berechnen und zu speichern.

Von EISENSTAT [35] liegt ein Vorschlag für eine effiziente Implementierung vor, wobei zuvor eine explizite Skalierung der Koeffizientenmatrix benötigt wird, pro Iteration dann aber Operationen gespart werden, sodaß letztlich mit dieser Implementierung kürzere Rechenzeiten erreicht werden.

#### 4.2.2.3 Unvollständige $LDL^T$ -Zerlegungen

Eine sehr effiziente Art der Vorkonditionierung ist die unvollständige  $LDL^T$ -Zerlegung. Hierzu wird eine Zerlegung

$$M = \hat{L}\hat{D}\hat{L}^T \approx LDL^T = A \quad (4.47)$$

vorgenommen, die möglichst gut die vollständige  $LDL^T$ -Zerlegung approximiert, dabei aber leicht bestimmbar und einfach lösbar ist. Da bei der vollständigen  $LDL^T$ -Zerlegung sogenannter *fill-in* entsteht, ist der Faktor  $L$  im Allgemeinen sehr viel dichter besetzt als die Koeffizientenmatrix  $A$ , d.h. dies gilt es zur effizienten Vorkonditionierung zu vermeiden.

Da durch das Vernachlässigen von *fill-in* der nachfolgende numerische Ablauf der Faktorisierung beeinflusst wird, ist oft die Reihenfolge der Abarbeitung der Gleichungen wesentlich. Als Folge einer Permutation der Koeffizientenmatrix wird demnach auch eine andere Vorkonditionierungsmatrix berechnet, die nicht nur durch eine Permutation der mit der unpermutierten Koeffizientenmatrix berechneten unvollständigen Zerlegung erreicht werden kann. Mittels geschickter Permutation der Koeffizientenmatrix kann somit auch eine weitere Verbesserung der Vorkonditionierung erzielt werden, wie in Abschnitt 4.3.2 gezeigt wird.

#### 4.2.2.4 Unvollständige $LDL^T$ -Zerlegung auf dem Speicherplatz von $A$ (MPILU)

Eine Möglichkeit, eine unvollständige Faktorisierung zu generieren, besteht darin, allen entstehenden *fill-in* zu vernachlässigen (ILU<sup>3</sup>), der Faktor  $\hat{L}$  hat dann die gleiche Besetzungsstruktur wie die Koeffizientenmatrix  $A$ . Die Vorkonditionierung muß vor dem Start des Iterationsverfahrens bestimmt werden, das Lösen des Vorkonditionierungssystems erfolgt durch Rückwärts- und Vorwärtseinsetzen, wozu etwa so viele Operationen benötigt werden wie für das Matrix-Vektor Produkt.

<sup>3</sup>Von Incomplete LU.

Es ist aber nicht sichergestellt, daß die so generierte unvollständige  $\hat{\mathbf{L}}\hat{\mathbf{D}}\hat{\mathbf{L}}^T$ -Zerlegung existiert, bzw. positiv definit ist. Deshalb muß bei der Durchführung sichergestellt werden, daß dies der Fall ist, andernfalls ist die Zerlegung mit einer modifizierten, diagonaldominanteren Koeffizientenmatrix

$$\hat{\mathbf{A}} = \mathbf{A} + \omega \operatorname{diag} \mathbf{A} \quad (4.48)$$

erneut durchzuführen (MPILU<sup>4</sup>). Für die Wahl von  $\omega$  in (4.48) gibt es keine allgemeine Aussage.  $\omega$  soll so klein wie möglich sein, damit  $\mathbf{M}$  die Koeffizientenmatrix  $\mathbf{A}$  gut approximiert, aber doch groß genug, damit die unvollständige Zerlegung positiv definit wird. Weitere Verfahren, um die Existenz und positive Definitheit zu sichern, werden z.B. in [4] mit weiteren Literaturhinweisen diskutiert. In der Praxis hat sich das eben beschriebene Verfahren als robust und praktikabel erwiesen, wobei  $\omega$  für die im Rahmen dieser Arbeit betrachteten Problemklassen fest mit 1/64 vorgegeben wurde. So sind nur einige wenige Wiederholungen der unvollständigen Faktorisierung erforderlich.

#### 4.2.2.5 Blockweise unvollständige $LDL^T$ -Zerlegung (Block-MPILU)

Von RASHID [115] wurde in anderem Zusammenhang zur Lösung dreidimensionaler Probleme vorgeschlagen, die Steifigkeitsmatrix zu permutieren, sodaß die Blöcke mit Freiheitsgraden gleicher Richtung entsprechen. Dieser Grundgedanke soll hier zur Konstruktion einer Variante der MPILU-Vorkonditionierung genutzt werden, die zwar die Herkunft der Koeffizientenmatrix aus der Finiten Element Methode ausnutzt, aber dennoch unabhängig von den Elementformulierungen ist.

Bei Volumenelementen besitzt jeder Knoten drei Verschiebungsfreiheitsgrade, jeweils einen in den drei Raumrichtungen  $x$ ,  $y$  und  $z$ . Nach Durchführung obiger Permutation hat die Koeffizientenmatrix  $\mathbf{A}$  die folgende Gestalt

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{xx} & \mathbf{A}_{xy} & \mathbf{A}_{xz} \\ \mathbf{A}_{yx} & \mathbf{A}_{yy} & \mathbf{A}_{yz} \\ \mathbf{A}_{zx} & \mathbf{A}_{zy} & \mathbf{A}_{zz} \end{bmatrix}. \quad (4.49)$$

Die Diagonal-Blockmatrix

$$\mathbf{M} = \begin{bmatrix} \mathbf{A}_{xx} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{yy} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{zz} \end{bmatrix} \quad (4.50)$$

kann als Näherung für  $\mathbf{A}$  zur Vorkonditionierung verwendet werden, was insbesondere bei dreidimensionalen Problemen sehr gute Resultate verspricht, da dort die Diagonale sehr dominant ist.

---

<sup>4</sup>Modified Pivot ILU.

Für Schalenprobleme kann diese Vorkonditionierung ebenfalls eingesetzt werden, jedoch muß es nicht immer günstig sein, dabei im Sinne von Gleichung (4.50) alle Freiheitsgrade voneinander zu entkoppeln, sondern gegebenenfalls mehrere unterschiedliche Freiheitsgrade in einem Block zu belassen. Zum Beispiel könnte für ein Schalenelement mit fünf Freiheitsgraden (drei Verschiebungs- und zwei Rotationsfreiheitsgraden) die Struktur

$$M = \begin{bmatrix} A_{xx} & A_{xy} & 0 & 0 & 0 \\ A_{yx} & A_{yy} & 0 & 0 & 0 \\ 0 & 0 & A_{zz} & 0 & 0 \\ 0 & 0 & 0 & \varphi_1 & 0 \\ 0 & 0 & 0 & 0 & \varphi_2 \end{bmatrix} \quad (4.51)$$

gewählt werden, was insbesondere dann sinnvoll sein kann, wenn in der  $xy$ -Ebene z.B. der Membrananteil und in  $z$ -Richtung der Biegeanteil für die Verschiebungen dominierend ist.

Da die Vorkonditionierungsmatrix (4.50) bzw. (4.51) noch dünner besetzt ist als die Koeffizientenmatrix  $A$ , kann sowohl die unvollständige Zerlegung, als auch das Rückwärts- und Vorwärtseinsetzen effizienter erfolgen als bei der vorher beschriebenen unvollständigen Zerlegung auf dem Speicherplatz von  $A$ . Da die Qualität der Vorkonditionierung gegenüber der unvollständigen Faktorisierung auf dem Speicherplatz von  $A$  durch weitere Vernachlässigungen sinkt, sind meist mehr Iterationen zu erwarten. Die gesamte Speichersparnis gegenüber der MPILU-Version ist allerdings unbedeutend, da für die Vorkonditionierungsmatrix nun ein neuer Satz von Zeigern für die neue Matrixstruktur generiert werden muß.

Ein Nachteil dieser Strategie liegt auch darin, daß die Koeffizientenmatrix zunächst permutiert werden muß. Der Permutationsvorgang selbst ist zwar wenig aufwendig, da vor dem Erstellen der Matrix ein Permutationsvektor berechnet und so die Matrix direkt in der gewünschten Form assembliert werden kann, jedoch ist die Bandbreite der permutierten Matrix sehr viel größer, als die der unpermutierten. Bei der iterativen Lösung hat dies den Effekt, daß die Rechenzeit eines Matrix-Vektor Produktes erhöht werden kann, da aufeinanderfolgend benötigte Vektorkomponenten sich seltener im *cache* befinden und nachgeladen werden müssen<sup>5</sup>. Als günstiger erweist es sich, die Koeffizientenmatrix nicht zu permutieren, sondern zur Vorkonditionierung die unerwünschten Einträge herauszufiltern. Die Berechnung des Matrix-Vektor Produktes bleibt so unbeeinflusst, allerdings wird eine andere unvollständige Zerlegung berechnet. Letztere hat sich aber gegenüber der permutierten Matrix als gleichwertig erwiesen, siehe Abschnitt 4.3.3. Insgesamt wird die Lösung des linearen Gleichungssystem damit beschleunigt.

#### 4.2.2.6 Unvollständige $LDL^T$ -Zerlegung mit *fill-in* erster Stufe (FLILU)

Eine bessere Approximation der vollständigen  $LDL^T$ -Zerlegung kann durch die Berücksichtigung von *fill-in* erzielt werden. Zum Beispiel kann der *fill-in*, der direkt durch Matrixeinträge erzeugt wird, berücksichtigt werden; *fill-in*, der in der Folge dieses *fill-ins* erzeugt

<sup>5</sup>Dieser Effekt heißt *cachefault* und ist von der verwendeten Rechnerarchitektur und den problembezogenen Vektorlängen abhängig.

wird, jedoch vernachlässigt werden. Dieses Vorgehen wird als *First Level* ILU (FLILU) bezeichnet, im Schrifttum findet sich ebenso die Bezeichnung ILU(1).

#### 4.2.2.7 Unvollständige $LDL^T$ -Zerlegung mit *numerical dropping* (ND-ILU)

Bei den bislang beschriebenen unvollständigen  $LDL^T$ -Zerlegungen wurde vorab die Besetzungsstruktur der Vorkonditionierungsmatrix unabhängig von den numerischen Einträgen festgelegt. Eine andere Vorgehensweise wird z.B. in [69, 105] oder [107] diskutiert, wo die Vernachlässigung von Einträgen in  $\hat{L}$  von ihrer numerischen Größe abhängt. Es bestehen zwischen diesen Ansätzen aber Unterschiede darin, wie die Entscheidung gefällt wird, ob ein Eintrag klein ist, oder nicht. Durch einen vom Benutzer vorzugebenden Parameter  $\psi$  kann die Größe der Vorkonditionierungsmatrix damit beeinflußt, nicht jedoch exakt vorbestimmt werden. Der größte Nachteil dieser Implementierung trotz der oft erreichbaren hohen Qualität der Vorkonditionierung liegt darin, daß zum Teil aufwendige Indexoperationen erforderlich sind, um die Struktur der Vorkonditionierungsmatrix zu ermitteln. Die Rechenzeit zur Erstellung des Vorkonditionierers kann dementsprechend sehr hoch sein und ein Mehrfaches des Aufwandes zur Lösung des linearen Gleichungssystems betragen. In [30] wird diese Vorkonditionierung insbesondere für 3D-Kontinuumsprobleme mit stark verzerrten Tetraederelementen empfohlen.

#### 4.2.2.8 Element-by-Element (EBE) Vorkonditionierung

Von HUGHES wurde in [67] die sogenannte *element-by-element* Methode eingeführt. Die Vorkonditionierungsmatrix ist durch

$$M = \text{diag} A^{\frac{1}{2}} \prod_{e=1}^{n_{el}} L^e \prod_{e=1}^{n_{el}} \text{diag} D^e \prod_{e=n_{el}}^1 L^{eT} \text{diag} A^{\frac{1}{2}} \quad (4.52)$$

mit

$$\overline{K}^e = L^e D^e L^{eT} = I + \text{diag} A^{-\frac{1}{2}} (K^e - \text{diag} K^e) \text{diag} A^{-\frac{1}{2}} \quad (4.53)$$

gegeben, siehe z.B. [66]. Dabei ist  $K^e$  die Elementsteifigkeitsmatrix des  $e$ -ten Elements,  $e = 1, 2, \dots, n_{el}$ . Die Regularisierung (4.53) ist erforderlich, da die Elementsteifigkeitsmatrizen  $K^e$  singular sind, sofern sie nicht durch Randbedingungen in ihren Freiheitsgraden eingeschränkt werden und ihre  $LDL^T$ -Zerlegung deshalb nicht existiert.

Die EBE-Vorkonditionierung zeichnet sich dadurch aus, daß sie vollständig auf der Elementebene durchgeführt werden kann; die Vorkonditionierungsmatrix  $M$  (4.52) muß nie assembliert werden. Im vorkonditionierten cg-Algorithmus in Tabelle 4.4 wird zur Vorkonditionierung das lineare Gleichungssystem  $Mz = r$  gelöst, hier mit der Vorkonditionierungsmatrix aus (4.52). Diese Lösung erhält man mit

$$M^{-1} = \text{diag} A^{-\frac{1}{2}} \prod_{e=1}^{n_{el}} L^{e-1} \prod_{e=1}^{n_{el}} \text{diag} D^{e-1} \prod_{e=n_{el}}^1 L^{eT-1} \text{diag} A^{-\frac{1}{2}}. \quad (4.54)$$

aus den faktorisierten, regularisierten Elementsteifigkeitsmatrizen durch sukzessive Multiplikation:

$$\mathbf{z}' = \text{diag} \mathbf{A}^{-\frac{1}{2}} \mathbf{r} \quad (4.55)$$

$$\mathbf{z}'' = \prod_{e=n_{el}}^1 \mathbf{L}^{eT-1} \mathbf{z}' \quad (4.56)$$

$$\mathbf{z}''' = \prod_{e=1}^{n_{el}} \text{diag} \mathbf{D}^{e-1} \mathbf{z}'' \quad (4.57)$$

$$\mathbf{z}'''' = \prod_{e=1}^{n_{el}} \mathbf{L}^{e-1} \mathbf{z}''' \quad \text{und} \quad (4.58)$$

$$\mathbf{z}''''' = \text{diag} \mathbf{A}^{-\frac{1}{2}} \mathbf{z}'''' . \quad (4.59)$$

Grundsätzlich bedarf es für diese Vorgehensweise nur Speicher für eine Elementsteifigkeitsmatrix, da alle Operationen sukzessive ausgeführt werden können. Dann müssen aber alle Elementsteifigkeitsmatrizen und deren  $\mathbf{LDL}^T$ -Zerlegung für jede Iteration doppelt berechnet werden, was aus Effizienzgründen nicht empfehlenswert ist, zumal bei modernen Finiten Elementen der Aufwand zur Berechnung der Elementsteifigkeitsmatrizen nicht unerheblich ist. Deutlich effizienter ist es, alle regularisierten, faktorisierten Elementsteifigkeitsmatrizen abzuspeichern.

Ergebnisse in [66] zeigen, daß mit dieser Vorkonditionierung bessere Konvergenzeigenschaften erzielt werden, als mit der Jacobi-Vorkonditionierung. WATHEN vermutet in [151] für die Poissongleichung auf dem Einheitsquadrat, daß die EBE-Vorkonditionierung spektral äquivalent zur Jacobi-Vorkonditionierung, für große Anzahlen von Freiheitsgraden letzterer jedoch überlegen ist. Insgesamt wird aber nicht die Qualität der MPILU-Vorkonditionierung erreicht. Da darüber hinaus durch das Abspeichern der faktorisierten Elementsteifigkeitsmatrizen sich auch der Speicherbedarf als nicht unerheblich erweist, ist diese Vorkonditionierung nicht sehr effizient.

### 4.2.3 Das Lanczos Verfahren

Von LANCZOS wurde 1950 in [90] ein Verfahren vorgeschlagen, das eine  $(n \times n)$  Matrix  $\mathbf{A}$  mittels einer Ähnlichkeitstransformation in eine Tridiagonalmatrix  $\mathbf{T}$  transformiert:

$$\mathbf{T} = \mathbf{Q}^T \mathbf{A} \mathbf{Q} \quad \text{bzw.} \quad \mathbf{A} \mathbf{Q} = \mathbf{Q} \mathbf{T}. \quad (4.60)$$

Die Matrix  $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n]$  ist dabei orthogonal

$$\mathbf{Q} \mathbf{Q}^T = \mathbf{I}, \quad (4.61)$$

<i>Phase I (Initialisierung)</i>	<i>Phase II (Iterationen)</i>
$\mathbf{r}_0 = \mathbf{q}_1$	für $j = 1, 2, \dots, n$
$\beta_0 = 1$	$\alpha_j = \mathbf{q}_j^T \mathbf{A} \mathbf{q}_j$
$\mathbf{q}_0 = \mathbf{0}$	$\mathbf{r}_j = \mathbf{A} \mathbf{q}_j - \alpha \mathbf{q}_j - \beta_{j-1} \mathbf{q}_{j-1}$
wähle $\mathbf{q}_1$	$\beta_j = \ \mathbf{r}_j\ _2$
	$\mathbf{q}_{j+1} = \mathbf{r}_j / \beta_j$

Tabelle 4.5: Tridiagonalisierung der Matrix  $\mathbf{A}$ 

und die Vektoren  $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_n$  spannen einen sogenannten Krylov-Unterraum  $\mathcal{K}$  auf. Mit

$$\mathbf{T} = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & & & \\ & & \ddots & & \\ & & & \ddots & \beta_{n-1} \\ & & & \beta_{n-1} & \alpha_n \end{bmatrix} \quad (4.62)$$

und (4.60) läßt sich durch Betrachtung der  $j$ -ten Spalte die Gleichung

$$\mathbf{A} \mathbf{q}_j = \beta_{j-1} \mathbf{q}_{j-1} + \alpha_j \mathbf{q}_j + \beta_j \mathbf{q}_{j+1} \quad (4.63)$$

ablesen. In den obigen Gleichungen wurde die Symmetrie von  $\mathbf{A}$  eingearbeitet, da sich andernfalls eine unsymmetrische Tridiagonalmatrix ergäbe, was aber allgemein durchaus zulässig ist. Aufgrund der Orthogonalität der Vektoren  $\mathbf{q}_j$  und mit (4.62) gilt demnach

$$\alpha_j = \mathbf{q}_j^T \mathbf{A} \mathbf{q}_j. \quad (4.64)$$

Setzt man zudem  $\mathbf{q}_{j+1} = \mathbf{r}_j / \beta_j$ , so erhält man aus (4.63)

$$\mathbf{r}_j = (\mathbf{A} - \alpha_j \mathbf{I}) \mathbf{q}_j - \beta_{j-1} \mathbf{q}_{j-1}. \quad (4.65)$$

Hiermit kann ein Algorithmus formuliert werden, der die Tridiagonalisierung im Krylov-Raum durchführt, siehe Tabelle 4.5 und z.B. [53]. Die Transformation erfolgt vollständig iterativ, wie Tabelle 4.5 entnommen werden kann, die Einträge  $\alpha_j$  und  $\beta_j$  der Tridiagonalmatrix  $\mathbf{T}$  in (4.62) werden sukzessive bestimmt. Nach  $n$  Iterationen ist die Transformation abgeschlossen. Die Tridiagonalmatrix  $\mathbf{T}$  kann anschließend auch zur Bestimmung der Eigenwerte von  $\mathbf{A}$  verwendet werden, oder sie kann zwecks der Lösung eines linearen Gleichungssystem mit der Koeffizientenmatrix  $\mathbf{A}$  faktorisiert werden. Letztes wurde 1952 von LANCZOS in [91] vorgeschlagen und wird hier im Folgenden beschrieben. Im Unterschied zum Verfahren der konjugierten Gradienten (vgl. Abschnitt 4.2.1), das etwa zeitgleich von HESTENES und STIEFEL entwickelt wurde, werden dazu keine Einschränkungen bezüglich der Definitheit der Koeffizientenmatrix gemacht.

In der  $j$ -ten Iteration stellt sich die Gleichung (4.60) wie folgt dar:

$$\mathbf{A}\mathbf{Q}_j = \mathbf{Q}_j\mathbf{T}_j + \mathbf{r}_j\mathbf{e}_j^T \quad \text{mit} \quad \mathbf{r}_j = \beta_j\mathbf{q}_{j+1}. \quad (4.66)$$

Dabei ist  $\mathbf{e}_j$  der  $j$ -te Einheitsvektor. Die Tridiagonalmatrix  $\mathbf{T}_j$  hat die Gestalt

$$\mathbf{T}_j = \mathbf{Q}_j^T \mathbf{A} \mathbf{Q}_j = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \ddots & & \\ & \ddots & \ddots & \beta_{j-1} & \\ & & & \beta_{j-1} & \alpha_j \end{bmatrix}. \quad (4.67)$$

Die Lösung des linearen Gleichungssystems im Krylov Unterraum  $\mathcal{K}$  erfolgt dann wie folgt (siehe z.B. [108] oder [109]). Für eine gegebene Anfangsnäherung  $\mathbf{x}^a$  gilt dabei

$$\mathbf{A}\mathbf{x}^c = \mathbf{r}_0 \quad \text{mit} \quad \mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}^a, \quad (4.68)$$

wobei die Summe aus  $\mathbf{x}^a$  und  $\mathbf{x}^c$  die wahre Lösung  $\mathbf{x}^*$  des linearen Gleichungssystems darstellt und  $\mathbf{r}_0$  das Anfangsresiduum bezeichnet. Mit der Lösung von (4.68) ist dann auch die Lösung von (4.1) bekannt. Unter Anwendung der Transformation

$$\mathbf{x}_j^c = \mathbf{Q}_j\mathbf{y}_j \quad (4.69)$$

folgt aus (4.68)

$$\begin{aligned} \mathbf{r}_0 &= \mathbf{A}\mathbf{Q}_j\mathbf{y}_j \\ \mathbf{Q}_j^T \mathbf{r}_0 &= \mathbf{Q}_j^T \mathbf{A} \mathbf{Q}_j \mathbf{y}_j \end{aligned} \quad (4.70)$$

und mit Gleichung (4.67)

$$\mathbf{Q}_j^T \mathbf{r}_0 = \mathbf{T}_j \mathbf{y}_j. \quad (4.71)$$

Die  $\mathbf{LDL}^T$ -Zerlegung der Tridiagonalmatrix  $\mathbf{T}_j$  in (4.71) liefert dann

$$\mathbf{L}_j \mathbf{D}_j \mathbf{L}_j^T \cdot \mathbf{y}_j = \mathbf{Q}_j^T \mathbf{r}_0. \quad (4.72)$$

Mit  $\mathbf{L}_j^T \mathbf{y}_j = \mathbf{z}_j$  und (4.69):

$$\mathbf{z}_j = \mathbf{L}_j^T \mathbf{Q}_j \mathbf{x}_j \quad (4.73)$$

wird (4.72) zu

$$\mathbf{L}_j \mathbf{D}_j \mathbf{z}_j = \mathbf{Q}_j^T \mathbf{r}_0. \quad (4.74)$$

Setzt man weiterhin

$$\mathbf{L}_j \mathbf{C}_j = \mathbf{Q}_j^T, \quad (4.75)$$

führt dies mit (4.69) auf

$$\mathbf{x}_j = \mathbf{C}_j \mathbf{L}_j^T \mathbf{y}_j = \mathbf{C}_j \mathbf{z}_j, \quad (4.76)$$

und man kann dann die Gleichungen (4.74) und (4.75) in der Form

$$\begin{bmatrix} 1 & & & & \\ \delta_2 & 1 & & & \\ & & \ddots & & \\ & & & \delta_j & 1 \end{bmatrix} \begin{bmatrix} d_1 & & & & \\ & \ddots & & & \\ & & & & d_j \end{bmatrix} \begin{bmatrix} \zeta_1 \\ \vdots \\ \zeta_j \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1^T \\ \vdots \\ \mathbf{q}_j^T \end{bmatrix} \mathbf{r}_0 \quad (4.77)$$

$$\begin{bmatrix} 1 & & & & \\ \delta_2 & 1 & & & \\ & & \ddots & & \\ & & & \delta_j & 1 \end{bmatrix} \begin{bmatrix} \mathbf{c}_1^T \\ \vdots \\ \mathbf{c}_j^T \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1^T \\ \vdots \\ \mathbf{q}_j^T \end{bmatrix} \mathbf{r}_0 \quad (4.78)$$

ausschreiben. Aus der ersten dieser beiden Gleichungen (4.77) folgt nun

$$d_j \zeta_j + d_{j-1} \zeta_{j-1} \delta_j = \mathbf{q}_j^T \mathbf{r}_0, \quad (4.79)$$

und daraus wiederum

$$\zeta_j = \frac{1}{d_j} (\mathbf{q}_j^T \mathbf{r}_0 - d_{j-1} \zeta_{j-1} \delta_j). \quad (4.80)$$

Aus der zweiten Gleichung (4.78) kann man direkt

$$\mathbf{c}_{j-1}^T \delta_j + \mathbf{c}_j^T = \mathbf{q}_j^T \quad (4.81)$$

und damit

$$\mathbf{c}_j = \mathbf{q}_j^T - \delta \mathbf{c}_{j-1}^T \quad (4.82)$$

ablesen. Nun läßt sich die gewünschte Korrektur  $\mathbf{x}^c$  aus (4.76) als

$$\mathbf{x}_j = \mathbf{C}_j \mathbf{z}_j = \mathbf{C}_{j-1} \mathbf{z}_{j-1} + \mathbf{c}_j \zeta_j = \mathbf{x}_{j-1} + \mathbf{c}_j \zeta_j \quad (4.83)$$

angeben und so die Lösung des linearen Gleichungssystems vollständig rekursiv formulieren, ohne daß alle Lanczos-Vektoren gespeichert werden müssen.

Analog zum cg-Algorithmus kann auch beim Lanczos Algorithmus die Anzahl der Iterationen, die notwendig sind, um eine ausreichend genaue Lösung des linearen Gleichungssystems zu berechnen, mit Hilfe von Vorkonditionierungsstrategien reduziert werden. Dann wird nicht das Gleichungssystem (4.1), sondern

$$\tilde{\mathbf{A}}\tilde{\mathbf{x}} = \tilde{\mathbf{b}} \quad (4.84)$$

mit

$$\tilde{\mathbf{A}} = \mathbf{W}^{-1}\mathbf{A}\mathbf{W}^{-1}, \quad \tilde{\mathbf{x}} = \mathbf{W}\mathbf{x} \quad \text{und} \quad \tilde{\mathbf{b}} = \mathbf{W}^{-1}\mathbf{b} \quad (4.85)$$

gelöst. Es ergibt sich dann mit

$$\mathbf{M} = \mathbf{W}\mathbf{W}^T \quad (4.86)$$

der Algorithmus in Tabelle 4.6, siehe auch [108]. Aufgrund der Wurzel zur Berechnung von  $\beta$  muß die Vorkonditionierungsmatrix positiv definit sein; KLAAS, NIEKAMP und STEIN haben in [78] den Algorithmus auch für indefinite Vorkonditionierungsmatrizen unter Verwendung komplexer Vektoren umformuliert. Im Allgemeinen haben sich die in Abschnitt 4.2.2 beschriebenen Verfahren zur Lösung der angestrebten Problemklassen aus

---

<p><i>Phase I (Initialisierung)</i></p> $\mathbf{x}_0, \mathbf{r}_1 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \mathbf{t}_1 = \mathbf{M}^{-1}\mathbf{r}_1,$ $\beta_1 = \sqrt{\mathbf{r}_1^T \mathbf{t}_1}, \mathbf{q}_1 = \frac{\mathbf{r}_1}{\beta_1},$ $\mathbf{q}_0 = \mathbf{c}_0 = \mathbf{d}_0 = \mathbf{0}, \delta_0 = 0$ <p><i>Phase II (Tridiagonalmatrix <math>\mathbf{T}_j</math>)</i></p> <p>für <math>j = 1, 2, \dots, n</math></p> $\mathbf{u}_j = \frac{\mathbf{t}_j}{\beta_j}$ $\mathbf{z}_j = \mathbf{A}\mathbf{u}_j$ $\alpha_j = \mathbf{u}_j^T \mathbf{z}_j$ $\mathbf{r}_{j+1} = \mathbf{z}_j - \alpha_j \mathbf{q}_j - \beta_j \mathbf{q}_{j-1}$ $\mathbf{t}_{j+1} = \mathbf{M}^{-1}\mathbf{r}_{j+1}$ $\beta_{j+1} = \sqrt{\mathbf{r}_{j+1}^T \mathbf{t}_{j+1}}$ $\mathbf{q}_{j+1} = \frac{\mathbf{r}_{j+1}}{\beta_{j+1}}$	<p><i>Phase III (<math>\mathbf{LDL}^T</math>-Faktorisierung von <math>\mathbf{T}_j</math>)</i></p> $d_j = \alpha_j - \delta_j^2 d_{j-1}$ $\delta_{j+1} = \frac{\beta_{j+1}}{d_j}$ <p><i>Phase IV (Rückwärts/Vorwärtselimination)</i></p> $\zeta_j = -\frac{\delta_j d_{j-1}}{d_j}$ $(\zeta_1 = \frac{\beta_1}{d_1})$ $\mathbf{c}_j = \mathbf{u}_j - \delta_j \mathbf{c}_{j-1}$ $\mathbf{x}_j = \mathbf{x}_{j-1} + \zeta_j \mathbf{c}_j$ <p><i>Phase V</i></p> <p>stop, falls <math>\frac{\beta_{j+1}\zeta_j}{\ \mathbf{b}\ } &lt; \varepsilon</math></p>
--	--

---

Tabelle 4.6: Vorkonditionierter Lanczos-Algorithmus

der Strukturmechanik jedoch als ausreichend erwiesen, sodaß üblicherweise nicht die Notwendigkeit des Einsatzes des in [78] beschriebenen Algorithmus besteht. Darüber hinaus kann auch der QMR-Algorithmus (Abschnitt 4.2.4) genutzt werden, falls indefinite Matrizen zur Vorkonditionierung eingesetzt werden sollen.

Ein von PAIGE und SAUNDERS in [104] behandeltes Problem des Lanczos-Algorithmus in Tabelle 4.6 für indefinite Matrizen stellt die  $LDL^T$ -Zerlegung dar, die nicht notwendigerweise existiert. Sie schlagen daher den Einsatz der stabileren  $LQ$ -Zerlegung vor, die etwas aufwendiger ist. Bei Berechnungen mit dem hier beschriebenen Algorithmus wurden solche Probleme jedoch bei allen betrachteten Beispielen sehr unterschiedlichen Aufbaus nicht beobachtet, sodaß aus Effizienzgründen der Einsatz der  $LDL^T$ -Zerlegung vorgezogen wurde.

#### 4.2.4 Das QMR-Verfahren

Von FREUND und NACHTIGAL [44] wurde das QMR-Verfahren (Quasi minimal residual) für nicht-hermitische Systeme entwickelt und in [43] bzw. [45] für symmetrische, stark indefinite Matrizen formuliert. Dieses Verfahren läßt – im Gegensatz zu anderen Krylov-Unterraumverfahren – auch indefinite Vorkonditionierungsmatrizen zu, was insbesondere bei indefiniten Koeffizientenmatrizen sinnvoll erscheint, da dann eine indefinite Matrix wiederum durch eine indefinite Matrix approximiert wird. Diese Eigenschaft macht das Verfahren auch für die hier betrachteten Probleme interessant, obwohl die bei nichtlinearen Finiten Element Berechnung auftretenden Steifigkeitsmatrizen nur wenige negative Eigenwerte aufweisen. Wie in Abschnitt 4.4.3 jedoch noch beschrieben wird, sind indefinite Vorkonditionierungsmatrizen in besonderen Fällen aus Effizienzgründen wünschenswert.

Beim QMR-Verfahren handelt es sich wiederum um ein Krylov-Unterraumverfahren, d.h. die Näherung in der  $j$ -ten Iteration läßt sich wie folgt darstellen:

$$\mathbf{x}_j = \mathbf{x}_0 + \mathcal{K}_j(\mathbf{A}, \mathbf{r}_0), \quad j = 1, 2, \dots \quad (4.87)$$

Die Lanczos-Vektoren  $\mathbf{v}_i$ ,  $i = 1 \dots j$  spannen dabei den Krylov-Unterraum  $\mathcal{K}_j(\mathbf{A}, \mathbf{r}_0)$  auf, und es gilt

$$\mathbf{x}_j = \mathbf{V}_j \mathbf{z}_j, \quad (4.88)$$

wobei  $\mathbf{z}_j$  ein freier Parametervektor ist. Beim QMR-Verfahren wird er als Lösung von

$$\| \mathbf{f}_j - \mathbf{T}_j \mathbf{z}_j \|_2 = \min_{\mathbf{z}} \| \mathbf{f}_j - \mathbf{T}_j \mathbf{z} \|_2 \quad \text{mit} \quad \mathbf{f}_j = [ \| \mathbf{r}_0 \|_2 \ 0 \ \dots \ 0 ]^T \quad (4.89)$$

gewählt. In der  $j$ -ten QMR-Iteration kann das Residuum dann als

$$\mathbf{b} - \mathbf{A} \mathbf{x}_j = \mathbf{V}_{j+1} (\mathbf{f}_j - \mathbf{T}_j \mathbf{z}_j) \quad (4.90)$$

---

<p><i>Phase I (Initialisierung)</i></p> $\mathbf{x}_0, \mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0, \mathbf{t} = \mathbf{W}_1^{-1}\mathbf{r}_0, \tau_0 = \ \mathbf{t}\ _2$ $\mathbf{q}_0 = \mathbf{W}_2^{-1}\mathbf{t}, \theta_0 = \mathbf{0}, \rho_0 = \mathbf{r}_0^T \mathbf{q}_0$ <p><i>Phase II (Lanczos Tridiagonalisierung)</i></p> <p>für <math>j = 1, 2, \dots</math></p> $\mathbf{t} = \mathbf{A}\mathbf{q}_{j-1}$ $\sigma_{j-1} = \mathbf{q}_{j-1}^T \mathbf{t}$ <p>falls <math>\sigma_{j-1} = 0</math>, stop</p> $\alpha_{j-1} = \frac{\rho_{j-1}}{\sigma_{j-1}}$ $\mathbf{r}_j = \mathbf{r}_{j-1} - \alpha_{j-1}\mathbf{t}$ <p><i>Phase III (QR-Zerlegung)</i></p> $\mathbf{t} = \mathbf{W}_1^{-1}\mathbf{r}_j$ $\theta_j = \frac{\ \mathbf{t}\ _2}{\tau_{j-1}}$	$c_j = \frac{1}{\sqrt{1+\theta_j^2}}$ $\tau_j = \tau_{j-1}\theta_j c_j$ $\mathbf{d}_j = c_j^2 \theta_{j-1}^2 \mathbf{d}_{j-1} + c_j^2 \alpha_{j-1} \mathbf{q}_{j-1}$ <p><i>Phase IV (Neue Iterierte)</i></p> $\mathbf{x}_j = \mathbf{x}_{j-1} + \mathbf{d}_j$ <p>falls <math>\rho_{j-1} = 0</math> stop</p> $\mathbf{u}_j = \mathbf{W}_2^{-1}\mathbf{t}$ $\rho_j = \mathbf{r}_j^T \mathbf{u}_j$ $\beta_j = \frac{\rho_j}{\rho_{j-1}}$ $\mathbf{q}_j = \mathbf{u}_j + \beta_j \mathbf{q}_{j-1}$ <p><i>Phase V (Abbruch)</i></p> <p>stop falls <math>\rho_j &lt; \varepsilon</math></p>
---	---

---

Tabelle 4.7: Vorkonditionierter QMR-Algorithmus

angegeben werden, d.h. die QMR-Iteration wird als Minimierung des zweiten Faktors in (4.90) charakterisiert, was die Namensgebung des Verfahrens motiviert.

Die Vorkonditionierung erfolgt mit der symmetrischen Vorkonditionierungsmatrix

$$\mathbf{M} = \mathbf{W}_1 \mathbf{W}_2 = \mathbf{W}_1^T \mathbf{W}_2^T = \mathbf{M}^T \quad (4.91)$$

und es kann der Algorithmus in Tabelle 4.7 formuliert werden (siehe [43]).

#### 4.2.5 Abbruchkriterium

Iterative Methoden erstellen eine Sequenz von Näherungslösungen  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_j$ , die zum Lösungsvektor  $\mathbf{x}^*$  konvergiert. Im Laufe dieses Lösungsprozesses muß entschieden werden, wann abgebrochen werden kann, d.h. wann die Lösung als ausreichend genau betrachtet wird. Dazu sollte primär der Fehler

$$\mathbf{e} = \mathbf{x}_j - \mathbf{x}^* \quad (4.92)$$

klein gegenüber den Einträgen in  $\mathbf{x}^*$  sein.

Desweiteren sollte auch nach einer – vorgegebenen – großen Anzahl von Iterationen (z.B. nach  $n$  Iterationen) abgebrochen werden, da dann das Iterationsverfahren entweder versagt hat (Gründe hierfür liegen i.a. in einer falschen mechanischen Modellierung) oder die Lösung bereits mit hoher Genauigkeit erreicht wurde (letzteres im Fall eines zu streng vorgegebenen Abbruchkriteriums). Da der Fehler  $\mathbf{e}$  nicht explizit vorliegt – in diesem Fall

wäre die exakte Lösung bekannt – muß eine Größe gefunden werden, die diesen Fehler gut approximiert und einfach zu ermitteln ist. Wegen

$$\| \mathbf{x}^* - \mathbf{x}_j \|_2 = \| \mathbf{A}^{-1} \mathbf{b} - \mathbf{A}^{-1}(\mathbf{b} - \mathbf{r}_j) \|_2 = \| \mathbf{A}^{-1} \mathbf{r}_j \|_2 \leq \| \mathbf{A}^{-1} \|_2 \cdot \| \mathbf{r}_j \|_2 \quad (4.93)$$

ist die Residuumsnorm  $\| \mathbf{r}_j \|_2$  ein sinnvolles Maß, wenngleich bei großem  $\| \mathbf{A}^{-1} \|_2$  auch eine große Fehlernorm bei kleiner Residuumsnorm auftreten kann. Ein ähnlicher Zusammenhang wird in [74] für die relative Fehler- und Residuumsnorm angegeben.

Darüber hinaus ist es sinnvoll, die Größenordnung der Werte der rechten Seite  $\mathbf{b}$  zu berücksichtigen um so z.B. Einflüsse der Wahl von Einheiten bei der mechanischen Modellierung auf die Genauigkeit der Lösung auszuschließen, indem als Abbruchkriterium gefordert wird:

$$\frac{\| \mathbf{r} \|}{\| \mathbf{b} \|} < \varepsilon. \quad (4.94)$$

Dabei ist  $\| \dots \|$  eine geeignete Norm der rechten Seite, z.B. die euklidische Norm  $\| \dots \|_2$  oder die  $\mathbf{M}^{-1}$ -Norm  $\| \dots \|_{\mathbf{M}^{-1}}$ . Der Parameter  $\varepsilon$  ist vom Anwender problemangepaßt vorzugeben. Werte in der Größenordnung zwischen  $10^{-4} < \varepsilon < 10^{-8}$  haben sich in der Berechnungspraxis bewährt.

Damit kann für das cg-Verfahren das Abbruchkriterium

$$\frac{\sqrt{\gamma_{j+1}}}{\| \mathbf{b} \|_{\mathbf{M}^{-1}}} < \varepsilon \quad \text{bzw.} \quad \frac{\sqrt{\gamma_{j+1}}}{\gamma_0} < \varepsilon \quad (4.95)$$

angegeben werden. Wird als Startnäherung  $\mathbf{x}_0 = \mathbf{0}$  verwendet, sind beide Kriterien identisch.

Analog erhält man für das Lanczos Verfahren

$$\frac{\tau_{j+1} \zeta_j}{\| \mathbf{b} \|_{\mathbf{M}^{-1}}} < \varepsilon \quad \text{bzw.} \quad \frac{\tau_{j+1} \zeta_j}{\tau_0} < \varepsilon \quad (4.96)$$

und für das symmetrische QMR-Verfahren

$$\frac{\tau_j \tau_0 s_1 \dots s_{j-1}}{\gamma_j \| \mathbf{b} \|_{\mathbf{M}^{-1}}} < \varepsilon \quad \text{bzw.} \quad \frac{\tau_j s_1 \dots s_{j-1}}{\gamma_j}, \quad (4.97)$$

siehe hierzu auch [93].

### 4.3 Vergleichsberechnungen für lineare Probleme

In dem folgenden Abschnitt werden die bereitgestellten Verfahren zur Lösung linearer Gleichungssysteme miteinander verglichen. Dabei werden zunächst lineare Probleme aus

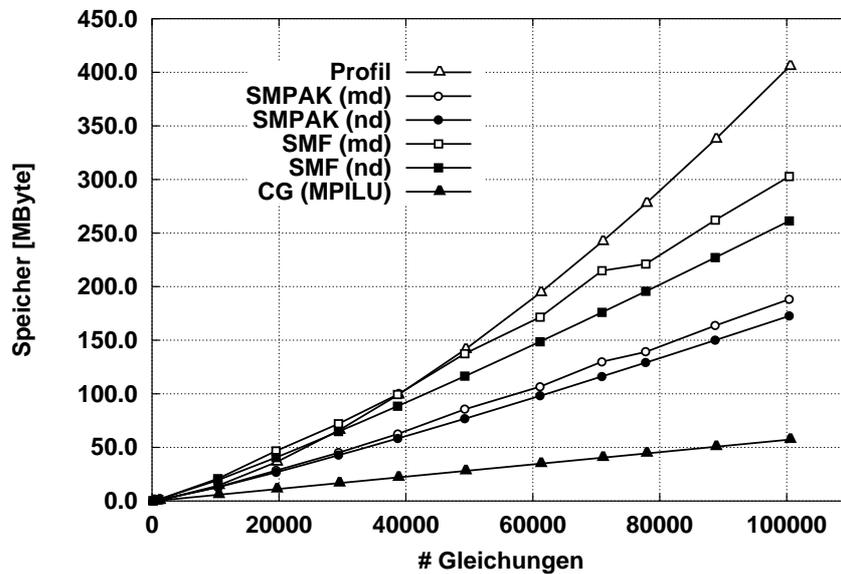


Abbildung 4.3: Speicherbedarf direkter Löser für Torus Beispiel

der Strukturmechanik betrachtet. Bei den in Abschnitt 3 eingeführten nichtlinearen Problemen wird hier lediglich das entsprechende lineare Problem gelöst. Anhand der hieraus gewonnenen Erkenntnisse kann eine Beschränkung der nachfolgenden nichtlinearen Berechnungen auf die effizienteren Verfahren erfolgen.

Als Toleranz beim Abbruchkriterium der iterativen Verfahren wird  $\varepsilon = 1.0 \cdot 10^{-5}$  verwendet. Bei der SSOR-Vorkonditionierung wird stets mit  $\omega = 1.0$  gerechnet, siehe Abschnitt 4.2.2.2. Für die NDILU-Vorkonditionierung wird zur Steuerung des *fill-in*  $\psi = 1.0 \cdot 10^{-6}$  benutzt, was bei den betrachteten Beispielen zu einer Vorkonditionierungsmatrix führt, die etwa doppelt so viele Einträge enthält wie die Koeffizientenmatrix, wodurch eine gute Abstufung der Vorkonditionierung gegenüber den Verfahren mit konstantem *fill-in* gegeben ist. Für Untersuchungen mit variierendem  $\psi$  siehe [93].

#### 4.3.1 Numerischer Vergleich direkter Lösungsstrategien bei wachsender Anzahl der Gleichungen

Das in Abschnitt 4.1 erläuterte Verfahren zur direkten Gleichungslösung soll in diesem Abschnitt anhand eines konkreten Problems mit anderen Verfahren zur direkten Gleichungslösung verglichen werden. Im Vergleich wird ein Profillöser [156] und ein Multifrontallöser [56, 33] sowie ein iterativer cg-Löser verwendet, um wesentliche Unterschiede aufzuzeigen. Besondere Berücksichtigung soll dabei der Einfluß der Hardwarearchitektur auf die Effizienz der jeweiligen Löser finden.

Für das Torus Beispiel aus Abschnitt 3.3 mit regelmäßigem Netz aber – im Gegensatz zu der dort angegebenen Diskretisierung – unterschiedlich feiner Diskretisierung ist in Abbildung 4.3 zuerst der Speicherbedarf zur Lösung des linearen Problems dargestellt.

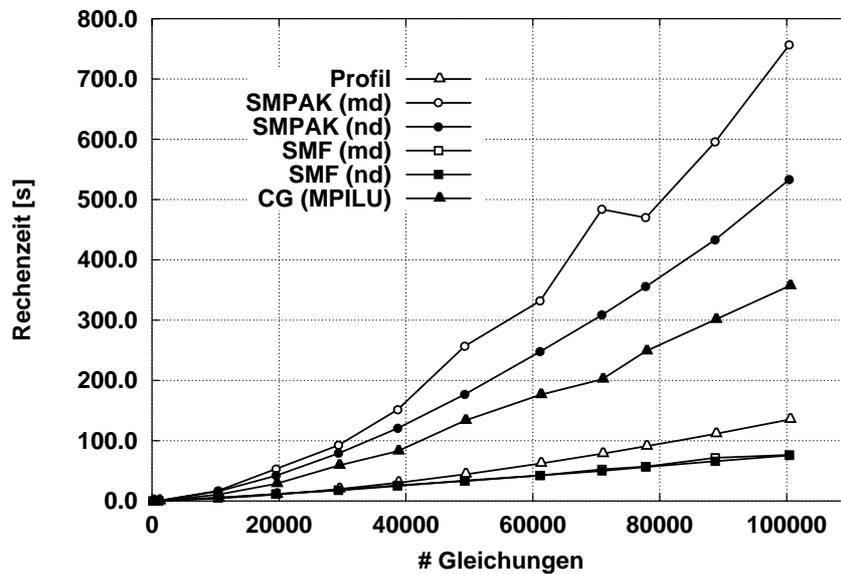


Abbildung 4.4: Rechenzeit auf SNI VPP 300 für Torus Beispiel (lineares Problem)

Deutlich sichtbar ist der hohe Speicherbedarf des Profillösers, der nicht zuletzt durch die Matrixstruktur mit Einträgen, weit von der Diagonalen entfernt, verursacht wird, siehe Abbildung 3.7 (rechts). Für feinere Diskretisierungen ist ein weiteres nichtlineares Ansteigen des Speicherbedarfes zu erwarten. Im Gegensatz hierzu zeigen die auf der *sparse*-Speicherstruktur beruhenden SMPAK<sup>6</sup>-Löser [36] und das Multifrontalverfahren<sup>7</sup> [33] einen geringeren Speicherbedarf, der auch für steigende Anzahlen von Unbekannten eher linear anwächst. Der Speicherbedarf des Multifrontallösers ist aber deutlich größer als der des SMPAK-Lösers, was durch das Abspeichern von dicht besetzten Untermatrizen bei der Faktorisierung verursacht wird, allerdings eine gute Vektorisierung ermöglicht. Auch ist am Speicherbedarf die Qualität der *fill-in* reduzierenden Heuristiken *minimum degree* (md) und *nested dissection* (nd) zu erkennen. Für das regelmäßige Netz des Torus ist hierbei ein Vorteil für die *nested dissection* Strategie zu beobachten. Für unregelmäßige Netze ist dieser Sachverhalt oft umgekehrt. Auffallend ist ein hoher Speicherbedarf bei der *minimum degree* Permutation für das Problem mit ca. 70.000 Gleichungen; sowohl für den Multifrontallöser als auch für den SMPAK-Löser wird hier ungewöhnlich viel Speicher benötigt, was offenbar die Folge einer in diesem Falle schlechten *tie-breaking* Strategie ist. Den mit Abstand geringsten Speicherbedarf weist der iterative cg-Löser auf, hier wurde die MPILU-Vorkonditionierung gewählt, wobei der zweifache Speicherbedarf für die Steifigkeitsmatrix entsteht.

In Abbildung 4.4 ist die Gesamtrechenzeit für das lineare Torus Problem auf einem SNI VPP 300 Vektorrechner dargestellt. Die schnellste Lösung wird mit dem Multifrontalverfahren erzielt. Der Unterschied als Folge der *fill-in* Minimierung ist dabei nicht ausgeprägt. Auch mit dem Profillöser wird eine kurze Rechenzeit erreicht. In beiden Fällen ist dies

<sup>6</sup>Sparse Matrix Package.

<sup>7</sup>Symmetrisches MultiFrontal Verfahren.

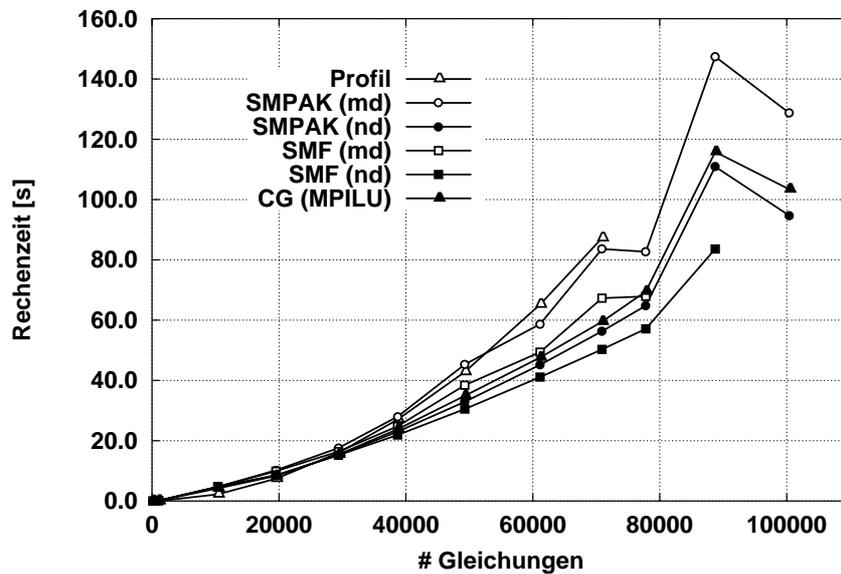


Abbildung 4.5: Rechenzeit auf IBM RS6000 für Torus Beispiel (lineares Problem)

eine Folge der guten Vektorisierbarkeit des Faktorisierungsvorganges, beim Multifrontallöser durch das Bearbeiten der dichtbesetzten Untermatrizen und beim Profillöser durch das Bearbeiten der Spaltenvektoren. Die Effizienz dieser Verfahren wird allerdings mit hohem Speicherbedarf erkauft. Deutlich langsamer erfolgt die Lösung durch den direkten SMPAK-Löser, da die dort erforderliche indirekte Adressierung nur schlecht vektorisierbar ist. Bei wachsender Größe des Gleichungssystems wird dieses Verfahren auf Vektorrechnern immer ineffizienter. Hier ist ein deutlicher Vorteil der *nested dissection* Permutationsstrategie gegenüber der *minimum degree* Permutation für dieses Beispiel zu bemerken. Gut zu erkennen ist auch in dieser Abbildung die schlechte *minimum degree* Permutation für ca. 70.000 Gleichungen, die nicht nur einen erhöhten Speicherbedarf sondern damit einhergehend eine erhöhte Rechenzeit zur Folge hat. Eine effizientere Lösung als mit dem direkten SMPAK-Löser wird auf dem Vektorrechner mit dem iterativen Löser erreicht. Da aber auch dieser mit indirekter Adressierung arbeitet, ist die Lösungszeit deutlich größer als für den Multifrontal- und Profillöser.

Im Gegensatz zu Vektorrechnern wirkt sich die indirekte Adressierung bei RISC-Prozessoren nicht so nachteilig aus, weshalb bei den Rechenzeiten auf einer IBM RS6000 Workstation keine Überlegenheit der Multifrontal- und Profillöser zu erwarten ist. Die bekannten Nachteile des hohen Speicherbedarfes bleiben natürlich bestehen. Für diesen Rechnertyp sind die Rechenzeiten für das gleiche Beispiel in Abbildung 4.5 angegeben. Der Profillöser ist dort das ineffizienteste Verfahren und wegen des hohen Speicherbedarfes kann damit das Problem nur mit maximal 70.000 Unbekannten gelöst werden, da auf dem Rechner nur bis zu 250 Megabyte Hauptspeicher zur Verfügung stehen. Die effizienteste Lösung wird mit dem Multifrontallöser bzw. dem SMPAK-Löser unter Verwendung der *nested dissection* Permutationsstrategie erzielt. Mit der *minimum degree* Permutation ist dagegen eine deutlich höhere Rechenzeit zur Lösung des Problems erforderlich. Wegen des rela-

tiv hohen Speicherbedarfs des Multifrontallösers können auch hier nur Lösungen für bis zu 80.000 (*minimum degree*) bzw. 90.000 (*nested dissection*) Gleichungen berechnet werden. Mit dem SMPAK-Löser hingegen ist eine Lösung auch für über 100.000 Gleichungen mit dieser Hardware möglich und die erreichten Rechenzeiten sind gegenüber dem Multifrontalverfahren nicht wesentlich höher. Auch auf diesem Rechner zeigt sich die schlechte Qualität der *minimum degree* Permutation des Gleichungssystems mit 70.000 Gleichungen, nun auch beim Multifrontallöser. Es sei bemerkt, daß bei anderen Problemen der SMPAK-Löser zum Teil auch effizienter in Bezug auf die Rechenzeit sein kann. Der iterative Löser zeigt für das Torus Beispiel auch eine gute Effizienz bei minimalem Speicherbedarf. Auffallend in Abbildung 4.5 sind übermäßig hohe Rechenzeiten bei dem Problem, das mit ca. 90.000 Gleichungen diskretisiert wurde – unabhängig vom Typ des Lösers. Da diese Zeiten reproduzierbar sind, liegt die Vermutung nahe, daß hier eine ungünstige Vektorlänge vorliegt und die hohe Rechenzeit (höher als für das nächstgrößere Problem!) durch häufige *cache faults* verursacht wird.

Zusammenfassend läßt sich (auch unter Berücksichtigung von hier nicht dokumentierten Beispielen) sagen, daß die Multifrontal- und Profillöser auf Vektorrechnern am effizientesten sind, sofern genügend Hauptspeicher verfügbar ist. Dies ist eine direkte Folge der jeweiligen Algorithmen, die auf der Bearbeitung dicht abgespeicherter Vektoren bzw. Matrizen beruhen und dadurch auf diesen Rechnertyp abgestimmt sind. Auf RISC-Prozessoren sind *sparse*-Löser effizient insbesondere wegen des stark reduzierten Speicherbedarfs. Die Permutationsstrategie hat bei den *sparse* Lösern einen großen Einfluß auf den Speicherbedarf und die Rechenzeit; im Allgemeinen ist bei regelmäßigen Netzen die *nested dissection* Permutation vorteilhafter, bei unregelmäßigen Netzen, die durch Adaptivität oder automatische Netzgeneratoren entstehen, oft die *minimum degree* Strategie. Iterative Löser sind vor allem wegen des sehr geringen Speicherbedarfs attraktiv, da selbst größte Probleme auf Rechnern mit nur wenig Hauptspeicher gelöst werden können.

#### 4.3.2 Permutationsstrategien zur Verbesserung der Qualität von unvollständigen Faktorisierungen

Bei den in Abschnitt 4.2.2.3 beschriebenen unvollständigen Faktorisierungen wird *fill-in* vernachlässigt. Denkt man sich einen vollständigen Faktorisierungsvorgang, so hat der bei einem ersten Durchgang der äußersten Schleife in Tabelle 4.1 vernachlässigte *fill-in* Einfluß auf die numerischen Resultate der nachfolgenden Schleifendurchläufe. Ändert man die Reihenfolge der abgearbeiteten Gleichungen, permutiert man also die Koeffizientenmatrix, so werden numerisch unterschiedliche Vorkonditionierungsmatrizen berechnet. Es stellt sich die Frage, wie eine Matrix permutiert werden muß, um die Qualität der Vorkonditionierungsmatrix zu verbessern. Von direkten Lösungsmethoden sind für Band- bzw. Profillöser Band- und Profilminderungsstrategien, für *sparse*-Löser die *minimum degree* und *nested dissection* Permutationsstrategien bekannt. Bei den Band- und Profilminderungsstrategien wird die Matrix "dichter" gepackt. Da *fill-in* nur innerhalb des Profils einer Matrix entsteht, wird die Menge des *fill-in* reduziert. Bei der unvollständigen Zerlegung wird also weniger *fill-in* vernachlässigt, was sich positiv auf die Vorkonditionierungsmatrix auswirken könnte. Mit derselben Argumentation könnte eine Verbesserung der Vorkondi-

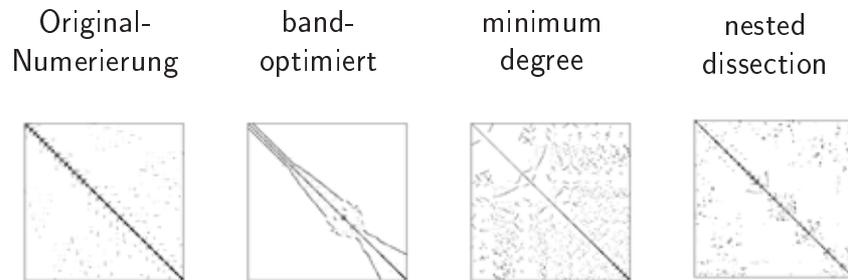


Abbildung 4.6: Matrixstrukturen für verschiedene Permutationen beim Rohrkreuz-Beispiel

tionierungsmatrix durch eine *minimum degree* oder *nested dissection* Permutation erwartet werden, da auch hier die Menge des *fill-in* bei einer vollständigen Faktorisierung reduziert wird. Jedoch stellt sich in beiden Fällen die Frage, ob der vernachlässigte *fill-in* dann nicht von größerem Einfluß ist.

Für das Rohrkreuz Beispiel sind in Abbildung 4.6 die Matrixstrukturen für die erwähnten Permutationen dargestellt. Das Finite Element Netz wurde automatisch generiert, weshalb eine unregelmäßige Matrixstruktur entsteht, deren Bandbreite mit einem Bandoptimierungsalgorithmus stark reduziert werden kann.

Permutation	# der Iterationen	Lösungszeit [s]
Original-Numerierung	593	252.4
Bandoptimiert	494	212.9
minimum degree	705	306.5
nested dissection	660	280.6

Tabelle 4.8: Iterationszahlen und Lösungszeit für das lineare Problem (CG-MPILU)

Für diese Permutationen sind die Iterationszahlen und die Rechenzeit für das lineare Problem mit der MPILU-Vorkonditionierung in Tabelle 4.8 angegeben. Die effizienteste Lösung wird mit der bandoptimierten Matrixstruktur erzielt. *Minimum degree* oder *nested dissection* haben keinen positiven Einfluß auf die Iterationszahlen. Hier wird zwar ebenfalls die Anzahl der vernachlässigten *fill-in* Einträge reduziert, aber offensichtlich ist diese Vernachlässigung von einer anderen Qualität.

Untersuchungen an anderen Problemen haben gezeigt, daß bei einer sorgfältigen, das Matrixprofil minimierenden Numerierung des Netzes keine Permutation notwendig ist, bei automatisch generierten bzw. adaptiven Netzen jedoch eine Band- oder Profilloptimierung auch für unvollständige Faktorisierungen sehr ratsam sein kann.

Blöcke	Speicher [MByte]	unpermutiert		permutiert	
		# Iterationen	$t_{LGS}$ [s]	# Iterationen	$t_{LGS}$ [s]
$ x y z \varphi_1 \varphi_2 $	1.9	422	25.0	422	25.7
$ x,y z \varphi_1 \varphi_2 $	2.0	422	27.1	422	27.5
$ x y,z \varphi_1 \varphi_2 $	2.0	422	27.2	422	27.8
$ x y z \varphi_1,\varphi_2 $	2.0	420	27.2	420	27.7
$ x,y,z \varphi_1,\varphi_2 $	2.7	420	33.1	420	33.4
$ x,y,z,\varphi_1,\varphi_2 $	2.8	159	16.9	159	17.2

Tabelle 4.9: Iterationszahlen und Rechenzeit für die flache Zylinderschale

### 4.3.3 CG-Verfahren mit blockweiser unvollständiger $LDL^T$ -Zerlegung (Block-MPILU)

Die Vorkonditionierung mit der blockweisen unvollständigen  $LDL^T$ -Zerlegung erlaubt eine problemangepasste Wahl der Blöcke. Wie auch schon in [115] beschrieben, bietet sich für Volumenelemente eine Wahl wie in Gleichung (4.50) angegeben an. Bei Schalenproblemen unter Verwendung des degenerierten Schalenkonzeptes ist eine Trennung von Verschiebungs- und Verdrehungsfreiheitsgraden naheliegend. Auch das Trennen von Vorzugsrichtungen, die eventuell im mechanischen Modell vorhanden sind, stellt eine möglicherweise sinnvolle Wahl dar.

In Abbildung 4.7 sind die sich ergebenden Matrixstrukturen für die flache Zylinderschale mit der Blockaufteilung  $|x,y|z|\varphi_1|\varphi_2|$  nach Gleichung (4.51) dargestellt. Die Blockaufteilung ist bei der permutierten Steifigkeitsmatrix deutlich zu erkennen, ebenso bei der Vorkonditionierungsmatrix; die unpermutierte, gefilterte Vorkonditionierungsmatrix behält die Struktur der Steifigkeitsmatrix bei, ist aber erkennbar dünner besetzt.

Für das Durchschlagproblem der flachen Zylinderschale sind in Tabelle 4.9 die gewählte Blockaufteilung, der benötigte Speicher, die Iterationsanzahl und die Rechenzeit zur Lösung des linearen Gleichungssystems angegeben. Es wurde sowohl mit der permutierten Matrix als auch mit der unpermutierten Steifigkeitsmatrix und der gefilterten Vorkondi-

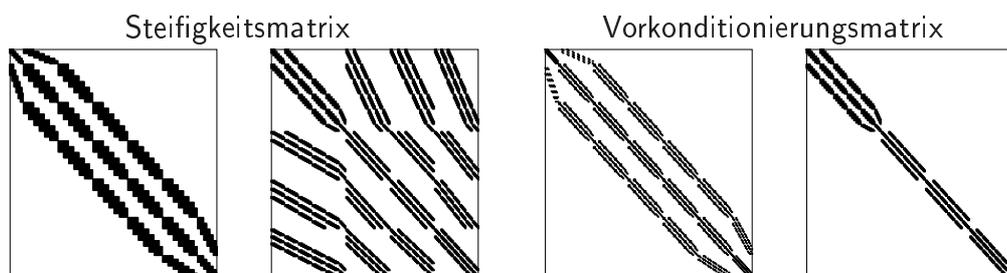


Abbildung 4.7: Matrixstrukturen der unpermutierten/permutierten Steifigkeitsmatrix und der zugehörigen Vorkonditionierungsmatrizen für die symmetrische flache Zylinderschale

Blöcke	Speicher [MByte]	unpermutiert		permutiert	
		# Iterationen	$t_{\text{LGS}}$ [s]	# Iterationen	$t_{\text{LGS}}$ [s]
$ x_u y_u z_u x_o y_o z_o $	4.3	1227	109.1	1229	111.6
$ x_u, x_o y_u, y_o z_u, z_o $	4.6	135	13.7	135	13.7
$ x_u, x_o, y_u, y_o z_u, z_o $	4.9	133	16.0	133	16.0
$ x_u, x_o y_u, y_o, z_u, z_o $	4.9	138	16.7	138	16.8
$ x_u, y_u, z_u, x_o, y_o, z_o $	5.6	136	22.4	136	22.3

Tabelle 4.10: Iterationszahlen und Rechenzeit für das Rohrkreuz

tionierungsmatrix gerechnet. Der benötigte Speicher ist für beide Varianten gleich, die gefilterte Variante führt zu einer unwesentlich schnelleren Lösung, die Zeitersparnis bei der gesamten Berechnung ist etwas größer, da zur Permutation ein größerer Aufwand zur Indexberechnung erforderlich ist. Die Qualität der Vorkonditionierung ist aber von der Permutation unbeeinflusst, die Iterationszahlen sind gleich. Bei den gewählten Blockeinteilungen sind kaum Unterschiede in den benötigten Iterationen zu beobachten. Eine Ausnahme stellt die Einteilung aller Freiheitsgrade in einen Block dar, die der unvollständigen  $LDL^T$ -Zerlegung entspricht, mit der deutlich weniger Iterationen benötigt werden. Die geringste Speicheranforderung wird bei der Entkopplung aller Freiheitsgrade erreicht, die benötigte Zeit pro Iteration ist ebenso minimal, wodurch die effizienteste Lösung erzielt wird. Aus Vorzugsrichtungen der mechanischen Struktur – Membrananteil in  $xy$ -Ebene, Biegeanteil in  $z$ -Richtung – ist bei einer entsprechenden Blockeinteilung kein Vorteil zu ziehen, erst durch die Bildung eines Blockes, der alle Freiheitsgrade enthält, kann eine effizientere Lösung erzielt werden.

In Tabelle 4.10 sind die Resultate für das Rohrkreuz Beispiel aus Abschnitt 3.4 unter Verwendung von “Solid-Shell” Elementen aus [60], die unter ausschließlicher Verwendung von Verschiebungsfreiheitsgraden der Schalenoberseite und -unterseite formuliert sind, dargestellt. Bei der Separation aller Freiheitsgrade ergibt sich eine 10 mal höhere Iterationszahl, als bei den anderen Varianten, bei denen jeweils zu einer Verschiebungsrichtung gehörende Freiheitsgrade in einem Block belassen wurden. Bei den letzteren Varianten ist kein signifikanter Unterschied zu beobachten, die Anzahl der benötigten Iterationen ist praktisch gleich, abweichend ist nur als Folge der unterschiedlichen Blockgröße der jeweils benötigte Speicher und damit die Rechenzeit zum Vorwärts- und Rückwärtseinsetzen sowie zum Lösen des Gleichungssystems. Die effizienteste Lösung erfolgt demnach mit der zweiten Variante. Auch die Bildung eines einzigen Blockes für eine gewöhnliche unvollständige Zerlegung bewirkt hier keine weitere Reduktion der Iterationszahlen.

#### 4.3.4 Vergleich zwischen direkten und iterativen Verfahren bei unterschiedlichen Problemklassen

In diesem Abschnitt sollen direkte und iterative Verfahren einander gegenübergestellt werden. Von besonderem Interesse soll dabei das Verhalten dieser Lösertypen bei unterschied-

lichen Problemklassen sein.

Zunächst wird ein Kontinuumsproblem, nämlich der Gummiblock von Abschnitt 3.5 betrachtet. Es wird auf einer IBM RS6000 Workstation unter Verwendung des direkten SMPAK-Lösers und des iterativen cg-Lösers gerechnet. In Abbildung 4.8 sind die benötigten Rechenzeiten und der Speicherbedarf dargestellt. Der direkte Löser benötigt deutlich mehr Speicher (bis zu über 10-facher Bedarf) als der iterative. Dabei wird für die *nested dissection* Permutation weniger Speicher benötigt als für die *minimum degree* Permutation. Für die *nested dissection* Strategie ergibt sich dann auch eine deutlich geringere Rechenzeit zur Lösung des Problems. Mit wenigen Ausnahmen wird zur iterativen Lösung des Problems nicht nur sehr viel weniger Speicher benötigt, sondern zudem auch drastisch weniger Rechenzeit. Lediglich für das NDILU-vorkonditionierte cg-Verfahren wird mehr Rechenzeit benötigt als für die effizienteste direkte Lösung. Hier ist die aufwendige Erstellung der Vorkonditionierung mit den dazu notwendigen Indexberechnungen Ursache

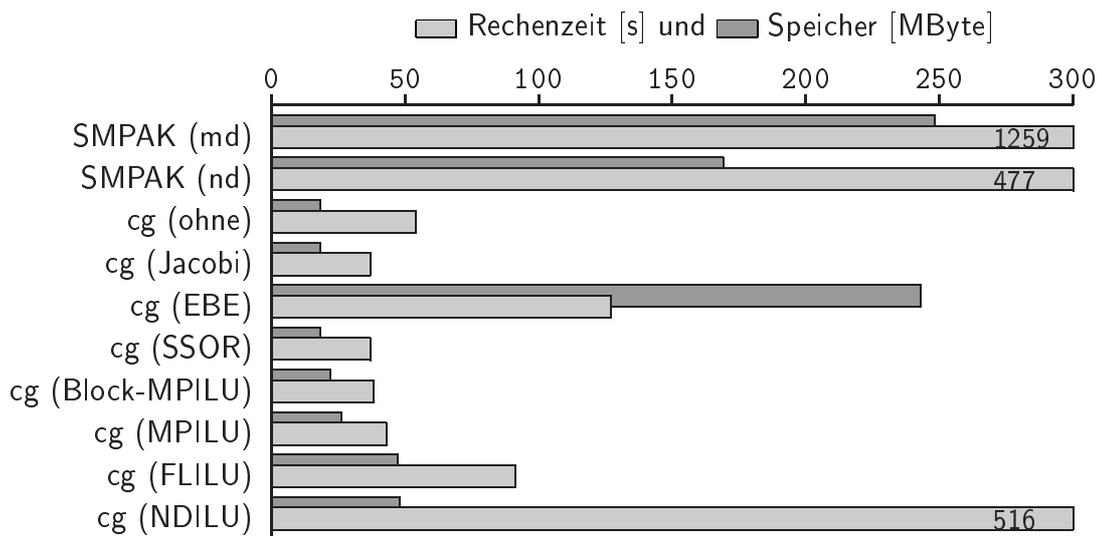


Abbildung 4.8: Rechenzeit und Speicherbedarf auf IBM RS6000 für den Gummiblock

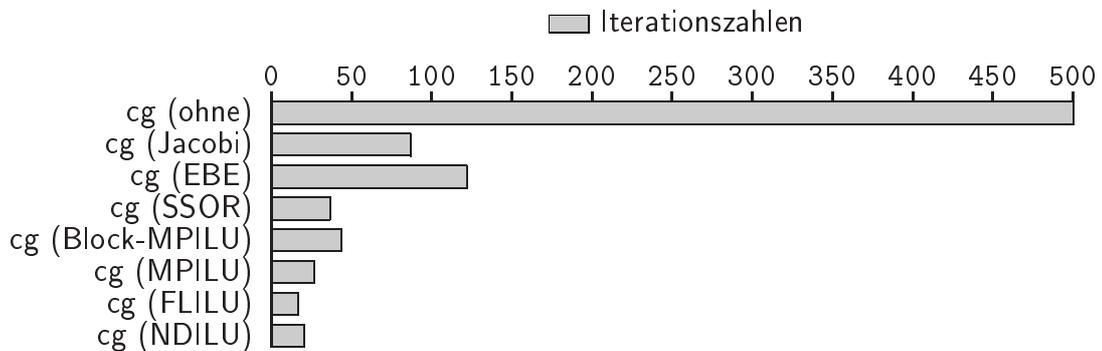


Abbildung 4.9: Iterationszahlen für den Gummiblock

der langen Berechnungsdauer. Die Anzahl der Iterationen ist hingegen gering, was die Qualität der NDILU-Vorkonditionierung verdeutlicht, siehe Abbildung 4.9. Die effizientesten Lösungen werden jedoch mit den einfachen Jacobi-, SSOR- und Block-MPILU-Vorkonditionierungen erzielt. Für die Block-MPILU-Vorkonditionierung wurden alle drei translatorischen Freiheitsgrade jedes Knotens voneinander entkoppelt. Bei diesem Problem lohnt sich der Einsatz einer aufwendigen Vorkonditionierung nicht, was insbesondere an der starken Verringerung der Iterationszahlen bei der Jacobi-Vorkonditionierung gesehen werden kann. Diese kann zwar weiter reduziert werden, z.B. mit der MPILU-, FLILU- oder NDILU-Vorkonditionierung; Letzteres hat jedoch eine nachteilige Wirkung auf die Effizienz, zudem wird mehr Speicherplatz benötigt. Bei der EBE-Vorkonditionierung liegen die Gründe für die auffallend hohe Rechenzeit bei einer nicht optimalen Implementierung und in der Tatsache, daß bei diesem Beispiel mehr Iterationen als z.B. für die Jacobi-Vorkonditionierung notwendig sind. Der hohe Speicherbedarf erklärt sich mit der Abspeicherung der faktorisierten Elementsteifigkeitsmatrizen. Die beste Vorkonditionierung, mit der die geringste Anzahl an Iterationen erreicht wird, ist in diesem Fall die FLILU-Vorkonditionierung.

In Abbildung 4.10 sind die Rechenzeit und der Speicherbedarf für das Rohrkreuz Beispiel aufgetragen. Hier ist eine geänderte Situation verglichen mit dem vorigen Problem erkennbar. Für die direkten Löser wird zwar immer noch mehr Speicherplatz als für die iterativen benötigt, aber der Unterschied ist deutlich geringer, da bei den Schalenstrukturen die Steifigkeitsmatrix dünner besetzt ist als bei den Kontinuumsproblemen. Die Rechenzeit für die direkten Löser ist jedoch nicht mehr so deutlich größer als für die iterativen Löser. Für die beiden *fill-in* minimierenden Strategien verhält sich der Löser fast gleich. Ohne Vorkonditionierung dauerte die iterative Lösung sogar länger als für die direkten Löser. Die längsten Zeiten benötigen hier die EBE- und NDILU-vorkonditionierten cg-Löser aus den oben geschilderten Gründen. Die Iterationszahl für die EBE-Vorkonditionierung ist

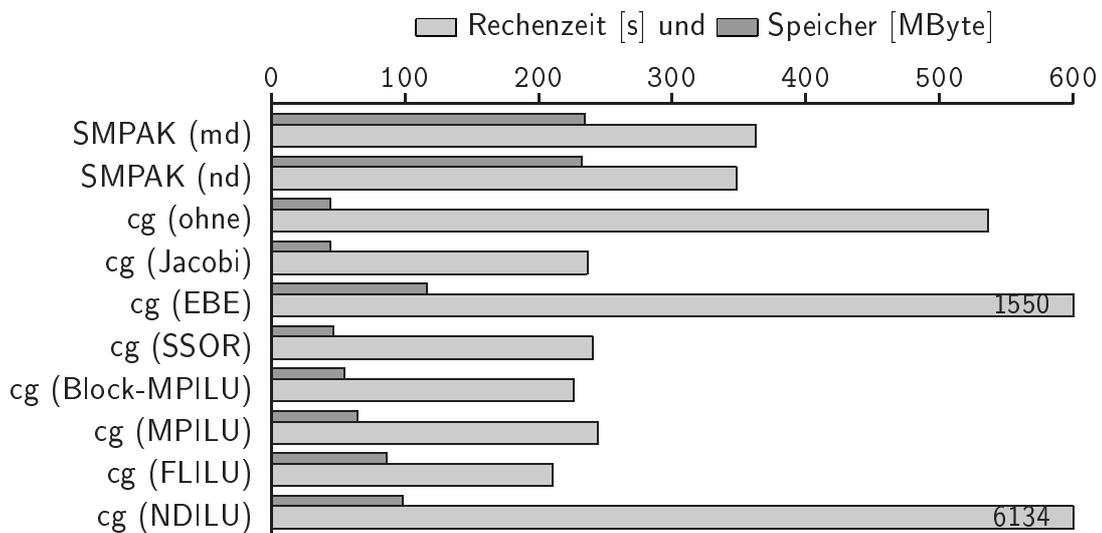


Abbildung 4.10: Rechenzeit und Speicherbedarf auf IBM RS6000 für das Rohrkreuz

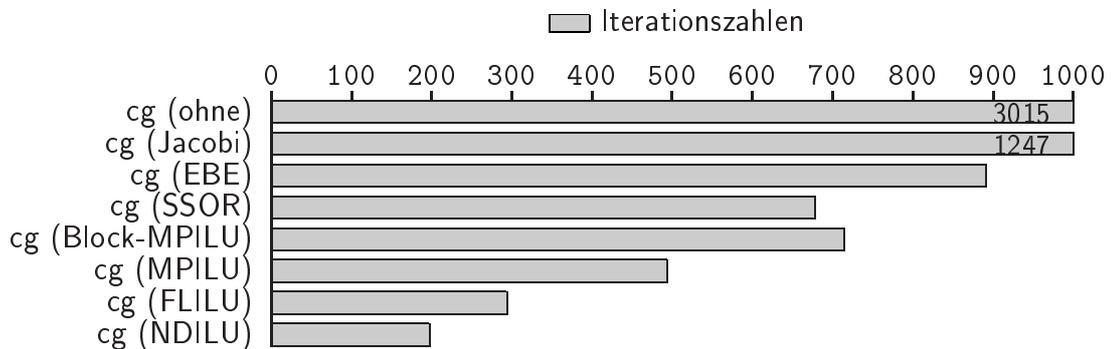


Abbildung 4.11: Iterationszahlen für das Rohrkreuz

hier aber geringer als für die Jacobi-Vorkonditionierung, siehe Abbildung 4.11. Dieser Abbildung kann auch entnommen werden, daß die Iterationszahlen mit wachsender Qualität der Vorkonditionierung abnehmen. Dieses Verhalten kann jedoch nicht auf die Rechenzeit übertragen werden, diese sind ähnlich für Jacobi, SSOR, Block-MPILU, MPILU und FLILU, wobei die schnellste Lösung für dieses Problem mit der FLILU Vorkonditionierung erzielt wird.

Insgesamt läßt sich feststellen, daß bei Kontinuumsproblemen iterative Löser den direkten Verfahren deutlich überlegen sind, was sowohl den Speicherbedarf als auch die Rechenzeit betrifft. Dies ist für schlechter konditionierten Schalenprobleme nicht immer der Fall, wie auch den Abbildungen 4.4 und 4.5 entnommen werden kann. Trotz ihres hohen Speicherbedarfes sind hier direkte Verfahren konkurrenzfähig und in vielen Fällen auch vorteilhaft. Bei regelmäßigen Netzen zeigt sich meist ein Vorteil der *nested dissection* Permutationsstrategie, bei unregelmäßigen Netzen wird oft mit der *minimum degree* Permutation die effizientere Lösung erzielt. Bei den iterativen Lösern geht mit einer aufwendigen Vorkonditionierung eine Senkung der Iterationszahl einher, was sich aber nicht immer in der Rechenzeit widerspiegelt, da der Aufwand zur Erstellung des Vorkonditionierers nicht immer durch die Einsparung bei den Iterationen aufgewogen wird. Insbesondere bei Kontinuumsproblemen ist eine einfache Vorkonditionierung, wie z.B. Jacobi- oder SSOR-Vorkonditionierung durchaus ausreichend, um eine effiziente Lösung bei minimalem Speicheraufwand zu erhalten. Die Wahl des Vorkonditionierers bei Schalenproblemen wird in erster Linie durch den verfügbaren Speicher bestimmt. Die MPILU- oder Block-MPILU-Vorkonditionierung hat sich hier als sehr robust gezeigt; die FLILU-Vorkonditionierung ist letzteren auch bei Schalenproblemen nicht immer überlegen, sodaß im Einzelfall die beste Vorkonditionierung mittels Testrechnungen bestimmt werden muß.

#### 4.4 Vergleichsuntersuchungen nichtlinearer Probleme

Bei den hier gezeigten Vergleichsuntersuchungen nichtlinearer Probleme können Folgerungen aus den Vergleichsberechnungen linearer Probleme bereits berücksichtigt werden. So wird z.B. nicht mehr ohne Vorkonditionierung gerechnet, da sich dies in keinem Falle

als vorteilhaft erwiesen hat. Ebenso wird die EBE-Vorkonditionierung nicht weiter betrachtet. Bei der Block-MPILU-Vorkonditionierung werden immer alle Freiheitsgrade separiert, außer bei Verwendung von “Solid Shell” Elementen. Dort werden die Freiheitsgrade der Schalenober- und Unterseite jeweils in einem Block belassen. Bei nicht von vornherein sorgfältig nummerierten Netzen wird ein Profilminimierungsalgorithmus vorgeschaltet. Ansonsten gelten weiterhin bezüglich der Parameterwahl die in Abschnitt 4.3 gemachten Angaben.

#### 4.4.1 Vorkonditionierungsstrategien bei nichtlinearen Berechnungen

In diesem Abschnitt soll die Frage des Aufdatierens der Vorkonditionierungsmatrix bei nichtlinearen Berechnungen diskutiert werden. Es werden drei Varianten erörtert:

- (a) Für jede Steifigkeitsmatrix wird eine neue Vorkonditionierungsmatrix berechnet.
- (b) In jedem Newton Schritt wird eine neue Vorkonditionierungsmatrix berechnet.
- (c) Es wird immer die zu Beginn der Berechnung aufgestellte Vorkonditionierungsmatrix genutzt.

Bei einfachen Vorkonditionierungsstrategien wie Jacobi- und SSOR-Vorkonditionierung gibt es keinen Grund, nicht für jede Steifigkeitsmatrix eine neue Vorkonditionierungsmatrix zu berechnen. Für SSOR mit dem Eisenstat-Trick ist die Neuberechnung sogar erforderlich, da die Steifigkeitsmatrix explizit skaliert werden muß.

Für andere, aufwendigere Vorkonditionierungsstrategien kann es aber sinnvoll sein, eine bereits berechnete Vorkonditionierungsmatrix wiederzuverwenden. Dies ist insbesondere dann erfolgreich, wenn sich die Steifigkeitsmatrix nicht zu stark verändert, d.h. bei nur mäßiger Nichtlinearität im Lastschritt bzw. bei kleinen Schrittweiten. Dabei sind einige Varianten denkbar. So könnte eine Vorkonditionierungsmatrix für  $k$  lineare Gleichungssysteme verwendet werden, wobei  $k$  fest gewählt wird oder wie (b) eine neue Vorkonditionierungsmatrix nur ein Mal pro Lastschritt (im Gleichgewicht) berechnet wird und dann für alle Newton Iterationen verwendet wird. Letzteres ist eine allgemeinere Strategie und zeigte sich als sehr effizient für die damit berechneten Probleme.

Bei nur schwach nichtlinearen Systemen ist es sogar möglich, eine konstante Vorkonditionierung für die Berechnung des gesamten Last-Verformungsverhaltens zu benutzen, d.h. die dem linearen Problem entsprechende Vorkonditionierung (c).

Es hat sich gezeigt, daß bei den berechneten Problemen für Jacobi- und SSOR- Vorkonditionierung (a) die beste Möglichkeit darstellt, wogegen bei den Varianten der unvollständigen Faktorisierung (b) bevorzugt werden sollte. In manchen Fällen führt (a) nicht zur Konvergenz des Newton Verfahrens, wenn aus Steifigkeitsmatrizen, die keinen Gleichgewichtslagen entsprechen, Vorkonditionierungsmatrizen berechnet werden. Methode (c) ist nur bei sehr “teuren” Vorkonditionierungen, wie z.B. NDILU oder der vollständigen Faktorisierung (siehe hierzu Abschnitt 4.4.3) zu empfehlen. Insbesondere in Fällen mit einer Versteifung des Systems führt dieses Verfahren nicht zu effizienten Lösungen.

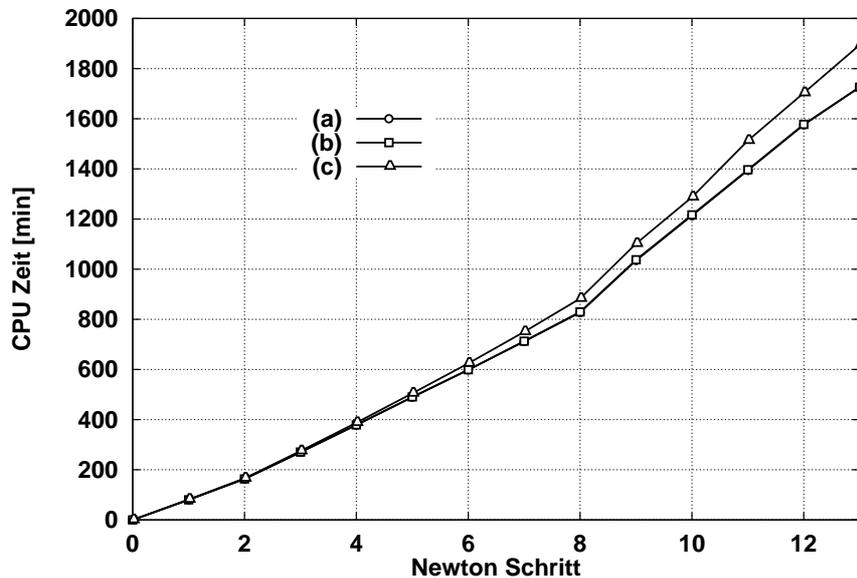


Abbildung 4.12: Rechenzeit für unterschiedliche Aufdatierungsintervalle der Vorkonditionierungsmatrix

In Abbildung 4.12 ist die Gesamtrechenzeit auf einem PC für das nichtlineare Rohrkreuz Problem für die oben beschriebenen Fälle (a) bis (c) für die MPILU-Vorkonditionierung dargestellt. Bei diesem Beispiel ist die Lösungszeit für die Variante (b) fast völlig identisch zu Variante (a). Variante (c) ist nur bis zum Durchschlagspunkt gut funktionsfähig. Danach ist die Abweichung der Vorkonditionierungsmatrix von der aktuellen Steifigkeitsmatrix zu groß, und diese einfache Vorkonditionierung wird deutlich ineffizienter.

#### 4.4.2 Abbruchkriterium bei nichtlinearen Berechnungen

Als Abbruchkriterium für die iterative lineare Gleichungslösung wird eine Größe verwendet, die auf die Norm der rechten Seite bezogen ist (siehe Abschnitt 4.2.5). Dies hat den Vorteil, daß der Abbruch der linearen Gleichungslösung unabhängig von den zur Beschreibung des mechanischen Modells verwendeten Dimensionen ist. So ist es z.B. nicht sinnvoll, bei einem Problem gleicher Steifigkeit mit großer Last  $\mathbf{b}_1$  bis zu der selben absoluten Residuumsnorm zu iterieren wie bei einem Problem mit kleiner Last  $\mathbf{b}_2$ , wobei  $\|\mathbf{b}_2\| \ll \|\mathbf{b}_1\|$ . Die Wahl der Toleranz wird dadurch problemunabhängig.

Bei nichtlinearen Problemen führt dies jedoch zu einer Verschärfung des Abbruchkriteriums zur iterativen linearen Gleichungslösung bei der Annäherung an eine Gleichgewichtslage. Die inneren Knotenkräfte stehen dann mit der äußeren Last nahezu im Gleichgewicht und damit sinkt die Norm der rechten Seite, wodurch das auf diese Norm bezogene Abbruchkriterium des iterativen Gleichungslösers strenger wird. Dies ist aber ein gewünschtes Verhalten, da die linearen Gleichungssysteme zur Bestimmung der Verschiebungskremente innerhalb des Newton Verfahrens vor allem in der Nähe der Gleichgewichtslage mit hoher Genauigkeit gelöst werden müssen. Um dieses Verhalten zu verstärken, wurden in

der Literatur sogenannte *truncated* Newton Methoden vorgeschlagen (siehe z.B. [27] oder [106]), die das Abbruchkriterium  $\varepsilon$  in Abhängigkeit des Fortschritts der Newton Iteration steuern. Auf diese Weise wird der Aufwand zur Gleichungslösung verringert, solange sich das System nicht in der Nähe einer Gleichgewichtslage befindet. Diesen Methoden ist gemeinsam, daß die aktuelle Toleranz  $\varepsilon$  von dem aktuelle Residuum abhängt. In [27] wird z.B.

$$\varepsilon_i = \min(1/i, \|\mathbf{G}\|^t) \quad (4.98)$$

vorgeschlagen, wobei  $i$  die Newton Iterationen zählt,  $\mathbf{G}$  das nichtlineare Residuum ist und  $0 < t \leq 1$  einen zu wählenden Parameter darstellt.

Allerdings wurde hierbei im Zusammenhang mit Bogenlängenverfahren, wie auch in [84], ein negativer Einfluß auf das Konvergenzverhalten des Newton-Raphson Verfahrens beobachtet. Da aber zusätzliche Newton-Raphson Iterationen aufwendiger sind als zusätzliche Iterationen zur Lösung der linearen Gleichungssysteme, wurde vom Einsatz solcher *truncated* Newton Methoden abgesehen.

#### 4.4.3 Vergleich zwischen iterativen und direkten Verfahren

Die in Abschnitt 4.4.1 erzielten Ergebnisse zeigen, daß eine Vorkonditionierungsmatrix zur Lösung mehrerer aufeinanderfolgender Gleichungssysteme effizient verwendet werden kann. Daher können bei nichtlinearen Berechnungen auch Vorkonditionierungsmatrizen vorteilhaft sein, deren Erstellung mit hohem Aufwand verbunden ist, sofern sie anschließend zur Lösung mehrerer Gleichungssysteme herangezogen werden.

So könnte die NDILU-Vorkonditionierung, die sich bei linearen Problemen als nicht effizient erwiesen hatte, dadurch attraktiver werden, daß die aufwendigen Indexoperationen zur Berechnung der Struktur der Vorkonditionierungsmatrix nur einmal durchgeführt werden und diese Matrixstruktur zur Berechnung aller weiteren Vorkonditionierungen beibehalten wird. D.h. auf dieser Matrixstruktur wird eine unvollständige Faktorisierung mit der aktuellen Steifigkeitsmatrix durchgeführt. Diese Vorkonditionierungsmatrix kann dann auch wiederum nur in jedem Lastschritt aufdatiert werden.

Weiterhin kann auch als beste Vorkonditionierung die vollständige Faktorisierung (LU) eingesetzt werden. Nach der ersten Berechnung der Faktoren der Steifigkeitsmatrix können sie zur direkten Lösung genutzt werden. Die nachfolgenden linearen Gleichungssysteme können dann iterativ gelöst werden, wobei die zuvor faktorisierte Matrix zur Vorkonditionierung verwendet werden kann. Zu dieser Vorgehensweise wird allerdings mehr Speicher benötigt als zur rein direkten Gleichungslösung, da zusätzlich Vektoren für das iterative Verfahren notwendig sind. Wegen der hohen Qualität der Vorkonditionierung sind dann nur wenige Iterationen zur Lösung der nachfolgenden Gleichungssysteme zu erwarten. Auch hier kann von der Variante (b) in Abschnitt 4.4.1, d.h. Aufdatieren der Vorkonditionierung in jedem Newton Schritt, Gebrauch gemacht werden. Allerdings wird für das cg und das Lanczos Verfahren eine positiv definite Vorkonditionierung gefordert, was in diesem Fall

nicht ohne hohen Rechenaufwand realisierbar ist, da die Steifigkeitsmatrix und damit auch ihre Faktorisierung bei nichtlinearen Problemen indefinit werden kann. Deshalb ist es bei nichtlinearen Fragestellungen, bei denen vorab nicht bekannt ist, ob die Steifigkeitsmatrix stets positiv definit ist, günstig, das QMR Verfahren aus Abschnitt 4.2.4 zu verwenden, das auch indefinite Vorkonditionierungsmatrizen zuläßt.

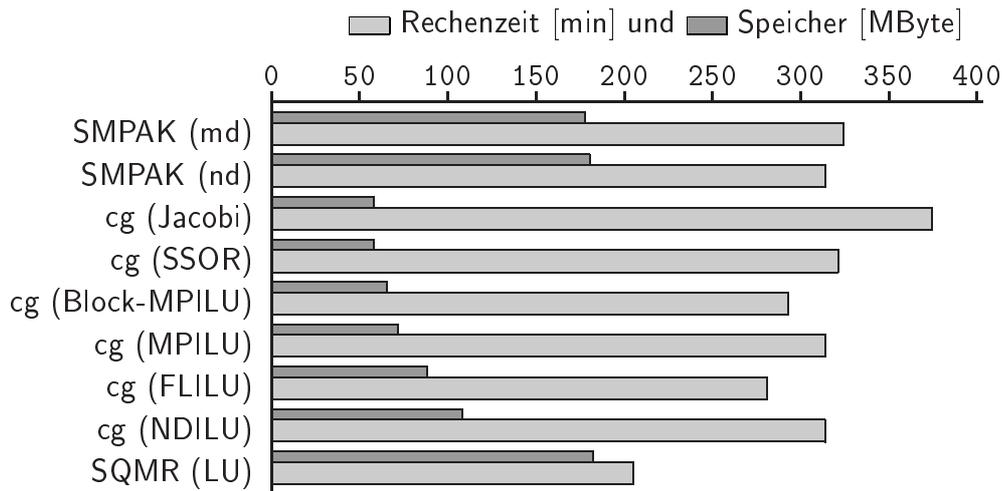


Abbildung 4.13: Rechenzeit und Speicherbedarf auf IBM RS6000 für den Torus (adaptives Netz)

Für den Torus mit dem adaptiven Netz ist in Abbildung 4.13 der Speicherplatzbedarf und die zur Berechnung des gesamten nichtlinearen Last-Verformungsverhaltens benötigte Rechenzeit dargestellt. Auch hier – analog zu den linearen Vergleichen – ist der Speicherplatzbedarf der direkten Verfahren deutlich höher als der der iterativen Verfahren. Die höchsten Speicherplatzanforderungen stellt die eben eingeführte gemischt direkt-iterative Variante SQMR-LU. Beim Vergleich der Rechenzeit muß nun aber wegen der Verwendung des Bogenlängenverfahrens beachtet werden, daß pro Newton Schritt zwei lineare Gleichungssysteme mit derselben Koeffizientenmatrix gelöst werden müssen, was einen Vorteil für direkte Löser darstellt, da die Matrix nur einmal faktorisiert werden muß. Bei der iterativen Lösung hingegen muß immer erneut iteriert werden und lediglich die vorher berechnete Vorkonditionierung kann weiterverwendet werden. Die Rechenzeiten, die mit den iterativen Verfahren erzielt wurden, sind ähnlich wie die der direkten Verfahren. Mit der Block-MPILU- und FLILU-Vorkonditionierung wurde sogar eine merklich geringere Rechenzeit erzielt. Die NDILU-Vorkonditionierung liefert bei nichtlinearen Aufgaben eine durchaus akzeptable Effizienz. Eine deutliche Reduktion der Rechenzeit wurde aber erst mit der Verwendung der Kombination aus direktem und iterativem Verfahren SQMR-LU erreicht. Mit letzterem wurde die benötigte Rechenzeit für dieses Problem um ein Drittel reduziert.

Noch deutlicher fallen die Ergebnisse beim Rohrkreuz aus (siehe Abbildung 4.14). Hier wird die Rechenzeit infolge des gemischt direkt-iterativen Verfahrens verglichen mit dem direkten Verfahren um mehr als die Hälfte reduziert. Mit iterativen Lösern allein wird für

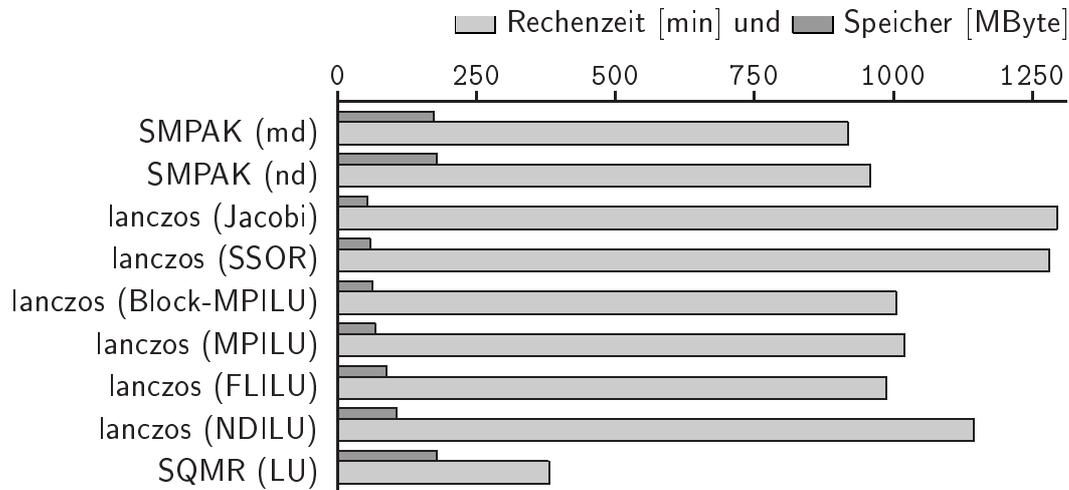


Abbildung 4.14: Rechenzeit und Speicherbedarf auf NEC SX4 für das Rohrkreuz

dieses Beispiel kein Vorteil gegenüber direkten Lösern erzielt, wenngleich die Rechenzeiten durchaus vergleichbar sind. Die effizientesten Vorkonditionierer sind auch für dieses Beispiel wiederum Block-MPILU und FLILU. Die einfachen Vorkonditionierungsstrategien wie Jacobi und SSOR, die bei linearen Problemen auch für Schalenprobleme effiziente Lösungen ermöglichten, erweisen sich bei nichtlinearen Problemen als schlechter. Grund hierfür ist die bei fortschreitender Nichtlinearität sich verschlechternde Kondition der Steifigkeitsmatrizen, die eine bessere Vorkonditionierung erfordert, siehe hierzu Tabelle 3.3.

## 4.5 Richtlinien zur Wahl eines Löser

Aus den gezeigten linearen und nichtlinearen Beispielen lassen sich nun einige Richtlinien zur Wahl eines Löser angeben. Da es sich dabei um Folgerungen und Verallgemeinerungen aus den vorgestellten Ergebnissen handelt, ist dabei zu beachten, daß die Gültigkeit begrenzt ist. So handelt es sich ausschließlich um dünn besetzten Matrizen, die meist positiv definit sind oder nur einige wenige negative Eigenwerte besitzen. Da für nichtlineare Probleme keine geschlossene Theorie vorliegt, können aber auch keine allgemeingültigen Aussagen getroffen werden. Dennoch erlaubt die breit angelegte Wahl der Benchmark Probleme mit unterschiedlichen mechanischen und damit auch mathematischen Eigenschaften einige Schlüsse, die für ähnliche Probleme die Wahl eines geeigneten Löser ermöglicht.

Selbstverständlich kann über Lösungsverfahren, die nicht in den Vergleich einbezogen wurden keine Aussage getroffen werden. Ebenso sind die erzielten Rechenzeiten nicht auf andere Hardwareplattformen übertragbar – das grundsätzlich Verhalten unterschiedlicher Rechnerarchitekturen wird aber ähnlich bleiben.

Unter diesen Voraussetzungen werden folgende Richtlinien angegeben:

- Steht nur wenig Hauptspeicher zu Verfügung oder ist das Problem sehr groß, ist ein

iterativer Löser vorteilhaft.

- Auf Vektorrechnern sind die direkten Multifrontal- und Profillöser den anderen Verfahren in Bezug auf die benötigte Rechenzeit überlegen.
- Bei Kontinuumsproblemen liefern iterative Löser mit einem einfachen Vorkonditionierer wie z.B. Jacobi, SSOR oder Block-MPILU sehr effiziente Ergebnisse. Hier sind direkte Löser deutlich unterlegen.
- Bei nichtlinearen Problemen wird für direkte *sparse*-Löser zwar mehr Speicher benötigt, letztere sind aber oft effizienter die Rechenzeit betreffend als iterative Verfahren. Dies gilt insbesondere bei Verwendung von Bogenlängenverfahren.
- Für nichtlineare Schalenprobleme ist die Wahl der Block-MPILU, MPILU oder FL-ILU-Vorkonditionierung am vorteilhaftesten; dabei sollte die Vorkonditionierungsmatrix nur ein Mal pro Newton Schritt aufdatiert werden.
- Ist das Finite Elemente Netz nicht sorgfältig numeriert, sollte unbedingt eine Band- oder Profilverminimierung vor der Berechnung einer unvollständigen Faktorisierung erfolgen.
- Für nichtlineare Probleme stellt die Kombination aus direktem und iterativem Verfahren die mit Abstand effizienteste Möglichkeit dar, sofern ausreichend Speicherplatz zu Verfügung steht.

---

## 5 Parallelverarbeitung

Den wachsenden Anforderungen der Anwender numerischer Simulationsprogramme an Rechenleistung stand bislang ein etwa äquivalenter Fortschritt in der Hard- und Softwaretechnologie gegenüber. Die Herstellbarkeit leistungsfähiger Prozessoren ist aber wirtschaftlich und technisch eingeschränkt, da die Entwicklung neuer Technologien hohe Investitionen erfordert und der voranschreitenden Miniaturisierung der Prozessoren und Speichereinheiten physikalische Grenzen gesetzt sind (die Informationen im Prozessor können nicht schneller als mit Lichtgeschwindigkeit fließen, die Leiterbahnbreite ist nicht beliebig reduzierbar usw.).

Eine Möglichkeit, diese Probleme zu umgehen, liegt in der Erhöhung der Rechenleistung durch den gleichzeitigen, parallelen Einsatz mehrerer Prozessoren, d.h. dem Übergang vom sequentiellen zum parallelen Rechner. Dazu ist das Gesamtproblem in kleinere Teilprobleme zu zerlegen, die dann parallel von verschiedenen Prozessoren bearbeitet werden. Allerdings ist in den meisten Fällen eine aufwendige Überarbeitung ("Parallelisierung") der Software notwendig. Aus diesem Grunde stehen bislang nur wenige kommerzielle Simulationsprogramme in einer parallelen Version zur Verfügung.

Nach einer kurzen Einführung in die Begriffswelt der Parallelverarbeitung wird die im Rahmen dieser Abhandlung durchgeführte Parallelisierung eines nichtlinearen, statischen und dynamischen, impliziten Finite Element Programmes beschrieben. Die gewählte Parallelisierungsstrategie beruht auf einer geometrischen Zerlegung des bereits diskretisierten Finite Element Modells durch allgemein verfügbare Algorithmen. Die Funktionsweise dieser Algorithmen zur Gebietszerlegung wird kurz erläutert und an einem Beispiel illustriert. Besondere Beachtung wird dann einigen Details zur Behandlung der durch die Gebietszerlegung entstehenden inneren Rändern geschenkt. Da die auf den Einzelprozessoren berechnete Steifigkeitsmatrix sich an diesen inneren Rändern von der Gesamtsteifigkeitsmatrix unterscheidet, ist die Kenntnis der Gleichungen, die diesem inneren Rand entsprechen, notwendig. Dieses ist jedoch abhängig von den jeweiligen Elementen bzw. den zugehörigen Knoten. Es wird eine Datenstruktur angegeben, die in einfacher Weise die Behandlung aller Elementtypen zuläßt. Auf die parallele Implementierung der iterativen Löser wird nicht eingegangen, da dies bereits in [94, 93, 121] geschehen ist.

Abschließend werden einige wenige numerische Resultate gezeigt, die die Funktionalität und Effizienz der parallelen Programmversion belegen. Die Mehrzahl der in Abschnitt 8 durchgeführten transienten Untersuchungen wurden aus Effizienzgründen ebenfalls mit der parallelen Programmversion durchgeführt, ohne daß dies dort explizit diskutiert wird.

### 5.1 Klassifikation von Parallelrechnern

Die Klassifikation von Parallelrechnern erfolgt zunächst aufgrund ihrer Speicherorganisation. Es wird im Wesentlichen zwischen *shared memory* und *distributed memory* Rechnern unterschieden.

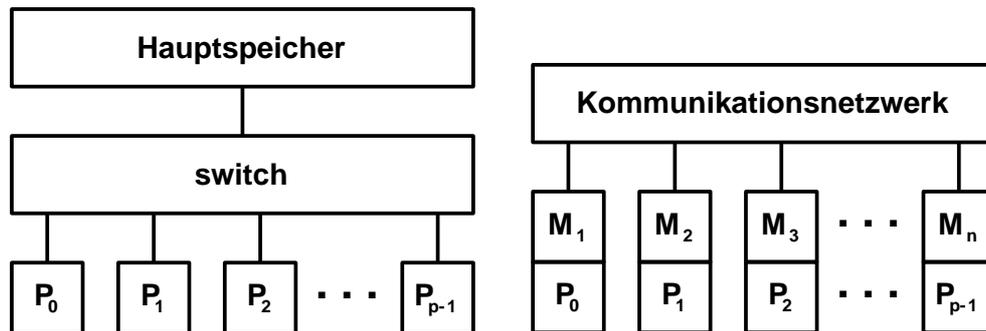


Abbildung 5.1: Schematische Darstellung einer *shared memory* Architektur (links) und einer *distributed memory* Architektur (rechts)

Bei *shared memory* Rechnern steht ein gemeinsamer Hauptspeicher zur Verfügung, der über einen *switch* von allen Prozessoren angesprochen werden kann, siehe Abbildung 5.1 (links). Die Zugriffszeit ist deshalb für alle Prozessoren gleich lang, es kann aber immer nur ein Prozessor auf einen Speicherbereich zugreifen. Benötigen andere Prozessoren das gleiche Speicherelement, so müssen sie auf das Ende der Operation des ersten Prozessors warten. Wegen dieser sequentiellen Nutzung entstehen Wartezeiten für Prozessoren, die nur durch eine effiziente Nutzung des Busses minimiert werden können. Eine solche Entlastung des Datenbusses und des Switches wird im Allgemeinen über große lokale Daten-caches erreicht. Die Organisation des Datenverkehrs übernimmt das System. Die Effizienz der Kommunikation wird deshalb auch maßgeblich durch das System und nicht durch die Programmierung bestimmt. Typische *shared memory* Rechner sind z.B. Cray Y-MP, Cray C90, NEC SX-3, SGI Origin; sie verfügen typischerweise über bis zu 16 Prozessoren.

Weitaus höhere Prozessoranzahlen sind bei *distributed memory* Rechnern üblich. Da hier, wie aus Abbildung 5.1 (rechts) ersichtlich, jeder Prozessor über einen eigenen lokalen Speicher verfügt, auf den er ohne Beteiligung des Netzwerkes zugreifen kann, ist diese Architektur auch für größere Prozessoranzahlen geeignet. Allerdings kann ein Prozessor nicht direkt auf Daten, die sich im lokalen Speicher eines anderen Prozessors befinden, zugreifen. Werden solche Daten benötigt, muß ein Datentransfer zwischen diesen beiden Prozessoren stattfinden. Es ist die Aufgabe des Programmierers, die Daten und die Rechenlast so auf die Prozessoren zu verteilen, daß alle Prozessoren gleich viel zu tun haben und möglichst wenig Datenaustausch notwendig ist. Entsprechend aufwendig gestaltet sich die Programmierung solcher *distributed memory* Rechner. Allerdings lassen sich durch eine effiziente Implementierung auch große Rechenzeitgewinne erzielen. Im Gegensatz zu *shared memory* Rechnern spielt es bei *distributed memory* Rechnern sehr wohl eine Rolle, welche Prozessoren miteinander kommunizieren. Wegen der vorhandenen Netzwerktopologie, siehe Abschnitt 5.4, unterscheidet sich die zur Kommunikation benötigte Zeit. Typische *distributed memory* Rechner sind z.B. INTEL PARAGON, NEC SX-4, DEC-ALPHA, CRAY-T3E oder IBM SP2.

In der industriellen Berechnungspraxis ist es zur Zeit üblich, auf *shared memory* Rechnern das *distributed memory* Konzept zu simulieren, d.h. die einzelnen Prozessoren machen keinen Gebrauch von dem gemeinsamen Hauptspeicher, aber die Kommunikation wird

dabei deutlich reduziert und eine erhebliche Rechenzeiterparnis erreicht.

## 5.2 Programmiermodelle

Das Ziel einer jeden parallelen Bearbeitung liegt darin, das vorhandene Problem in möglichst gleichen Teilen auf die Prozessoren zu verteilen und dieselbe Lösung zu erzielen wie im seriellen Fall. Die Lastverteilung soll dabei gleichmäßig sein, damit kein Prozessor auf das Teilergebn anderer Prozessoren warten muß. Außerdem soll wenig Kommunikation benötigt werden, da die Geschwindigkeit des Netzwerkes im Allgemeinen sehr gering im Vergleich zur Rechengeschwindigkeit ist.

Es existieren grundsätzlich vier unterschiedliche Programmiermodelle, je nachdem ob das gleiche Programm auf allen Prozessoren abläuft (single program) oder ob jeder Prozessor ein eigenes Programm ausführt (multiple program) und ob jedem Prozessor die selben Daten vorliegen (single data) oder ob die Daten verteilt sind (multiple data).

Die üblichste Variante für *distributed memory* Rechner ist das SPMD Modell. Dabei läuft auf jedem Prozessor das identische Programm ab, jeder Prozessor wendet das Programm aber auf andere Daten an.

Beim MPMD Modell werden von den Prozessoren unterschiedliche Programme geladen, die auch auf unterschiedliche Daten angewendet werden. Hier ist vor allem das *master/slave* Konzept zu erwähnen, bei dem ein Prozessor – der *master* – den Programmablauf steuert und den anderen Prozessoren – den *slaves* – die auszuführenden Teilaufgaben zuweist.

Das klassische serielle Modell ist SPSP, das MPSP Modell erzeugt durch unterschiedliche Anweisungen Ausgabedaten aus den Eingangsdaten und nutzt so den Pipelineeffekt.

## 5.3 *Message Passing*

Die einzige Möglichkeit für *distributed memory* Rechner, Daten auszutauschen, liegt im expliziten *message passing*, also im Verschicken von Datenpaketen. Die Kontrolle, welche Daten wann von welchem Prozessor wohin verschickt werden, liegt vollständig beim Programmierer. Dies macht zwar die Programmierung aufwendig (und zum Teil wenig portabel), ermöglicht aber auch ein Maximum an Effizienz.

Das *message passing* kann sowohl synchron, als auch asynchron erfolgen. Beim synchronen *message passing* sendet ein Prozessor, während der andere empfängt. Der Vorgang der Kommunikation erfolgt gleichzeitig, synchron. Über das Kommunikationsnetzwerk werden die Prozessoren verbunden, die im Sendepuffer bereitstehenden Daten übertragen und in den Empfangspuffer geschrieben. Mit dieser Kommunikation wird das Programm synchronisiert. Die an der Kommunikation beteiligten Prozessoren befinden sich im Programmablauf an der selben Stelle. Erreicht ein Prozessor diese Stelle aufgrund schlechter Lastverteilung früher, so muß er im Programmablauf warten. Ein weiteres Problem der synchronen Kommunikation liegt in der Gefahr des *deadlocks*. Dies soll an einem kleinen

Beispiel kurz erläutert werden: Bei einer Berechnung mit zwei Prozessoren soll Prozessor Null ein Zwischenergebnis an Prozessor Eins schicken und umgekehrt. Beide wollen zunächst synchron senden und dann empfangen. Beim Senden wartet jeder Prozessor auf die Empfangsbereitschaft des anderen, die nie eintritt – der Programmablauf ist gestoppt, was bei komplizierteren, datenabhängigen Konfigurationen nicht ohne weiteres vorher absehbar ist.

Nicht möglich ist eine solche Situation bei der asynchronen Kommunikation. Hier wird ein Sende- bzw. Empfangsvorgang zunächst durch die Bereitstellung entsprechender Puffer initiiert, danach kann der Programmablauf, z.B. weitere Berechnungen, wieder aufgenommen werden. Der Zustand des Kommunikationsvorganges kann dann über entsprechende Funktionen überprüft und gegebenenfalls abgeschlossen werden. Während dieser Zeit muß der Programmierer dafür Sorge tragen, daß weder der Sende- noch der Empfangspuffer überschrieben werden, bevor der Kommunikationsvorgang abgeschlossen ist. Durch eine solche asynchrone Kommunikation ist ein sogenanntes *latency hiding*, ein Verstecken der Kommunikation hinter der Berechnung möglich, zudem kann kein *deadlock* auftreten, da nach initiiertem Sendevorgang auch ein Datenempfang möglich ist.

Ein großes Problem bei der Programmierung mit explizitem *message passing* bestand lange in der Portabilität der Programme zwischen unterschiedlichen Rechnern, da die FORTRAN bzw. C- Erweiterungen nicht genormt waren. Dieses Problem wurde durch die Einführung des *Message-Passing Interface* [134] und von *Parallel Virtual Machine* [47] behoben.

## 5.4 Netzwerktopologien

Die Topologie, d.h. der geometrische Aufbau eines Netzwerkes, ist für die Kommunikationsgeschwindigkeit mitverantwortlich. Entscheidend bei Kommunikation zwischen zwei Prozessoren, der sogenannten der Punkt-zu-Punkt Kommunikation, ist die "Entfernung" zweier Prozessoren im Netzwerk. Sind die Prozessoren direkt miteinander verbunden, erfolgt die Kommunikation schneller, als wenn der Datenaustausch indirekt über andere Prozessoren stattfinden muß. Optimal für den Kommunikationsvorgang ist es, wenn die beiden kommunizierenden Prozessoren benachbart sind, da dann eine direkte Verbindung existiert. Ist dies nicht der Fall, wird durch einem sogenannten *routing*-Algorithmus die günstigste Verbindung vom System gewählt. Bei bekannter Topologie ist es programmtechnisch möglich, den Datenaustausch effizient zu gestalten, indem z.B. die sogenannte *nearest neighbor* Kommunikation mit benachbarten Prozessoren ausgenutzt wird.

In den folgenden Abschnitten werden einige Topologien und deren Vor- und Nachteile vorgestellt.

### 5.4.1 Vollständiges Netzwerk

Beim vollständigen Netzwerk ist jeder Prozessor mit allen anderen Prozessoren verbunden, siehe Abbildung 5.2 (links). Diese Topologie besitzt ideale Eigenschaften für die Punkt-zu-

Punkt Kommunikation, da jede Verbindung eine *nearest neighbor* Verbindung ist. Allerdings ist diese Topologie nur schwer erweiterbar, bei großen Netzwerken ist der Aufwand zur Herstellung erheblich.

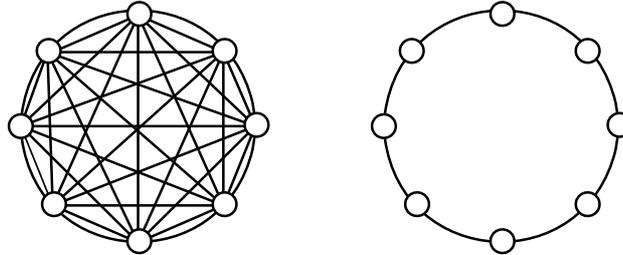


Abbildung 5.2: Vollständiges Netzwerk und Ringtopologie mit jeweils 8 Knoten

#### 5.4.2 Ringtopologie

In der Ring-Topologie ist jeder Prozessor nur mit seinen topologisch nächstgelegenen Nachbarn direkt verbunden, siehe Abbildung 5.2 (rechts). Im ungünstigsten Fall ist ein *routing* über  $p/2$  Prozessoren notwendig, wenn  $p$  die Anzahl der Prozessoren ist. Die Hardwarevoraussetzungen für einen Ring sind verglichen mit dem vollständigen Netzwerk sehr gering.

#### 5.4.3 Gitter- bzw. Torustopologie

In der Gitter- bzw. Torustopologie ist die Anzahl der direkten Verbindungen pro Prozessor unabhängig von der Gesamtanzahl  $p$  der Prozessoren. Ein Gitter der Ordnung 1 entspricht einer einfachen Pipeline, ein Torus der Ordnung 1 ist identisch mit einem Ring. Gitter und Tori der Ordnung 2 sind in Abbildung 5.3 dargestellt. Das *routing* erfolgt bei den Gittertopologien entlang der "Zeilen" und "Spalten", es existieren mehrere gleichwertige Wege, von denen einer vom *routing*-Algorithmus ausgewählt wird. Ein Beispiel für einen Parallelrechner mit einem Rechteckgitter ist die INTEL PARAGON.

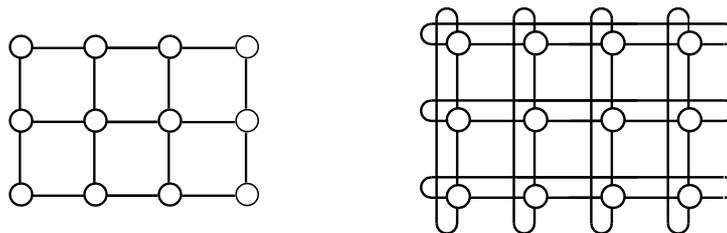


Abbildung 5.3: Gitter- und Torustopologie 2. Ordnung

#### 5.4.4 Mehrstufige *Crossbar Switches*

Ein *crossbar switch* ist ein vollständiges Netzwerk. Jeder Eingang ist mit jedem Ausgang verbunden, siehe Abbildung 5.4 links. Dadurch ist jede Kommunikation der Prozessoren, die mit einem *crossbar switch* verbunden sind, eine *nearest neighbor* Kommunikation.

Ein mehrstufiger *crossbar switch* wird aus mehreren einstufigen *crossbar switches* zusammengesetzt. Ein Beispiel für einen zweistufigen *crossbar switch* ist in Abbildung 5.4 rechts abgebildet, er besteht aus acht einstufigen *crossbars*. Soll eine Nachricht von Prozessor  $P_1$  zu Prozessor  $P_3$  gesendet werden, so ist dazu ein sogenannter *hop* notwendig. Das bedeutet, daß die Nachricht im gleichen *crossbar* zugestellt werden kann, ein weiterer *crossbar* wird zur Übermittlung der Nachricht nicht benötigt. Eine Nachricht von Prozessor  $P_1$  zu Prozessor  $P_{11}$  benötigt zwei *hops*, da nun zwei *crossbars* an der Übertragung beteiligt sein müssen. Der Vorteil der *crossbars* liegt darin, daß selbst bei großen Prozessoranzahlen nur wenige *hops* benötigt werden, bei 64 Prozessoren sind beispielsweise maximal vier *hops* erforderlich. Es läßt sich dann aber keine *nearest neighbor* Kommunikation mehr implementieren, da benachbarte Prozessoren nicht immer durch weniger *hops* erreichbar sind als weit auseinanderliegende. In Abbildung 5.4 ist leicht zu sehen, daß eine Nachricht von Prozessor  $P_3$  zum benachbarten Prozessor  $P_4$  zwei *hops* benötigt, ebenso wie eine Nachricht zu dem weit entfernten Prozessor  $P_{14}$ . Allerdings ist die Anzahl der gleichzeitig möglichen Datentransfers von der Gesamtlast des Rechners abhängig, da auch andere Applikationen Kommunikation über den *crossbar* haben können und damit das Netzwerk belasten. Deshalb ist die Zeit für einen Kommunikationsvorgang nicht vorhersagbar. Einen typischen Rechner dieser Netzwerktopologie stellt die IBM SP2 dar.

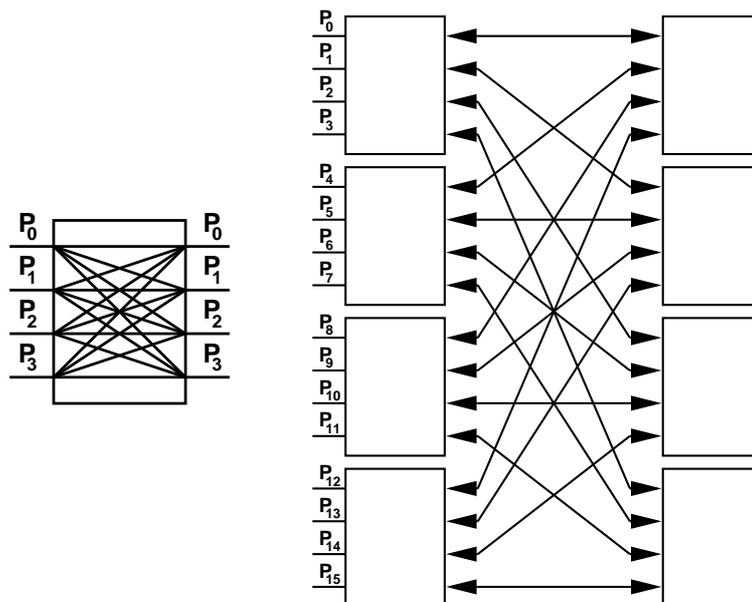


Abbildung 5.4: Ein- und zweistufiger *crossbar switch*

## 5.5 *Speedup*, Effizienz, Amdahl's Gesetz

Die Beschleunigung einer parallelen Applikation im Vergleich zur seriellen Bearbeitung wird als *speedup* bezeichnet. Er ist als Verhältnis der Rechenzeit  $t_p$  mit  $p$  Prozessoren und der Rechenzeit  $t_1$  mit einem Prozessor definiert:

$$S_p = \frac{t_1}{t_p}. \quad (5.1)$$

Wird mit der Rechenzeit auf einem Prozessor verglichen, so sollte dies grundsätzlich mit dem effizientesten seriellen Algorithmus geschehen. Dieser muß mit dem parallelen nicht identisch sein, d.h. es können verschiedene Verfahren an der *speedup* Bestimmung beteiligt sein. Oft wird aber auch das parallele Programm auf einem Prozessor abgearbeitet, was zu einem besseren *speedup* führt. Zur parallelen Verarbeitung notwendiger Mehraufwand ist dann auch in der seriellen Vergleichszeit enthalten, und die Forderung nach dem effizientesten sequentiellen Algorithmus ist nicht notwendigerweise erfüllt. Ein ineffizienter Algorithmus kann dann einen hervorragenden *speedup* liefern und dennoch keine effiziente Lösung des Problems. Aus diesen Gründen ist ein gewisses Mißtrauen vor *speedup* Angaben angebracht, soweit nicht offengelegt wird, was als serielles Referenzmaß verwendet wurde.

Ein anderes Maß zur Beurteilung der Beschleunigung eines Algorithmus ist die Effizienz

$$E_p = \frac{S_p}{p}, \quad 0 \leq E_p \leq 1. \quad (5.2)$$

Bei der vollständigen Parallelisierung einer Applikation könnte ohne Berücksichtigung des Aufwandes zur Organisation des Datenaustausches bei der Verwendung von  $p$  Prozessoren eine Verringerung der Rechenzeit mit dem Faktor  $p$  erwartet werden, d.h. ein linearer *speedup* bzw. eine Effizienz der Größe Eins. Bei einem überlinearen *speedup* bzw. einer Effizienz größer Eins, wie es gelegentlich in Veröffentlichungen zu sehen ist, wirken sich andere Phänomene wie beispielsweise die Ausnutzung des *caches* bei unterschiedlicher Vektorlänge aus. In der Regel wird ein unterlinearer *speedup* erzielt, da bei der parallelen Bearbeitung eines Problems stets zusätzlicher Aufwand zur Organisation und Kommunikation erforderlich wird, die Rechenlast praktisch nie exakt gleich auf die Prozessoren verteilt werden kann, die Prozessoren meist eine fertigungsbedingt leicht unterschiedliche Rechengeschwindigkeit besitzen, usw. Doch selbst wenn alle diese Einflüsse unberücksichtigt bleiben, hat AMDAHL [2] gezeigt, daß im realitätsnahen Fall, bei der Verwendung vieler Prozessoren und nicht vollständiger Parallelisierung (einige Teile von Programmen sind oft nicht parallelisierbar bzw. der Aufwand zur Parallelisierung ist sehr hoch) ein deutlich unterlinearer *speedup* zu erwarten ist. Vorausgesetzt, alle  $N$  Operationen werden von den Prozessoren gleich "schnell" in der Zeit  $T$  ausgeführt und ein Anteil  $\alpha$  aller Operationen wird vollständig parallel ausgeführt, so ist die für den parallelen Anteil benötigte Zeit  $NT\alpha/p$ . Der verbleibende serielle Anteil der  $N$  Operationen wird in der Zeit  $NT(1 - \alpha)$

abgearbeitet. Die Gesamtrechenzeit mit Nutzung von  $p$  Prozessoren betragt demnach

$$t_p = \frac{NT\alpha}{p} + NT(1 - \alpha) = NT \frac{\alpha + (1 - \alpha)p}{p} \geq NT(1 - \alpha), \quad (5.3)$$

und mit (5.1) ergibt sich

$$S_p = \frac{p}{\alpha + (1 - \alpha)p}, \quad \alpha < 1, \quad S_p \leq \frac{1}{1 - \alpha}. \quad (5.4)$$

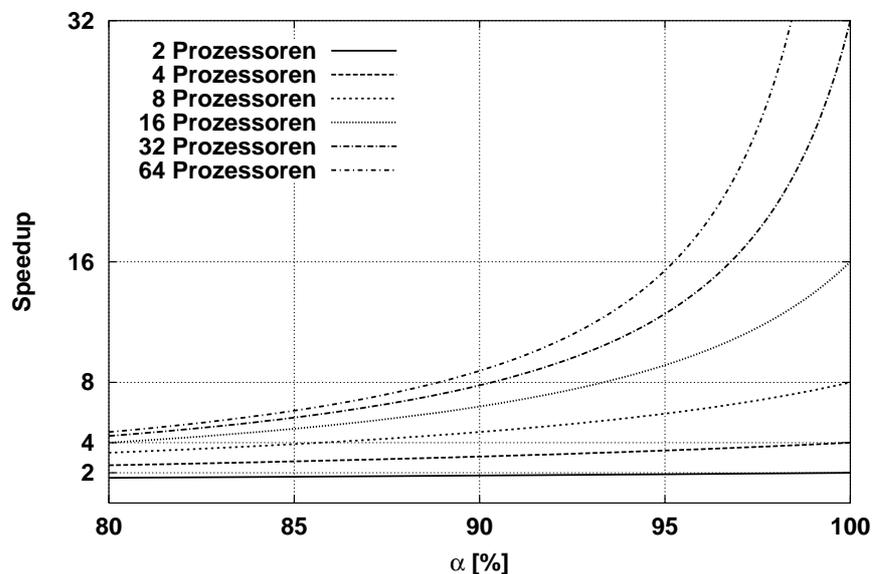


Abbildung 5.5: Amdahl's Gesetz – Zusammenhang zwischen dem Parallelisierungsgrad und dem *speedup* Faktor

Dieser formelmaige Zusammenhang wird als *Amdahl'sches Gesetz* bezeichnet und ist in Abbildung 5.5 graphisch dargestellt. Dabei ist feststellbar, da bei der Verwendung zweier Prozessoren auch bei einem Parallelisierungsgrad von nur  $\alpha = 80\%$  bereits ein *speedup* von nahezu Zwei erreicht wird. Bei der Verwendung hoherer Prozessoranzahlen zeigt sich jedoch schnell ein deutlich geringerer *speedup*, bei 16 Prozessoren und ebenfalls einem Parallelisierungsgrad  $\alpha = 80\%$  kann nur ein *speedup* von etwa 4 erreicht werden. Erst bei einer fast vollstandigen Parallelisierung ( $\alpha \approx 100\%$ ) kann ein linearer *speedup* erzielt werden.

Ein Nachteil des Modells der Messung der Rechenzeitverkurzung bei der Nutzung paralleler Rechner mittels *speedup* besteht darin, da es in der praktischen Anwendung wenig Sinn macht, ein Problem vorgegebener Groe auf immer mehr Prozessoren zu verteilen, da hierbei der Rechenanteil pro Prozessor wegen der kleineren Teilgebiete sinkt und gleichzeitig der Kommunikationsaufwand mit mehr Prozessoren entlang des zunehmenden Randanteils steigt.

Eine weitere Möglichkeit, die Qualität einer parallelen Implementierung zu beurteilen, liegt im sogenannten *scaleup*. Mit wachsender Prozessorenanzahl wächst hierbei auch die Problemgröße, sodaß das von jedem Prozessor bearbeitete Teilproblem eine konstante Größe aufweist. Die insgesamt benötigte Rechenzeit soll dann auch möglichst konstant bleiben und nicht anwachsen, der *scaleup* also Eins betragen. Diese Vorgehensweise kommt dem eigentlichen Anwenderbedarf – immer “größere” Simulationen in der möglichst gleichen Zeit durchzuführen – näher als die Betrachtung des *speedup*. Jedoch ist hierzu anzumerken, daß der Aufwand zur Erstellung eines *scaleup* Vergleichs sehr hoch ist, da das Problem für jede Prozessoranzahl neu diskretisiert werden muß, was meist nicht vollständig automatisch durchgeführt werden kann.

## 5.6 Parallelisierung eines Finite Element Programmes

In den nachfolgenden Abschnitten wird die im Rahmen dieser Abhandlung durchgeführte Parallelisierung des Finite Element Programms FEAP [131] beschrieben. Die Parallelisierungsstrategie basiert auf einem geometrischen Ansatz zur Verteilung der Rechenlast, d.h. der Menge der numerischen Berechnungen auf einzelne Prozessoren. Das gegebene Finite Element Netz wird so zerteilt, daß jedem Prozessor ein möglichst gleich großer und zusammenhängender Netzanteil zugewiesen wird, wie es in Abbildung 5.6 dargestellt ist. Die Gebietszerlegung wird dabei in einem seriellen *preprocessing* Schritt auf einem Prozessor durchgeführt. Die dazu verwendeten Verfahren werden in Abschnitt 5.6.1 beschrieben und an einem Beispiel illustriert. Derzeit wird eine statische Verteilung vorgenommen, d.h. es findet während der Berechnung keine Repartitionierung statt, die Gebiete bleiben während der gesamten nichtlinearen Berechnung gleich auf die Prozessoren verteilt.

Die aus der Gebietszerlegung resultierende Datenstruktur wird im Abschnitt 5.7 diskutiert; der zur linearen Gleichungslösung eingesetzte iterative parallele Löser im Abschnitt 5.8 erläutert. Weitere Aspekte der Parallelisierung für nichtlineare Finite Element Berechnungen werden anschließend im Abschnitt 5.9 besprochen.

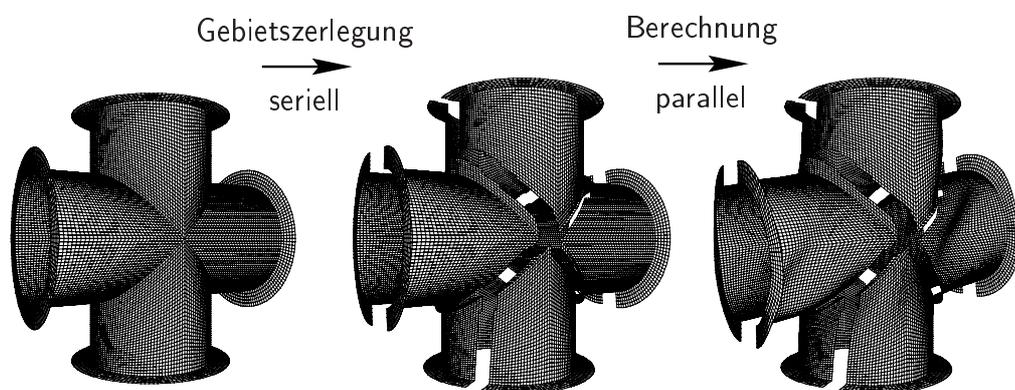


Abbildung 5.6: Schema des Parallelisierungskonzeptes

### 5.6.1 Gebietszerlegungsverfahren

Zur parallelen Bearbeitung eines Problems muß dieses in Teilprobleme zerlegt werden, die dann auf die einzelnen Prozessoren verteilt werden. Das Ziel dieser Partitionierung liegt darin, möglichst gleich große Teilprobleme zu erhalten, um ein gutes *load balancing*, d.h. eine gleichmäßige Verteilung der Rechenlast zu erreichen. Um die Kommunikation zu minimieren, soll gleichzeitig eine große Datenlokalität gegeben sein. Es wird also angestrebt, daß jeder Prozessor die von ihm benötigten Daten in seinem Speicher hat. Darüber hinaus könnten weitere Kriterien berücksichtigt werden, wie z.B. eine besondere Zuordnung der Gebiete für eine bestimmte Prozessortopologie, um damit bevorzugt *nearest neighbor* Kommunikation zu erzielen. Letzteres wird jedoch oft nicht beachtet, da die Kommunikationszeiten zunehmend von der Netzwerktopologie unabhängig werden (siehe z.B. Abschnitt 5.4.4). Durch die Optimierung auf eine spezielle Topologie sinkt außerdem die Portabilität, d.h. die Übertragbarkeit der Aufteilung.

Im Kontext der Finiten Element Methode gibt es grundsätzlich zwei Möglichkeiten der Verteilung des Problems auf die Prozessoren. Es können entweder die Elemente auf die Prozessoren verteilt werden, die im Wesentlichen den Rechenaufwand beim Assemblieren der Steifigkeitsmatrix bestimmen. Andererseits können die Knoten verteilt werden, die hauptsächlich den Aufwand bei der Lösung der linearen Gleichungssysteme beeinflussen. Üblicherweise werden die Elemente auf die Prozessoren verteilt, womit ein vollständig paralleles Assemblieren der Steifigkeitsmatrix ermöglicht wird. Diese Vorgehensweise wurde im Rahmen dieser Abhandlung gewählt. Die Forderung eines guten *load balancing*s bedeutet daher: Die Anzahl der Elemente in jedem Teilgebiet soll möglichst gleich sein. Beim Lösen der linearen Gleichungssysteme kann es aber infolgedessen zu einem sogenannten Lastungleichgewicht kommen, einzelne Prozessoren können höheren Rechenaufwand haben als andere. Knoten, deren Freiheitsgrade mit Randbedingungen versehen sind, liefern nämlich keine Gleichungen, weshalb manche lokale Gleichungssysteme kleiner sein können als andere. Dies hat wiederum Einfluß auf die Rechenlast, z.B. beim Matrix Vektor Produkt innerhalb eines iterativen Lösungsverfahrens. Dieser Tatsache könnte durch eine entsprechende Vorabgewichtung entsprechender Elemente bei der Gebietszerlegung begegnet werden. Die zweite Forderung nach großer Datenlokalität führt unter mechanischen Gesichtspunkten zu der Forderung nach zusammenhängenden und kompakten Teilgebieten. Knoten – genauer deren Freiheitsgrade – die nicht durch ein Element miteinander verbunden sind, liefern in der Steifigkeitsmatrix keine Koppeleinträge. Es ist daher unmittelbar einsichtig, daß gerade die Knoten in einem Teilnetz vereinigt werden sollen, die durch Elemente verbunden sind und Koppelterme in der Steifigkeitsmatrix verursachen.

Das Aufbrechen geschlossener mechanischer Strukturen durch die Gebietszerlegung kann zu geänderten mechanischen Eigenschaften der Teilgebiete führen. Angestrebt wird eine Lokalität des Lösungsprozesses, der also durch die Eigenschaften der lokalen Probleme beeinflußt wird. Damit hat also bereits die Gebietszerlegung einen Einfluß auf die Effizienz der Gesamtlösung. In [121] wurde diese Eigenschaft an einem speziellen Beispiel diskutiert. Im allgemeinen Fall werden solche Überlegungen aber nicht in die Gebietszerlegungsalgorithmen eingehen, da das Erkennen mechanischer Eigenschaften nur schwer in numerische Algorithmen einzubringen ist.

Im Schrifttum liegen bereits viele Veröffentlichungen zu Verfahren zur Gebietszerlegung vor. Einige Verfahren arbeiten dabei mit geometrischen Größen, um eine Zerlegung zu bestimmen, die weitaus größere Zahl verwendet jedoch graphenorientierte Methoden, die zum Teil lokalen oder auch globalen Charakter besitzen. Während die Verfahren mit lokalem Charakter von einem gewählten Startpunkt aus mit unterschiedlichen Kriterien versuchen ein Gebiet auszudehnen, berücksichtigen Verfahren mit globalem Charakter die Gesamtstruktur des Graphen.

In den folgenden Abschnitten sollen einige wesentliche der in Veröffentlichungen genannten Verfahren kurz vorgestellt und anhand eines Beispiels die damit erhaltenen Gebietszerlegungen diskutiert werden. Für eine ausführlichere Übersicht über Gebietszerlegungsverfahren wird z.B. auf [89] verwiesen, wo auch genetische Algorithmen und neuronale Netze besprochen werden. Um graphenorientierte Verfahren zur Zerlegung eines unstrukturierten Finite Elemente Netzes zu verwenden, werden zunächst einige Überlegungen zur Überführung der Netztopologie in einen Graphen angestellt.

### 5.6.2 Vom FE-Netz zum Graphen

Zentral bei graphenorientierten Verfahren ist die Erzeugung eines Graphen aus einem vorgegebenen Finite Element Netz, der dann mit den nachfolgend beschriebenen Partitionierungsalgorithmen zerlegt werden kann. Wie bereits erwähnt, sollen die Elemente auf die Prozessoren verteilt werden. Diese stellen somit die Knoten  $V$  des Graphen dar. Nicht eindeutig hingegen ist die Bestimmung der Kanten des Graphen. Letztere müssen den Kommunikationsaufwand wiedergeben; je weniger Kanten geschnitten werden, desto weniger Kommunikation soll resultieren und umgekehrt.

In Abbildung 5.7 ist ein grobes FE-Netz eines Teils eines Zahnrades abgebildet, zu dem in Abbildung 5.8 zwei mögliche Graphen angegeben werden.

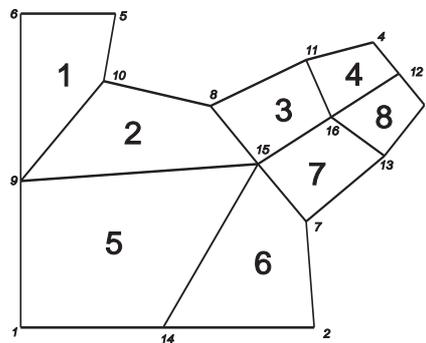


Abbildung 5.7: FE-Netz eines Teils eines Zahnrades

Im linken Graph in Abbildung 5.8, dem sogenannten dualen Diagonalgraphen, werden alle Elemente, die mindestens einen gemeinsamen Knoten besitzen, mit einer Kante verbunden. Da den gemeinsamen Knoten auch gemeinsame Einträge in der Steifigkeitsmatrix

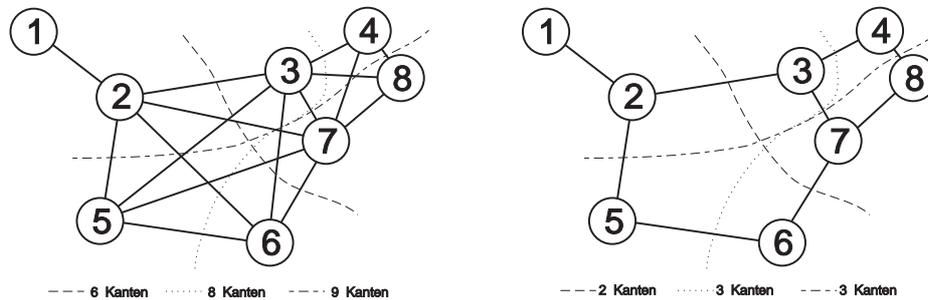


Abbildung 5.8: Dualer Diagonalgraph und dualer Graph für das Zahnrad mit möglichen Schnitten

entsprechen, ist dies der zunächst naheliegendste Ansatz, da beim Lösen der linearen Gleichungssysteme aufgrund der verteilt vorliegenden Teilsteifigkeitsmatrizen Kommunikation an diesen Stellen notwendig wird. Jedoch kann man leicht einsehen, daß dann ein Schnitt mit einem Knoten, an den viele Elemente angrenzen, stärker bewertet wird (z.B. Knoten 15), als ein Schnitt mit einem Knoten, an den wenige Elemente angrenzen (z.B. Knoten 16), die anfallende Kommunikation aber nicht größer ist.

Im zweiten, dualen Graphen werden lediglich Elemente, die eine gemeinsame Elementkante besitzen, durch Graphenkanten verbunden. Der so erstellte Graph besitzt deutlich weniger Kanten, was zu numerischen Vorteilen bei der Zerlegung führt, da eine kleinere Datenmenge verarbeitet werden muß. Außerdem ist das oben beschriebene Problem der ungleichen Gewichtung der Knoten beim Diagonalgraphen eliminiert.

Für beide Graphentypen in Abbildung 5.8 sind die Schnitte, die zwei zusammenhängende Teilgebiete mit jeweils vier Elementen ergeben durch gestrichelte Linien gekennzeichnet und die Anzahl der dadurch geschnittenen Kanten angegeben. Für beide Graphen ergibt sich dieselbe optimale Zerlegung. Dies wird auch bei größeren Netzen beobachtet, sodaß duale Graphen meist zu bevorzugen sind.

### 5.6.3 Geometrisch basierte Heuristiken zur Partitionierung

#### Koordinatenbisektion, Schwereachsenmethode

Ein Finites Elementnetz kann z.B. rein aufgrund seiner geometrischen Ausrichtung im Raum aufgeteilt werden. Entlang beliebig gewählter Achsen können die Elemente sortiert und sukzessive den Teilgebieten zugeordnet werden. Werden Koordinatenachsen benutzt, spricht man von Koordinatenbisektion, werden Schwereachsen verwendet wird das Verfahren als Schwereachsenmethode bezeichnet, siehe z.B. [89].

Diese Verfahren sind sehr einfach und effizient. Es liegt aber grundsätzlich kein Kriterium im Partitionierungsalgorithmus vor, das die Forderung nach kleinen Rändern beinhaltet und somit die Kommunikation minimiert. Es entstehen daher oft Gebiete mit sehr großen

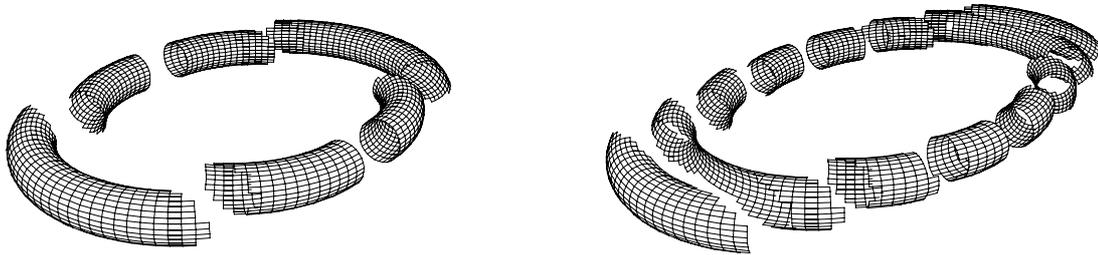


Abbildung 5.9: Koordinatenbisektion eines Torus in 4 und 8 Teilgebiete

Rändern. Außerdem ist das Verfahren nicht allgemein für beliebig geformte Strukturen gleichermaßen geeignet. Für entlang einer Richtung gestreckte mechanische Strukturen liefert es jedoch mit wenig Aufwand effiziente Partitionierungen. In Verbindung mit einer rekursiven Anwendung z.B. der Schwereachsenmethode können die Ergebnisse drastisch verbessert werden (siehe [89]).

In Abbildung 5.9 sind die mittels Koordinatenbisektion in  $x$ -Richtung (nach rechts hinten in Abbildung 5.9) erhaltenen Teilgebiete für 4 und 8 Prozessoren eines in der  $xy$ -Ebene liegenden Torus dargestellt. Die zu benutzende Achse ist im Prinzip beliebig wählbar und muß nicht mit den Koordinatenachsen zusammenfallen. Wie bereits an dem einfachen Beispiel in Abbildung 5.9 ersichtlich, können die erhaltenen Teilgebiete auch nicht zusammenhängend sein. Für das gezeigte Beispiel ist diese Aufteilung nicht vorteilhaft, da die Gebiete unzusammenhängend sind und große Ränder entstehen. Für einfache, entlang einer Linie ausgerichtete mechanische Strukturen, ist dieses Verfahren aber durchaus geeignet.

#### 5.6.4 Graphenorientierte Heuristiken zur Partitionierung

##### Kernighan-Lin Algorithmus

Bereits 1970 wurde von KERNIGHAN und LIN in [75] ein Verfahren vorgestellt, das ausgehend von einer vorgegebenen Partitionierung diese durch eine lokale Strategie verbessert. Das in [75] behandelte Problem bestand nicht in der Gebietszerlegung zur parallelen Berechnung, sondern in der Verteilung elektronischer Bauteile auf Platinen. Jede Platine kann mit einer bestimmten Anzahl elektronischer Bauteile bestückt werden; für die gesamte Schaltung müssen die Platinen mit (teuren) Leitungen verbunden werden. Die Bauteile sollen nun so auf die Platinen verteilt werden, daß möglichst wenig solcher Leitungen notwendig werden. In der Übertragung auf die Graphentheorie stellen die elektronischen Bauteile die Knoten, die Leitungen die Kanten dar. Die bei der Partitionierung geschnittenen Kanten müssen als Leitungen, die nicht geschnittenen Kanten können als Leiterbahnen auf den Platinen realisiert werden.

Es werden nun punktuell Knoten der Partitionen ausgetauscht und die so neu entstehenden Partitionen beurteilt. Verringert sich die Anzahl der geschnittenen Kanten, so werden die

Knoten tatsächlich ausgetauscht. Kann die Partitionierung nicht weiter verbessert werden, so wird das Verfahren abgebrochen.

Die Rechenzeit für dieses Verfahren ist relativ hoch (es wurde nur für kleine Graphen entwickelt) und liefert nur lokale Minima für die geschnittenen Kanten. Für eine andere Ausgangszuordnung wird im Allgemeinen eine andere Partitionierung erzielt. Jedoch kann dieses Verfahren sehr gut zur Verbesserung einer bereits guten Ausgangspartitionierung eingesetzt werden.

### Greedy-Heuristiken

Von FARHAT wurde in [40] ein Gebietszerlegungsalgorithmus vorgeschlagen, der von einem Element ausgehend die Nachbarelemente sukzessive in das erste Teilgebiet aufnimmt, bis die gewünschte Anzahl von Elementen im Gebiet erreicht ist. Für die anderen Gebiete wird äquivalent vorgegangen. Auf diese Art wird eine lokale Minimierung der Randknoten erreicht, allerdings ist die Methode sehr stark von der Wahl des Startknotens abhängig und es können durchaus nicht zusammenhängende Teilgebiete entstehen.

In [71] wird diese Vorgehensweise auf Graphen übertragen, ausgehend von einem Startknoten werden benachbarte Knoten in die Partition eingefügt. Es werden diejenigen Knoten bevorzugt, die die Anzahl der geschnittenen Kanten reduzieren (oder am wenigsten erhöhen). Die Abhängigkeit vom Startknoten wird dadurch reduziert, daß der Vorgang mehrfach mit unterschiedlichen Startknoten durchgeführt und das beste Ergebnis verwendet wird.

Damit der Algorithmus auch für große Graphen effizient bleibt, wird mit einer sogenannten *multilevel* Methode gearbeitet. Mittels unterschiedlicher Heuristiken, die in [71] ausführlich beschrieben sind, wird der ursprüngliche Graph vergrößert und dieser grobe, sehr viel kleinere Graph wird partitioniert. Um daraus eine Partitionierung des ursprünglichen Graphen zu erhalten, wird der aufgeteilte grobe Graph wieder verfeinert. Die Partitionierung muß aber beim Verfeinern mittels lokaler Strategien (z.B. mit dem Kernighan-Lin Algorithmus) wieder verbessert werden.

Auf eine Partitionierung wird in [55] vollständig verzichtet; der Vergrößerungsvorgang wird so lange fortgesetzt, bis der grobe Graph soviel Knoten besitzt, wie Teilgebiete gefordert wurden. Die hiermit erreichte Qualität der Partitionierung entspricht der in [71], die Laufzeit wurde jedoch drastisch gesenkt.

In Abbildung 5.10 ist die Aufteilung des Torus für 4 und 8 Prozessoren mittels des Greedy Algorithmus aus [72] abgebildet. Für 8 Prozessoren kann eine unvorteilhafte ungleichmäßige Aufteilung beobachtet werden. Wird aber die Anzahl der Teilgebiete weiter erhöht, so ergibt sich in diesem Fall eine der Spektralbisektion (siehe folgenden Abschnitt und Abbildung 5.11) ähnliche Aufteilung.

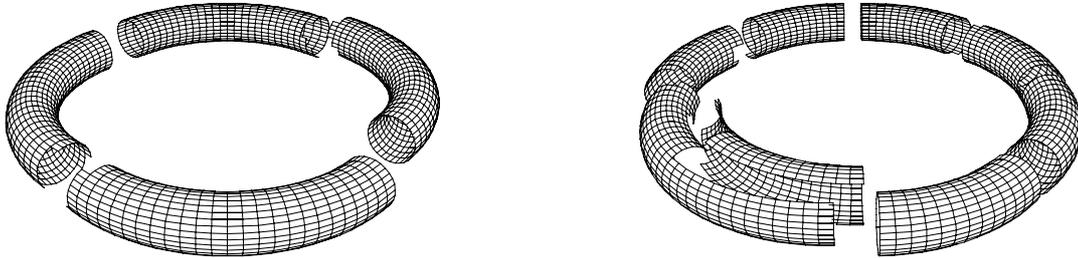


Abbildung 5.10: Greedy Partitionierung eines Torus in 4 und 8 Teilgebiete

### Spektralzerlegung

Ein leistungsfähiges Verfahren zur Graphenbisektion wurde von POTHEN, SIMON und LIOU in [112] vorgestellt. Zunächst wurde es für die Zerlegung dünnbesetzter Matrizen zur parallelen Faktorisierung entwickelt, es ist aber auch allgemein zur Partitionierung beliebiger ungerichteter Graphen geeignet. Der wesentliche Unterschied zu den bislang vorgestellten Methoden liegt hierbei in der Verwendung globaler Information über den Graphen.

Ziel ist auch hier, gleich große Untergraphen  $V_i$  zu erhalten, d.h.  $\max_i |V_i| \rightarrow \min$  soll optimiert werden mit der Nebenbedingung, daß  $\sum_{ij} |C_{ij}| \rightarrow \min$ ,  $C_{ij} = \{(v, w) | v \in V_i, w \in V_j\}$ , also daß die Anzahl der Schnittkanten  $C_{ij}$  zwischen den Untergraphen  $V_i$  und  $V_j$  möglichst klein bleibt. Dazu wird zunächst die Laplacematrix des Graphen aufgestellt, die Zeilen und Spalten der Laplacematrix werden wie die Knoten des Graphen durchnummeriert:

$$L(G) = (l_{ij}) \quad \text{mit} \quad l_{ij} = \begin{cases} -1 & \text{für} \quad (v_i, v_j) \in E \\ \text{deg}(v_i) & \text{für} \quad i = j \\ 0 & \text{sonst.} \end{cases} \quad (5.5)$$

Die Laplacematrix (5.5) des Graphen ist positiv semidefinit, der zweite Eigenwert ist positiv, sofern der Graph zusammenhängend ist, wie z.B. in [41] untersucht. Der zu diesem Eigenwert gehörende Eigenvektor, der sogenannte Fiedlervektor, wird zur Partitionierung verwendet. Dazu wird der Medianwert der Eigenvektoreinträge ermittelt und es werden die Knoten mit Eigenvektorkomponenten kleineren Wertes einer Partition zugeordnet, die restlichen Knoten der anderen Partition. Durch rekursive Anwendung lassen sich Partitionen mit  $2^n$ ,  $n = 1, 2, \dots$  Teilgebieten erzeugen.

Da die Berechnung des zweiten Eigenwertes (z.B. mit dem Lanczos Algorithmus) sehr rechenintensiv ist, wird auch hier der Partitionierungsvorgang durch eine *multilevel* Strategie beschleunigt, wie es in [3] beschrieben wird. HENDRICKSON und LELAND schlagen in [62] einen modifizierten Algorithmus vor, der auch die Hypercube-Architektur bei der Aufteilung berücksichtigt. Dabei werden nicht nur rekursive Bisektionen, sondern auch

Quadri- und Oktasektionen vorgenommen, wozu jeweils zwei bzw. drei Eigenwerte berechnet werden müssen. Die Anzahl der geschnittenen Kanten ist jedoch nur in Verbindung mit der lokalen Kernighan-Lin Strategie gegenüber dem gewöhnlichen Spektralzerlegungsverfahren verringert.

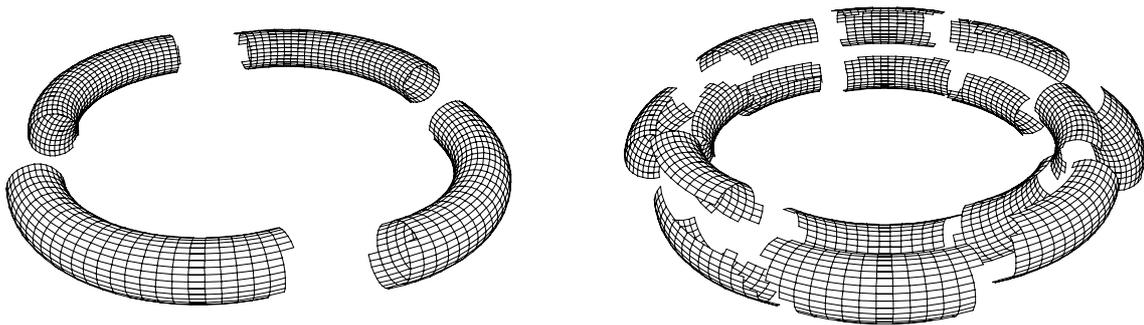


Abbildung 5.11: Spektralzerlegung eines Torus in 4 und 16 Teilgebiete

In Abbildung 5.11 sind die mit dem Spektralbisektionsverfahren aus [3] erhaltenen Aufteilungen des Torus für 4 und 16 Prozessoren dargestellt. Die Minimierung der geschnittenen Kanten des Graphen (Elementkanten des Finite Element Netzes) ist an Hand des Aufteilens quer zur radialen Richtung deutlich zu sehen. Der Algorithmus liefert sehr zuverlässig sehr gute Partitionierungen und ist als Folge der *multilevel* Implementierungen aus [3] und [72] auch in Bezug auf die Rechenzeit durchaus mit den vorher genannten Verfahren konkurrenzfähig.

### Manuelle Aufteilung

Eine Partitionierung kann natürlich auch manuell vorgenommen werden, was für große Probleme allerdings mühsam ist und höchstens mit graphischen Hilfsmitteln durchgeführt werden kann. Eine manuelle Aufteilung kann z.B. durch die Elementnumerierung vorgenommen werden. Indem den Prozessoren immer fortlaufend nummerierte Elemente zugewiesen werden, wird so eine Partitionierung vorgenommen. In Abbildung 5.12 sind die so erhaltenen Partitionen des Torus für 4 und 8 Prozessoren dargestellt, der Torus wird durch die Elementnumerierung immer in radialer Richtung geteilt. Für größere Prozessoranzahlen ist diese Aufteilung als Folge der Forderung nach gleichen Elementanzahlen in den Teilgebieten nicht vollständig radial (siehe Abbildung 5.12 für 8 Prozessoren mittig vorn und hinten). Hier widersprechen sich die beiden Forderungen nach gleichen Elementanzahlen und Minimierung der inneren Ränder. Eine Priorität der ersten Forderung liefert die dargestellte Aufteilung. Eine vollständig radiale Teilung würde in ungleichen Elementanzahlen je Teilgebiet resultieren.

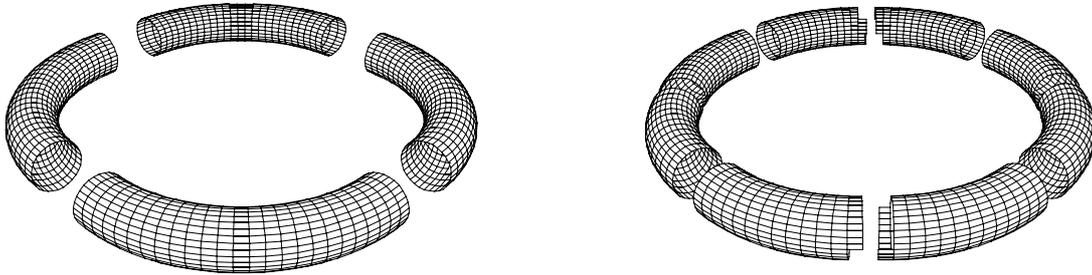


Abbildung 5.12: Manuelle Partitionierung eines Torus in 4 und 8 Teilgebiete

## 5.7 Datenverteilung

Nach erfolgter Gebietszerlegung wird in dem seriellen *preprozessing* Schritt für jeden Prozessor, d.h. für jedes Gebiet, ein separater Finite Element Datensatz generiert. Diese Einzeldatensätze umfassen für das jeweilige Teilnetz Daten, wie z.B. Knotenkoordinaten, Elementtopologien, Randbedingungen usw. Die Numerierung der Elemente und Knoten in den lokalen Netzen beginnt dabei jeweils bei 1. Darüber hinaus müssen zu Ausgabezwecken ebenso Informationen über das globale Problem wie globale Knoten- und Elementnummern hinzugefügt werden. Notwendig zur Lösung des Gesamtproblems sind insbesondere Informationen über die Ränder, die der lokale Datensatz ebenso beinhaltet. So ist für jeden Randknoten eine Liste der Gebiete bzw. Prozessoren enthalten, an denen der Knoten ebenfalls beteiligt ist.

Nach dem Lesen der lokalen Datensätze kann jeder Prozessor für sein Teilgebiet bzw. Teilnetz die lokale Steifigkeitsmatrix  $\mathbf{K}_i$  aufstellen. Die globale Steifigkeitsmatrix kann dann als Summe der  $n_P$  lokalen Steifigkeitsmatrizen dargestellt werden:

$$\mathbf{K} = \sum_{i=1}^{n_P} \mathbf{K}_i. \quad (5.6)$$

Dieser Vorgang kann vollständig parallel ohne jede Kommunikation ablaufen, sofern jeder Prozessor über die aktuellen Verschiebungen verfügt. Ebenso kann die Erstellung der Elementresidualvektoren vollständig parallel erfolgen.

### 5.7.1 Vektoren mit lokalen und globalen Einträgen

Das zuvor beschriebene Vorgehen führt automatisch zu zwei unterschiedlichen Typen von Vektoren: Vektoren mit *globalen* Einträgen und Vektoren mit *lokalen* Einträgen. Bei Vektoren mit *lokalen* Einträgen ergibt analog zu (5.6) erst die Summe über alle Teilgebiete den tatsächlichen Vektoreintrag. Besitzen alle Teilvektoren auf den unterschiedlichen Prozessoren den vollen Eintrag, soll von einem Vektor mit *globalen* Einträgen gesprochen werden.

Ein Vektor, der stets *globale* Einträge enthält, ist der Verschiebungsvektor. Damit die einzelnen Prozessoren ihre lokalen Steifigkeitsmatrizen erstellen können, ist es erforderlich, daß jeder Prozessor über die Verschiebungen aller Knoten des Teilnetzes, also auch die der Randknoten verfügt. Die diesen Randknoten entsprechenden Einträge müssen also auf allen Prozessoren gleichermaßen vorhanden sein.

Bei der Berechnung der Steifigkeitsmatrizen und der Elementresidualvektoren ergeben sich dann aber *lokale* Komponenten, d.h. erst eine Aufsummierung über alle Prozessoren ergibt den tatsächlichen globalen Wert. Um diese Aufsummierung durchzuführen, ist Kommunikation notwendig, die es aber aus Effizienzgründen möglichst zu vermeiden gilt. In [93] ist durch ein geschicktes Arbeiten mit Vektoren mit *globalen* und *lokalen* Einträgen eine effiziente Berechnung der Skalarprodukte im iterativen Lösungsprozess realisiert. Dieses kann effizient auf die Implementierung des parallelen Finiten Elemente Programmes ausgeweitet werden, indem bestimmte Vektoren mit *globalen* und andere mit *lokalen* Einträgen gehalten werden.

### 5.7.2 Last- bzw. Verschiebungsvorgabe

Der Elementresidualvektor, der im Rahmen einer nichtlinearen Newton Iteration die rechte Seite des zu lösenden Gleichungssystems ist, wird automatisch als *lokaler* Vektor berechnet und kann durch die Implementierung des Löser ohne weitere Kommunikation an diesen übergeben werden. Für die Eingabedaten heißt dies aber auch wiederum, daß die Eingabe der äußeren eingepprägten Kräfte ebenfalls *lokalen* Charakter haben muß, d.h. die Summe der Kräfte auf Randknoten muß den tatsächlichen Wert ergeben<sup>8</sup>.

Bei dem verwendeten Programm wird zur Vorgabe von Verschiebungen diese Verschiebung als Last im Lastvektor eingegeben, der Freiheitsgrad aber durch eine Randbedingung gekennzeichnet. In diesem Fall müssen alle Prozessoren die gewünschte Verschiebung erhalten, dieser Wert des Lastvektors darf nicht *lokal* behandelt werden. Mit *globalen* Einträgen der Verschiebungsvorgabe können alle Einträge auf die rechte Seite umgerechnet werden, sodaß sich wiederum eine *lokale* rechte Seite des zu lösenden Gleichungssystems ergibt.

## 5.8 Iterative Gleichungslösung

Zur Lösung linearer dünn besetzter Gleichungssysteme auf Parallelrechnern werden üblicherweise iterative Verfahren bevorzugt, da sie einfacher parallelisiert werden können und effizienter arbeiten. Die hier verwendete parallele Implementierung der aus Abschnitt 4.2 bekannten iterativen Löser ist bereits in [121, 93] und [94] beschrieben und soll daher hier nur kurz vorgestellt werden. Die Schwierigkeit bei der parallelen Implementierung der

---

<sup>8</sup>Wird also auf einen Knoten an einem inneren Rand zwischen zwei Prozessoren die Kraft  $F$  aufgebracht, so darf die Last  $F$  im parallelen Datensatz entweder auf diesen Knoten im Datensatz des ersten Prozessors *oder* im Datensatz des zweiten Prozessors aufgebracht werden. Alternativ ist es möglich, eine Last auf beiden Prozessoren mit dem Betrag von jeweils  $F/2$  aufzubringen.

iterativen Gleichungslösung liegt in der Bestimmung einer wirkungsvollen und dennoch einfach zu berechnenden parallelen Vorkonditionierung.

Die wesentlichen Operationen, die für den cg-, Lanczos bzw. QMR-Algorithmus erforderlich sind, betreffen

- Vektor-Aufdatierungen,
- Skalarprodukte,
- Matrix-Vektor Multiplikationen und das
- Lösen des Vorkonditionierungssystems.

Sobald diese Operationen parallel mit der vorhandenen Datenverteilung durchgeführt werden können, ist auch das parallele Lösen eines linearen Gleichungssystems möglich. Diese Operationen sollen in den nun folgenden Abschnitten am Beispiel des Lanczos Algorithmus aus Tabelle 4.6 kurz beschrieben werden.

### 5.8.1 Vektor-Aufdatierungen

Vektor-Aufdatierungen der Form

$$\mathbf{x}_j = \mathbf{x}_{j-1} + \zeta_j \mathbf{c}_j \quad (5.7)$$

können lokal auf jedem Prozessor ohne Kommunikation ausgeführt werden. Dabei muß der Skalar  $\zeta$  global vorliegen, das Ergebnis hat lokalen oder globalen Charakter abhängig von dem Charakter der aufzuaddierenden Vektoren  $\mathbf{x}$  und  $\mathbf{c}$ .

### 5.8.2 Skalarprodukte

Im vorkonditionierten Lanczos Algorithmus sind zwei Skalarprodukte zu berechnen:

$$\alpha = \mathbf{u}^T \mathbf{z} \quad \text{und} \quad \beta^2 = \mathbf{r}^T \mathbf{t}. \quad (5.8)$$

Wenn nun die Vektoren  $\mathbf{u}$  und  $\mathbf{t}$  *globale* und die Vektoren  $\mathbf{z}$  und  $\mathbf{r}$  *lokale* Einträge besitzen, so können die Skalarprodukte als

$$\alpha = \mathbf{u}^T \mathbf{z} = \sum_{i=1}^{n_P} \mathbf{u}^T \mathbf{z}_i \quad \text{und} \quad \beta^2 = \mathbf{r}^T \mathbf{t} = \sum_{i=1}^{n_P} \mathbf{r}_i^T \mathbf{t} \quad (5.9)$$

geschrieben werden. Es können dann also zunächst lokale Skalarprodukte  $\mathbf{u}^T \mathbf{z}_i$  und  $\mathbf{r}_i^T \mathbf{t}$  gebildet werden, die dann über einen *fan-in* Algorithmus bzw. eine MPI\_ALLREDUCE Operation über alle Prozessoren so aufsummiert werden müssen, daß anschließend jeder Prozessor den globalen Wert für  $\alpha$  bzw.  $\beta^2$  vorliegen hat. Bei der Berechnung von  $\mathbf{u}$ ,  $\mathbf{z}$ ,  $\mathbf{r}$

und  $\mathbf{t}$  ist also darauf zu achten, daß der gewünschte *lokale* oder *globale* Charakter erreicht wird. Für  $\mathbf{u}$  und  $\mathbf{t}$  wird der *globale* Charakter durch einen Kommunikationsschritt erzielt, der *lokale* Charakter von  $\mathbf{r}$  und  $\mathbf{z}$  wird automatisch durch den *lokalen* Charakter der rechten Seite des Gleichungssystems und einer entsprechenden Formulierung des Matrix-Vektor-Produktes erreicht.

### 5.8.3 Matrix-Vektor Multiplikation

Da die Steifigkeitsmatrizen *lokalen* Charakter haben, kann das Matrix-Vektor Produkt

$$\mathbf{z} = \mathbf{K}\mathbf{u} \quad \text{mit} \quad \mathbf{K} = \sum_{i=1}^{n_P} \mathbf{K}_i \quad (5.10)$$

als

$$\mathbf{z} = \sum_{i=1}^{n_P} \mathbf{K}_i \mathbf{u} = \sum_{i=1}^{n_P} \mathbf{z}_i \quad (5.11)$$

geschrieben werden. Es werden also lokale Matrix-Vektor Produkte auf den Prozessoren mit dem *globalen* Vektor  $\mathbf{u}$  gebildet. Als Ergebnis werden *lokale* Vektoren  $\mathbf{z}_i$  erhalten, wie sie zur Berechnung der Skalarprodukte erwünscht sind.

### 5.8.4 Lösen des Vorkonditionierungssystems

Pro Lanczos Iteration ist ein Vorkonditionierungssystem  $\mathbf{M}\mathbf{t} = \mathbf{r}$  zu lösen. Dabei muß die Vorkonditionierungsmatrix symmetrisch und positiv definit sein und das System soll einfach – auch parallel – lösbar sein. Beim parallelen Vorgehen werden Approximationen  $\mathbf{M}_i$  der Matrizen  $\mathbf{K}_i$  berechnet, wobei die in Abschnitt 4.2.2 vorgestellten Verfahren wie Jacobi, SSOR und Varianten der unvollständigen Faktorisierung eingesetzt werden.

Da die lokalen Steifigkeitsmatrizen  $\mathbf{K}_i$  singular sein können, wenn in einem isoliert betrachteten Gebiet noch translatorische bzw. rotatorische Freiheitsgrade enthalten sind, werden die überlappenden Matrixeinträge über die Prozessoren aufaddiert, um positiv definite lokale Steifigkeitsmatrizen zu erhalten. Um dies effizient zu gestalten, werden die Elemente, die innere Randknoten enthalten, zuerst numeriert, um so diese Elementsteifigkeiten zuerst zu berechnen. Nach der Berechnung aller Elementsteifigkeitsmatrizen an den inneren Rändern können alle überlappenden Matrixeinträge asynchron versendet werden. Während des Kommunikationsvorganges können dann die restlichen Elementsteifigkeiten berechnet werden. Das Empfangen und Aufaddieren der Einträge erfolgt erst im Anschluß an die Berechnung der lokalen Steifigkeitsmatrizen. Auf diese Weise kann der Aufwand zur Globalisierung der Matrixeinträge hinter der Berechnung der Steifigkeitsmatrix versteckt werden (*latency hiding*), da sich im Allgemeinen sehr viel mehr Elemente im Innern des Netzes befinden, als an den Rändern. Zur Matrix-Vektor Multiplikation wird weiterhin die lokale Steifigkeitsmatrix verwendet.

Vollständig parallel kann anschließend die Berechnung der lokalen Vorkonditionierungsmatrizen  $M_i$  und das Lösen der lokalen Vorkonditionierungsprobleme  $M_i t_i = r_i$  erfolgen. An den überlappenden Knotenfreiheitsgraden werden auf unterschiedlichen Prozessoren unterschiedliche Werte berechnet, sodaß eine Vor- und Nachgewichtung, wie sie in [93] bzw. [121] beschrieben ist, durchgeführt werden muß. Als vorteilhafteste Gewichtung hat sich eine Mittelung, d.h. eine Gewichtung

$$M^{-1} = \sum_{i=1}^{n_P} E_i M_i^{-1} E_i \quad (5.12)$$

mit positiv definiten Diagonalmatrizen  $E_i$  herausgestellt, wobei die Diagonaleinträge in  $E_i$  Null sind, sofern die Zeilen in  $M_i$  Null-Vektoren sind und ansonsten den Wert  $\frac{1}{\sqrt{m}}$  besitzen. Dabei ist  $m$  die Anzahl der Prozessoren, die den betreffenden Knoten gemeinsam besitzen. Mit dieser Wahl wird die Beziehung

$$\sum_{i=1}^{n_P} E_i^2 = I \quad (5.13)$$

erfüllt. Andere Gewichtungsmöglichkeiten und Vergleiche zwischen den einzelnen Schemata werden ausführlich in [93] und [94] diskutiert. Es ist zu Beachten, daß bei der praktischen Durchführung dieser Gewichtung erneut ein Kommunikationsschritt notwendig wird.

## 5.9 Weitere Aspekte der Parallelisierung

### 5.9.1 Details zur Globalisierung der lokalen Steifigkeitsmatrizen

Wie bereits in Abschnitt 5.8.4 geschildert, müssen die lokalen Steifigkeitsmatrizen zur Vorkonditionierung globalisiert werden, d.h. es müssen alle Matrixeinträge, die den inneren Rändern zwischen den Prozessoren entsprechen, über alle angrenzenden Prozessoren aufsummiert werden. Dabei gibt es zum einen Matrixeinträge, die sich für die Knoten an den inneren Rändern ergeben und Matrixeinträge, die von den Kanten entlang den inneren Rändern stammen. Die Knoten an den inneren Rändern sind bereits in den Datensätzen gekennzeichnet (siehe Abschnitt 5.7), die inneren Randkanten müssen aber noch bestimmt werden. Da Letztere vom Elementtyp abhängig sind, muß ein Algorithmus gefunden werden, der unabhängig vom Elementtyp unter Kenntnis der Randknoten die Randkanten ermittelt. Es wird also in jedem Fall Information über den Elementtyp benötigt, die zusätzlich (z.B. über Kennungen zu Postprocessing-Zwecken) verfügbar sein muß. Dann können, wie aus Abbildung 5.13 und Tabelle 5.1 ersichtlich ist, für jeden Knoten des jeweiligen Elementtyps alle benachbarten Knoten (und damit die möglichen Kanten) bestimmt werden. Bei einem ebenen 4-knotigen Element ist beispielsweise der Knoten 2 immer mit den Knoten 1 und 3 durch eine Kante verbunden, bei einem 8-knotigen Scheibenelement stets mit den Knoten 5 und 6, bei einem 8-knotigen Volumenelement mit den Knoten 1, 3 und 6

ebenes 4-Knoten Element	ebenes 8-Knoten Element	8-Knoten Volumenelement	10-Knoten Tetraederelement
1   2   4	1   5   8	1   2   4   5	1   5   7   8
2   1   3	2   5   6	2   1   3   6	2   5   6   9
3   2   4	3   6   7	3   2   4   7	3   6   7   10
4   1   3	4   7   8	4   1   3   8	4   8   9   10
	5   1   2	5   1   6   8	5   1   2
	6   2   3	6   2   5   7	6   2   3
	7   3   4	7   3   6   8	7   1   3
	8   4   1	8   4   5   7	8   1   4
			9   2   4
			10   3   4

Tabelle 5.1: Mögliche Randkanten für verschiedene Elementtopologien

und bei einem 10-knotigen Tetraederelement mit den Knoten 5, 6 und 9. Sind nun zwei Knoten Randknoten zum selben Gebiet und existiert eine Elementkante zwischen diesen beiden Knoten, so handelt es sich um eine Randkante und es müssen die entsprechenden Matrixeinträge aufaddiert werden.

Für ein einfaches Beispiel mit einem geraden Rand zwischen zwei Gebieten ist der Zusammenhang zwischen Randknoten bzw. -kanten und den Matrixelementen in Abbildung 5.14 dargestellt. Zur größeren Übersichtlichkeit wird angenommen, daß es sich um 4-knotige Scheibenelemente mit zwei Freiheitsgraden pro Knoten handelt. Im linken Teilgebiet haben die drei betrachteten Randknoten die Nummern 17, 18 und 19, im rechten Teilgebiet (grau unterlegt) 1, 2 und 3. In jedem Teilgebiet haben die Knoten im Allgemeinen unterschiedliche Nummern, die Reihenfolge der Numerierung ist aber immer gleich. Weist ein Knoten im seriellen Ausgangsdatensatz eine größere Nummer auf als ein zweiter Knoten, so ist dies auch im parallelen Datensatz der Fall. Dadurch sind die sich entsprechenden Randknoten eindeutig in den Teilgebieten identifizierbar. Den Freiheitsgraden der Knoten 17, 18 und 19 im linken Teilgebiet seien die Gleichungsnummern 51, 52, 53, 54 und 55 zugeordnet. Ein Freiheitsgrad des Knotens 19 sei festgehalten. In dem rechten Teilgebiet entsprechen den selben Knoten 1, 2 und 3 die Freiheitsgrade 1, 2, 3, 4 und 5. Im linken Teilgebiet können nun die dem inneren Rand entsprechenden Matrixelemente be-

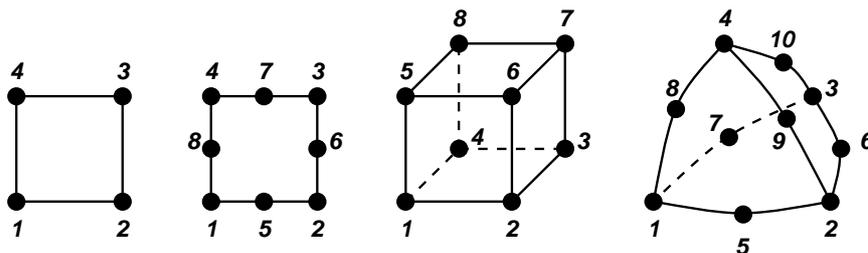


Abbildung 5.13: Verschiedene Elementtopologien

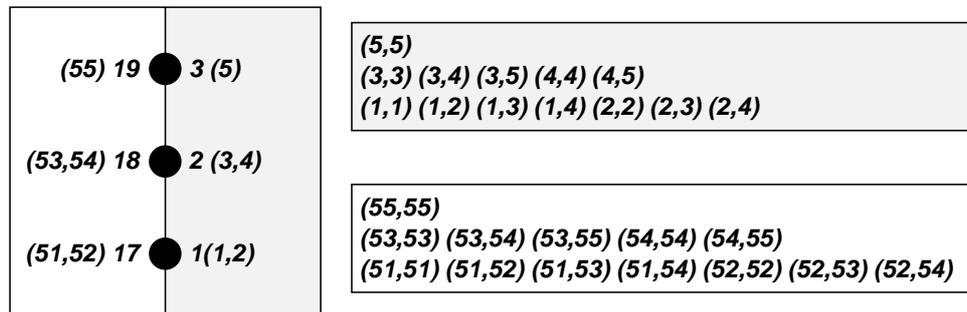


Abbildung 5.14: Randkanten und entsprechende Matrixelemente

ginnend mit dem ersten Freiheitsgrad des Knotens 17 bestimmt werden. Da die Steifigkeitsmatrix symmetrisch ist, müssen nur Matrixelemente im oberen Dreieck berücksichtigt werden. Dem ersten Freiheitsgrad des Knotens 17 entsprechen die Matrixeinträge  $(51,51)$  und  $(51,52)$ , der Randkante zum Knoten 18 die Matrixelemente  $(51,53)$  und  $(51,54)$ . Dem zweiten Freiheitsgrad entsprechen die Matrixelemente  $(52,52)$ ,  $(52,53)$  und  $(52,54)$ , aus Symmetriegründen muß der Matrixeintrag  $(52,51)$  kein zweites Mal verschickt werden. Ebenso können im linken Teilgebiet die weiteren Matrixelemente bestimmt werden. Die entsprechenden Matrixeinträge im rechten Teilgebiet können Abbildung 5.14 entnommen werden.

### 5.9.2 Statische nichtlineare Finite Element Berechnungen

Bei nichtlinearen, statischen Berechnungen unter Verwendung des Newton-Raphson Verfahrens muß in jedem Lastschritt mindestens eine Gleichungslösung durchgeführt werden. Dafür wird zunächst eine neue tangentielle Steifigkeitsmatrix erstellt. Da der iterative Löser auf jedem Prozessor die *globalen* Lösungskomponenten liefert, kann die Erstellung ohne Kommunikation erfolgen. Gleiches gilt für die Erstellung des Residualvektors, der anschließend *lokalen* Charakter besitzt, was zur Lösung des nächsten inkrementellen Schrittes korrekt ist, da der Löser eine *lokale* rechte Seite verlangt. Auch das Aufdatieren der Verschiebungen kann lokal ablaufen, da alle berechneten Verschiebungen stets *globalen* Charakter besitzen. Ebenso läuft die Durchführung einer Verschiebungssteuerung ohne zusätzlichen Aufwand ab, da der Verschiebungsvektor stets *globalen* Charakter hat, und wie zuvor in Abschnitt 5.7.2 geschildert, die Lastvorgabe auch *global* vorgegeben ist.

Bei Verwendung des Bogenlängenverfahrens hingegen, siehe Abschnitt 2.3.2, sind weitere Skalarprodukte zu berechnen, die Kommunikation erfordern. Dabei ist es nicht (wie beim iterativen Gleichungslöser) der Fall, daß einer der beiden Vektoren *lokalen* und der andere *globalen* Charakter hat. Deshalb muß Letzteres gegebenenfalls erzwungen werden. Haben beide Vektoren *globalen* Charakter, kann einer der beiden Vektoren durch Multiplikation mit der Gewichtungsmatrix  $E_i$  in einen Vektor mit *lokalem* Charakter überführt werden. Dann kann das Skalarprodukt, wie in Abschnitt 5.8.2 beschrieben, berechnet werden. Haben dagegen beide Vektoren *lokalen* Charakter, so müssen die Komponenten eines Vektors über alle Prozessoren aufsummiert werden.

## 5.10 Parallele Vergleichsberechnungen

In diesem Abschnitt werden einige Ergebnisse für die in Abschnitt 3 vorgestellten Probleme präsentiert, die mit der in den vorigen Abschnitten dargestellten Vorgehensweise erzielt wurden.

Zunächst wird ein lineares, anschließend ein nichtlineares Problem betrachtet.

### 5.10.1 Lineares Problem: Zahnkrone

Zunächst wird das Netz für das Problem aus Abschnitt 3.1 mittels des Spektralbisektionsverfahrens auf die Prozessoren verteilt, siehe Abbildung 5.15. Die Verteilung der Elemente auf die Prozessoren erfolgt hierbei unabhängig von den Materialeigenschaften im ursprünglichen Netz.

Die parallelen Berechnungen wurden auf einer IBM SP2 mit bis zu 32 Prozessoren durchgeführt. Zur iterativen Lösung wurde das cg-Verfahren mit Jacobi- bzw. MPILU-Vorkonditionierung eingesetzt. In Abbildung 5.16 ist der inkrementelle *speedup* dargestellt; beim Einsatz von doppelt so vielen Prozessoren wird eine Halbierung der Rechenzeit angestrebt, d.h. ein inkrementeller *speedup* von Zwei. Dies ist bei diesem Problem bis zu 32 Prozessoren nahezu durchgängig erfüllt. Der inkrementelle *speedup* liegt stets zwischen 1.7 und 2.0.

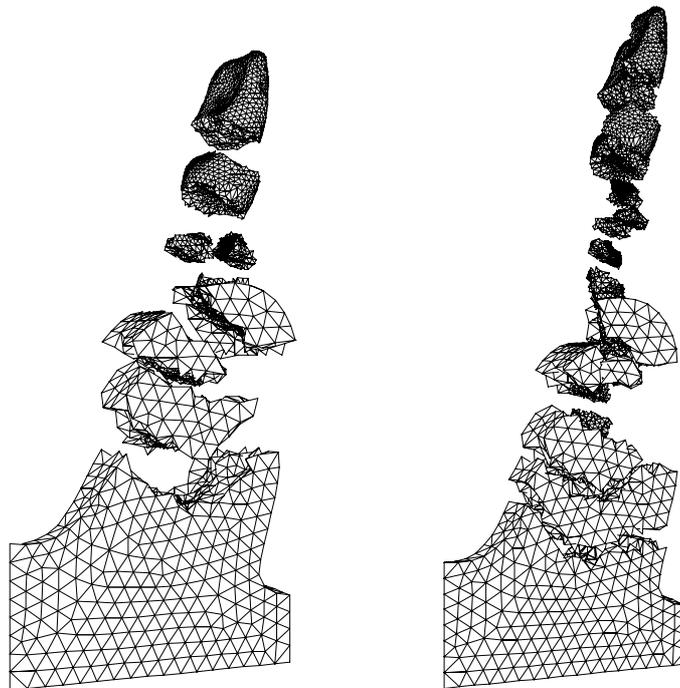
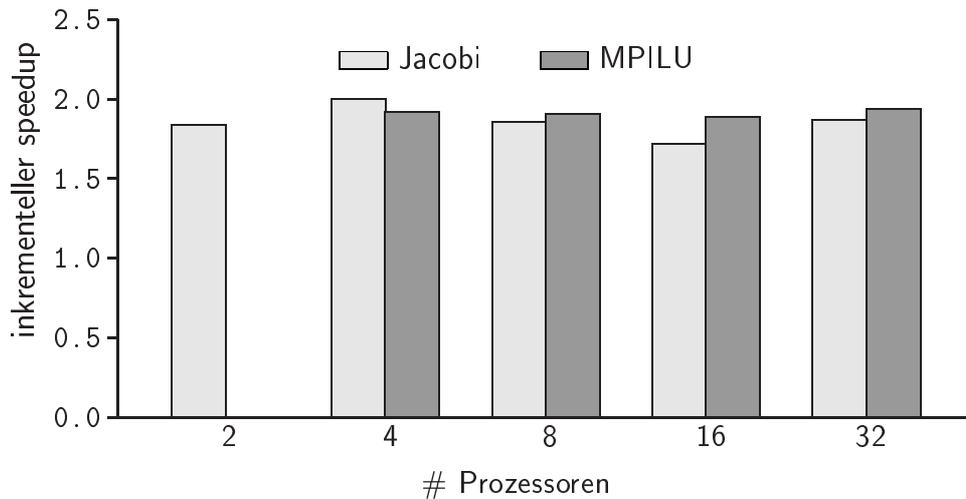


Abbildung 5.15: Zahnkrone – Gebietszerlegung für 8 und 16 Prozessoren (Spektralbisektion)

Abbildung 5.16: Inkrementeller *speedup* für die Zahnkrone

Für die MPILU-Vorkonditionierung konnte kein serielles Ergebnis erzielt werden, da hierfür mehr als 250 MByte Hauptspeicher notwendig wären, die nicht verfügbar waren. Daher wird die benötigte Zeit für die Rechnung auf zwei Prozessoren zum Vergleich genutzt.

Es ist feststellbar, daß das Verfahren mit der MPILU-Vorkonditionierung besser skaliert als mit Jacobi-Vorkonditionierung, trotz seines lokalen Charakters. Darüber hinaus führt diese Vorkonditionierung auch zur absolut schnelleren Lösung des Problems (bei 32 Prozessoren: Jacobi 115 s, MPILU 75.3 s). Der leicht überlineare *speedup* bei 4 Prozessoren für die Jacobi Vorkonditionierung ist mit hoher Wahrscheinlichkeit das Ergebnis von Verzögerungen beim Schreiben von Ausgabedaten auf die Festplatte bei der Berechnung mit zwei Prozessoren.

### 5.10.2 Nichtlineares Problem: Rohrkreuz

Für die komplette nichtlineare Berechnung des Rohrkreuzes sind die Rechenzeiten auf der IBM SP2 für unterschiedliche Prozessorzahlen und unterschiedliche Vorkonditionierungsstrategien in Abbildung 5.17 dargestellt. Die Berechnungen auf einem Prozessor wurden zwar alle auf dem gleichen Rechner durchgeführt wie die parallelen, jedoch ist aus Abbildung 5.17 ersichtlich, daß die seriellen Berechnungen verglichen mit der parallelen Durchführung auf zwei Prozessoren unerwartet lange Rechenzeiten benötigen. Da in der parallelen Programmversion (dies gilt insbesondere für die Jacobi Vorkonditionierung) jedoch das serielle Verfahren äquivalent abläuft, ist der beobachtete superlineare *speedup* nicht das Ergebnis des Algorithmus, sondern die Ursache hierfür muß in der unterschiedlichen Behandlung paralleler und serieller Berechnungen seitens der Systemadministration vermutet werden. Dieser Effekt wurde erst nach einer Erweiterung des eingesetzten Parallelrechners beobachtet, daher ist nur dieses Ergebnis hiervon betroffen. Die Angabe eines *speedup* ist hier demnach nicht sinnvoll.

Bei einer Verdoppelung der Prozessoranzahl ist in Abbildung 5.17 über weite Bereiche

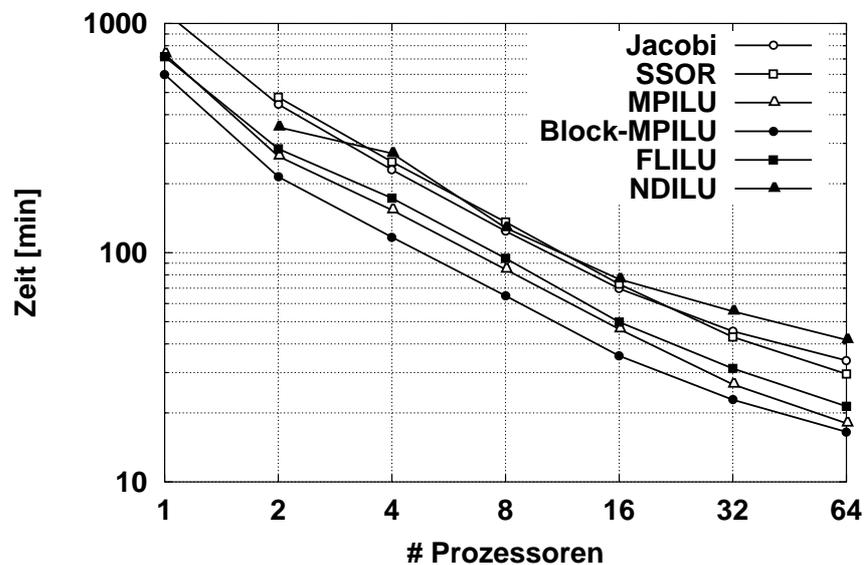


Abbildung 5.17: Rechenzeiten für Rohrkreuz auf IBM SP2 mit unterschiedlichen Prozessorzahlen

jeweils eine nahezu lineare Abnahme in der doppelt logarithmischen Darstellung der Rechenzeit bis zu 16 Prozessoren zu beobachten. Es ist also ein nahezu konstanter inkrementeller *speedup* festzustellen, der für eine Verdoppelung der Prozessoranzahl etwa bei 1.8 liegt. Bei der Verwendung höherer Prozessoranzahlen verschlechtert sich diese Eigenschaft zunehmend. Ursache hierfür ist die konstante Problemgröße, d.h. die Teilgebiete, die von den einzelnen Prozessoren bearbeitet werden, werden mit steigender Anzahl von Prozessoren immer kleiner – der Berechnungsaufwand sinkt, im Verhältnis dazu steigt jedoch der Kommunikationsaufwand aufgrund der inneren Ränder.

Bei diesem Beispiel erweist sich die Block-MPILU-Vorkonditionierung, ähnlich wie bei der seriellen Vergleichsberechnung (siehe Abbildung 4.14), als die effizienteste Wahl, gefolgt von der MPILU- und FLILU- Vorkonditionierung. Auch bei der parallelen Bearbeitung ist die NDILU-Vorkonditionierung nicht konkurrenzfähig, obwohl die Bestimmung einer Besetzungsstruktur mit immer kleiner werdenden, lokalen Teilmatrizen deutlich effizienter sein sollte.

Insgesamt ist zu bemerken, daß die Gesamtrechenzeit dieses Problems von ca. 10 Stunden auf nur noch etwa 20 Minuten reduziert werden konnte. Auch im Hinblick auf die direkten Lösungsverfahren (siehe Abbildung 4.14) sind die hier mit den iterativen Verfahren erzielten Rechenzeiten ausgezeichnet.

## 5.11 Wertung der Resultate

Die in Abschnitt 5.10 ermittelten Ergebnisse zeigen, daß mit der in der vorliegenden Abhandlung vorgestellten Parallelisierungsstrategie eine effiziente Durchführung paralleler

---

nichtlinearer Strukturanalysen möglich ist. Insbesondere für die Problemklassen, für die iterative Löser vorteilhaft sind, d.h. gut konditionierte Kontinuumsprobleme, kann bei entsprechend großen Systemen ein praktisch optimaler *speedup* erreicht werden. Bei schlechter konditionierten Schalenproblemen ist zwar gegenüber der sequentiellen iterativen Lösung ebenfalls ein guter *speedup* erzielbar, im Vergleich zu direkten Lösungsstrategien bzw. gemischt iterativ-direkten Verfahren (siehe Abschnitt 4.4.3) verliert die parallele Version mit dem rein iterativen Gleichungslöser an Effizienz, siehe hierzu auch [123]. Dort wird dies an der statischen axialen Belastung eines imperfekten Zylinders gezeigt. Nur mit dem Einsatz vieler Prozessoren kann überhaupt eine Reduktion der Berechnungszeit gegenüber der sequentiellen Berechnung erreicht werden. Dies ist aber nicht die Folge der parallelen Implementierung. Dieses Verhalten kann nämlich auch bei rein sequentiellen iterativen Berechnungen beobachtet werden. Für solche extrem schlecht konditionierten Probleme ist der hier vorgestellte iterative Gleichungslöser ungeeignet. Eine sequentielle Berechnung mit direkten bzw. gemischt iterativ-direkten Lösungsverfahren oder auch eine Mehrgitter- oder Multilevelstrategie ist für solche Probleme zu bevorzugen.

## 6 Einige numerische Aspekte bei Finiten Element Berechnungen

In diesem Abschnitt werden rein numerische Aspekte bei der Lösung von Finiten Element Problemen anhand zweier Beispiele diskutiert. Speziell handelt es sich um Implementierungsfragen. Bei dieser Fragestellung soll gezeigt werden, daß diese für das jeweils untersuchte Verfahren von hoher Bedeutung sind, da die Bewertung – sowohl bezüglich Funktionsfähigkeit als auch bezüglich Effizienz des Verfahrens – in hohem Maße davon betroffen ist. Im ungünstigsten Falle könnte das Verfahren als solches als nicht funktionsfähig verworfen werden, obwohl es nur nachlässig bzw. ungünstig implementiert wurde. Interessant in diesem Zusammenhang ist insbesondere, daß allein die Parallelisierung eines Algorithmus das numerische Ergebnis ändern kann und schlimmstenfalls zum Versagen der parallelen Version führen kann. Dieser Fall des Versagens, wie z.B. der Verlust von Konvergenz, kann vom Anwender erkannt und die Berechnung entsprechend bewertet werden. Fälle, in denen jedoch numerisch unterschiedliche Ergebnisse aus den Berechnungen resultieren, fallen allerdings nur dann auf, wenn sie der Erfahrung des Ingenieurs grob widersprechen und sind deshalb besonders “gefährlich”. Aus diesem Grund sollten numerische Aspekte bei der Diskussion von erzielten Ergebnissen verstärkt beachtet werden, um auch numerische Sensitivitäten der Verfahren bzw. Algorithmen offenzulegen.

Im folgenden Abschnitt wird zuerst die statische Kondensation von sogenannten EAS-Parametern [133] untersucht, was sowohl für sequentielle als auch für parallele Berechnungen von Interesse ist. Ein rein paralleles Problem folgt mit der Betrachtung der parallelen Berechnung des Skalarproduktes.

### 6.1 Über die Kondensation von EAS-Parametern bei “Solid Shell” Elementen

Eine Methode zur Vermeidung unerwünschter Versteifungseffekte stellt die sogenannte EAS-Methode<sup>9</sup> dar, die auf SIMO und RIFAI [133] zurückgeht. In [61] wurde diese Methode auch auf die in [60] entwickelten “Solid Shell” Elemente angewendet, die im Gegensatz zu den auf dem degenerierten Schalenkonzept basierenden Elementen nur mit Verschiebungsfreiheitsgraden formuliert sind. Bei den numerischen Untersuchungen für nahezu inkompressibles Materialverhalten traten unerwartet starke Konvergenzprobleme innerhalb des Newton-Raphson Verfahrens auf. Letztere stellten sich als Folge einer Kombination aus ungeschickter Implementierung und numerischen Einflüssen heraus.

Dazu wird kurz die Behandlung der EAS-Parameter eingegangen, wie sie in [60] geschildert ist, um anschließend numerisch unterschiedliche Lösungsprozeduren zu formulieren. Anhand eines einfachen Beispiels werden dann die verschiedenen Algorithmen miteinander verglichen. Um das Konvergenzverhalten der EAS-Elemente einzuschätzen, findet ein weiterer Vergleich mit dem EAS3Dq Element aus [60] statt, das in Dickenrichtung ohne

---

<sup>9</sup>Enhanced Assumed Strain.

die erweiterte Verzerrungen (EAS) formuliert ist.

### 6.1.1 Kondensation innerer EAS-Parameter

Bei den EAS-Parametern handelt es sich um innere Elementvariablen, die für eine Lösung vorab auskondensiert werden können. Beim sogenannten Auskondensieren wird eine Block-Gauß Elimination ausgeführt, die im Sprachgebrauch der Mechanik als “statische Kondensation” bezeichnet wird (siehe z.B. [66]) und in der Mathematik als “Schur-Komplement” bekannt ist [22].

Die Elementsteifigkeitsmatrix kann formuliert werden als [60]

$$\bar{\mathbf{K}}_{EAS}^e = \begin{bmatrix} \mathbf{K}^e & \mathbf{L}^{eT} \\ \mathbf{L}^e & \mathbf{D}^e \end{bmatrix}. \quad (6.1)$$

Dabei ist  $\mathbf{K}^e$  die Elementsteifigkeitsmatrix, zu der die Verschiebungsfreiheitsgrade  $\mathbf{d}^e$  gehören, und  $\mathbf{D}^e$  ist die EAS-Matrix, der die EAS-Parameter  $\alpha$  zugeordnet sind. Die Kopplung wird durch die Matrix  $\mathbf{L}^e$  bzw.  $\mathbf{L}^{eT}$  beschrieben. Die Bezeichnungen dieser Matrizen entstammen [60] und sind nicht zu verwechseln mit den in Kapitel 4 verwendeten Symbolen für die Faktoren bei der direkten Gleichungslösung.

Die Ausführung der statischen Kondensation führt auf die tangentielle Elementsteifigkeitsmatrix

$$\mathbf{K}_T^e = \mathbf{K}^e - \mathbf{L}^{eT} \mathbf{D}^{e-1} \mathbf{L}^e \quad (6.2)$$

und den Elementresidualvektor

$$\mathbf{p}^e = \mathbf{p}_{ext}^e - \mathbf{r}^e + \mathbf{L}^{eT} \mathbf{D}^{e-1} \mathbf{p}_{EAS}^e. \quad (6.3)$$

Der Vektor  $\mathbf{p}_{EAS}^e$  ergibt sich aus der nichtlinearen Formulierung der EAS-Methode, siehe z.B. [60]. Das Aufdatieren der internen EAS-Parameter  $\alpha$  in der Iteration  $k+1$  erfolgt über

$$\alpha_{k+1}^e = \alpha_k^e - \mathbf{D}^{e-1} (\mathbf{L}_k^e \Delta \mathbf{d}_k^e + \mathbf{P}_k^e). \quad (6.4)$$

Für die Bedeutung der Matrix  $\mathbf{P}^e$  wird auf [60] verwiesen.

Alternativ kann aber auch die Matrix (6.1) als Elementsteifigkeitsmatrix in das globale Gleichungssystem eingebaut werden. Die EAS-Parameter werden dann als globale Freiheitsgrade betrachtet, das Aufdatieren erfolgt analog zu den Verschiebungsfreiheitsgraden. Diese Variante soll anschließend zum Vergleich dienen.

### 6.1.1.1 Algorithmus EAS-1

In einem ersten Algorithmus wird zunächst die Matrix  $D^{e-1}$  berechnet, die ebenfalls für die Gleichungen (6.2) und (6.3) benötigt wird. Die Aufdatierung der EAS-Parameter  $\alpha$  erfolgt durch

$$\alpha_{k+1}^e = \alpha_k^e - D^{e-1} L_k^e \Delta d_k^e - D^{e-1} P_k^e, \quad (6.5)$$

vergleiche Gleichung (6.4). Offensichtlich ist die Aufdatierung nach Gleichung (6.5) weniger effizient als nach Gleichung (6.4). Durch die Berechnung nach Gleichung (6.5) treten aber einige numerische Eigenschaften deutlicher hervor. In Abschnitt 6.1.3 wird dieser Sachverhalt nochmals diskutiert. Dieses Vorgehen wird im Folgenden als ‘‘EAS-1’’ bezeichnet.

### 6.1.1.2 Algorithmus EAS-2

In einer alternativen Vorgehensweise wird die Gleichung (6.4) wie folgt umgestellt. Zunächst wird der Term  $L_k^e \Delta d_k^e + P_k^e$  ausgewertet und in einem Vektor  $x$  abgespeichert. Daraus resultiert die Gleichung

$$\alpha_k^e - \alpha_{k+1}^e = D^{e-1} x, \quad (6.6)$$

die mit  $D^e$  von links multipliziert werden kann und

$$D^e (\alpha_k^e - \alpha_{k+1}^e) = x \quad (6.7)$$

liefert. Das Gleichungssystem (6.7) kann mittels eines Gauß’schen Eliminationsverfahrens aufgelöst werden. Zur Erhöhung der numerischen Genauigkeit kann dabei auch noch eine numerische Pivotsuche erfolgen. Diese Variante wird im Folgenden als ‘‘EAS-2’’ bezeichnet.

In gleichem Sinne läßt sich auch die Berechnung von (6.2) und (6.3) umformulieren. Dabei wurden aber keine numerischen Unterschiede bemerkt. Die wesentlichen Unterschiede zwischen den zwei vorgeschlagenen Algorithmen liegen in der Vermeidung eines in Algorithmus EAS-1 notwendigen zusätzlichen Matrix-Matrix Produktes und in dem Ersetzen von Multiplikationen mit der invertierten Matrix  $D^{e-1}$  durch das Lösen linearer Gleichungssysteme.

## 6.1.2 Vergleich der Algorithmen EAS-1 und EAS-2 am Beispiel eines Kragarmes

Für numerische Untersuchungen wird der in [61] verwendete Kragarm unter Einsatz der EAS3DEAS Elementformulierung herangezogen, siehe Abbildung 6.1.

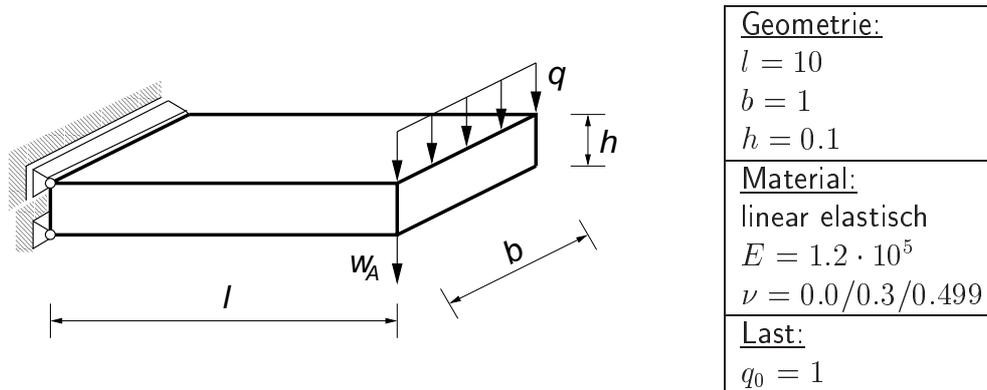


Abbildung 6.1: Geometrie und Materialdaten des Kragarmes

Zum Vergleich wird auch die oben angesprochene Möglichkeit herangezogen, die inneren EAS-Parameter in das globale Gleichungssystem zu integrieren. Der Nachteil ist hierbei in einem (nicht unerheblich) größeren Gesamtproblem zu sehen, was insbesondere für parallele Berechnungen nachteilig ist, da Berechnungen in den Elementen vollständig parallel ablaufen, wohingegen zur Lösung des linearen Gleichungssystems Kommunikation erforderlich ist. Dieses Vorgehen wird als "ohne Kondensation" bezeichnet.

In Abbildung 6.2 ist der Last-Verformungspfad der Durchbiegung  $w_A$  für unterschiedliche Werte von  $\nu$  bei Berechnung ohne Kondensation angegeben, in den Abbildungen 6.3 und 6.4 sind die Last-Verformungspfade für die Algorithmen EAS-1 und EAS-2 dargestellt. Hierbei wurde in allen Fällen übereinstimmend das in [61] angegebene Ergebnis erreicht.

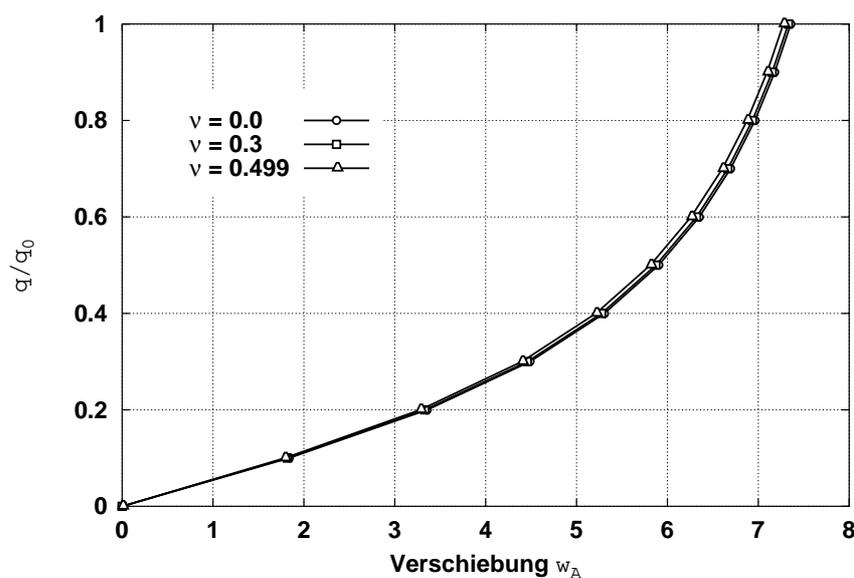


Abbildung 6.2: Last-Verformungspfade für Berechnung ohne Kondensation

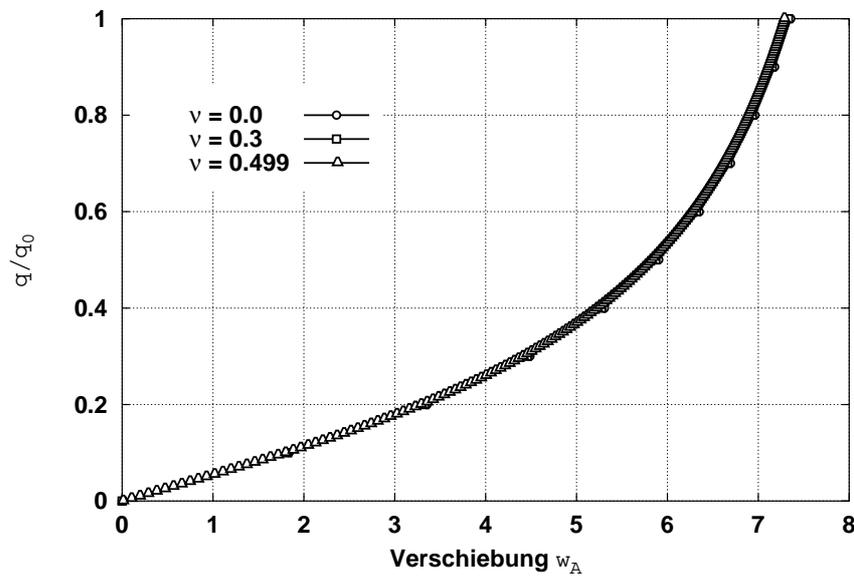


Abbildung 6.3: Last-Verformungspfade für Berechnung mit Algorithmus EAS-1

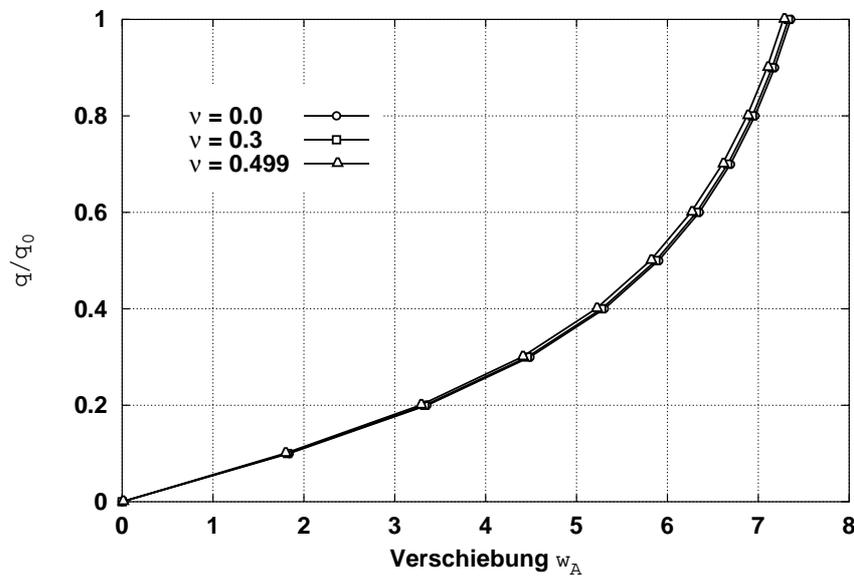
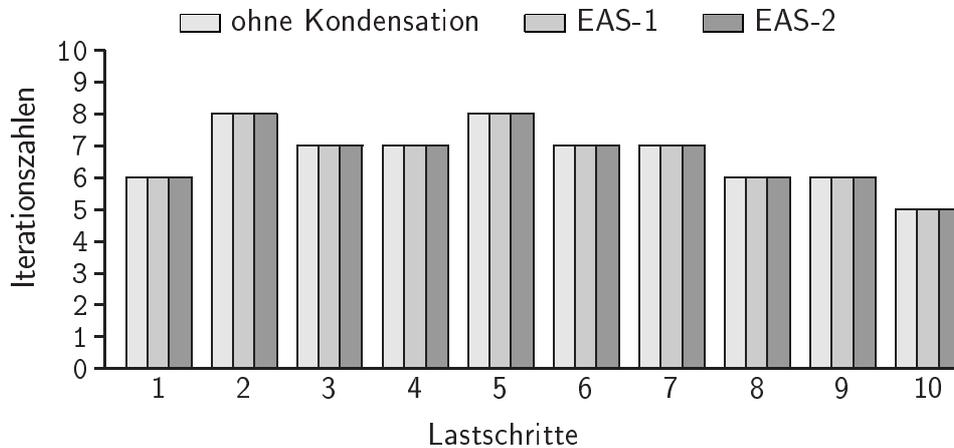


Abbildung 6.4: Last-Verformungspfade für Berechnung mit Algorithmus EAS-2

Für den Algorithmus EAS-1 mußte bei  $\nu = 0.499$  die Schrittweite jedoch auf  $1/50$  reduziert werden, um Konvergenz zu erzielen, siehe Abbildung 6.3. Dennoch waren zusätzlich zum Teil über 50 Iterationen pro Lastschritt notwendig.

Deshalb soll das unterschiedliche Konvergenzverhalten der unterschiedlichen Algorithmen im folgenden Abschnitt genauer betrachtet werden.

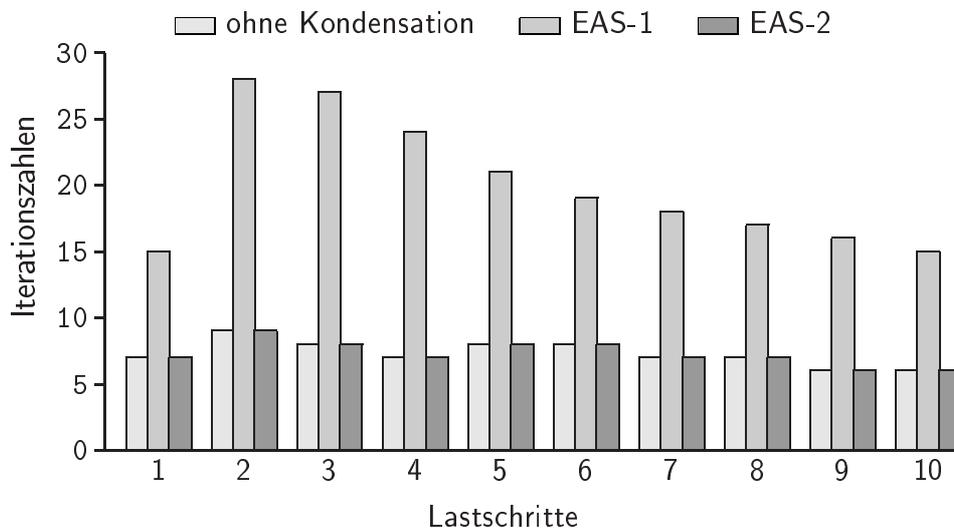
## 6.1.2.1 Konvergenzverhalten

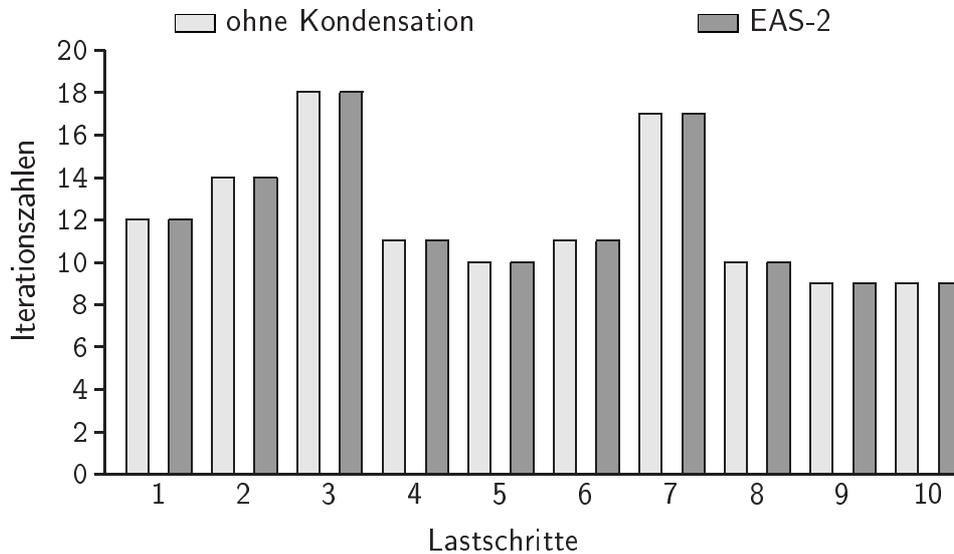
Abbildung 6.5: Iterationszahlen für  $\nu = 0.0$ 

In Abbildung 6.5 ist die Anzahl der Iterationen für jeden der 10 Lastschritte für die drei Berechnungsvarianten und  $\nu = 0.0$  abgebildet. Es ist zu sehen, daß sich die drei Varianten nicht unterscheiden – weder in der Lösung, noch im numerischen Verhalten.

In Abbildung 6.6 sind die Iterationen pro Lastschritt für  $\nu = 0.3$  abgebildet. Hier zeigt sich ein deutlicher Anstieg der Iterationen bei dem mit EAS-1 bezeichneten Algorithmus. Die Iterationszahlen der Berechnung ohne Kondensation und EAS-2 unterscheiden sich hingegen nicht.

In Abbildung 6.7 sind die Iterationen pro Lastschritt für  $\nu = 0.499$ , also den nahezu in-

Abbildung 6.6: Iterationszahlen für  $\nu = 0.3$

Abbildung 6.7: Iterationszahlen für  $\nu = 0.499$ 

kompressiblen Fall, abgebildet. Hier sind keine Angaben für EAS-1 enthalten, da dafür nur mit einem 50-fach kleineren Lastschritt eine Lösung erzielt werden konnte. Es sei bemerkt, daß für das EAS-1 Vorgehen auch deutlich mehr Iterationen pro Lastschritt benötigt werden, als für die beiden anderen Varianten. Die Iterationszahlen dieser beiden letztgenannten Varianten unterscheiden sich auch für hohe Werte von  $\nu \rightarrow 0.5$  nicht. Dabei wurde für den Algorithmus EAS-2 weder zur Lösung der Gleichung (6.7), noch zur Invertierung von  $D^e$  eine numerische Pivotsuche eingesetzt.

### 6.1.2.2 Vergleich mit dem ANS3Dq Element

In den Abbildungen 6.5 bis 6.7 steigen die benötigten Iterationen mit wachsender InkompRESSIBILITÄT. Um zu entscheiden, ob dieses Verhalten spezifisch für Elemente nach dem EAS-Konzept ist, werden zum Vergleich für  $\nu = 0.0$ ,  $0.3$  und  $0.499$  die Iterationszahlen pro Lastschritt für ein ANS3Dq Element [60] angegeben, bei denen die untersuchte Kondensation von EAS-Parametern nicht erforderlich ist, siehe Abbildung 6.8.

Auch für dieses Element steigt die Anzahl der Iterationen in etwa dem gleichen Umfang an, wie es auch bei dem untersuchten EAS3DEAS Element der Fall ist. Dieser Anstieg ist folglich keine Folge der EAS-Implementierung.

### 6.1.3 Zusammenfassung und Diskussion

In diesem Abschnitt wurden drei Varianten zur Behandlung von EAS-Parametern eingeführt. Eine Variante besteht darin, die EAS-Parameter nicht zu kondensieren sondern als globale Freiheitsgrade zu betrachten und im globalen Gleichungssystem zu lösen. Die-

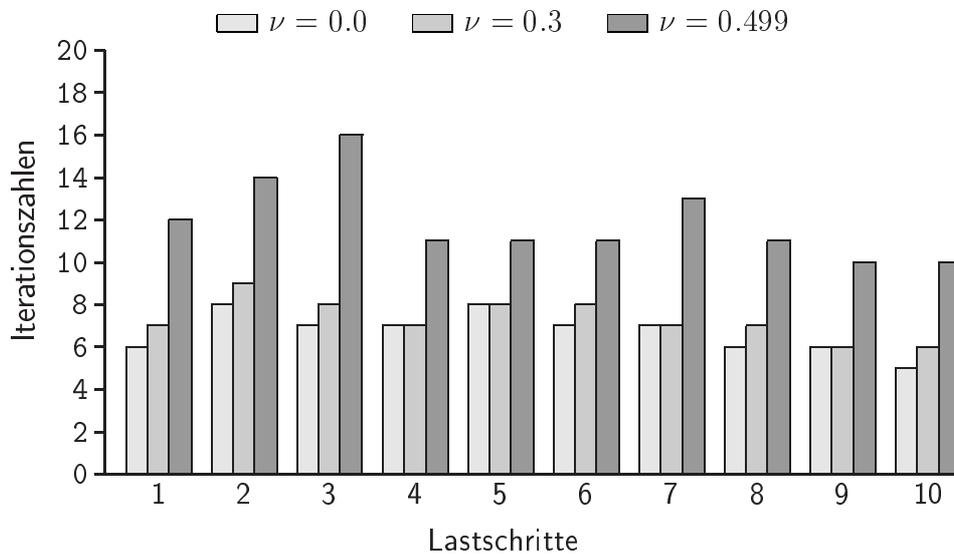


Abbildung 6.8: Iterationszahlen für ANS3Dq Element

se Variante ist aus Effizienzgründen weniger attraktiv als die statische Kondensation. Zwei auf einer statischen Kondensation basierende Algorithmen unterscheiden sich nur hinsichtlich der numerischen Behandlung der Aufdatierung der EAS-Parameter. Dennoch treten hier erhebliche numerische Unterschiede auf, die ihre Ursache in zusätzlichen numerischen Operationen beim Algorithmus EAS-1 gegenüber EAS-2 haben. Daß die Ursache nicht in der Inversion der Matrix  $D^e$  zu sehen ist, wird zum einen dadurch bestätigt, daß eine Multiplikation der Inversen mit der ursprünglichen Matrix wieder die Einheitsmatrix ergibt und zum anderen durch die Tatsache, daß die mit  $D^{e-1}$  berechneten Tangentensteifigkeitsmatrizen und Elementresiduen zu guter Konvergenz und zu korrekten Ergebnissen im Zusammenhang mit dem Algorithmus EAS-2 führen. Ein Vermeiden dieser Inversion durch wiederholte Gleichungslösung (im Sinne der Vorgehensweise bei der Aufdatierung der EAS-Parameter bei Algorithmus EAS-2) führt auch bei  $\nu = 0.499$  zu den gleichen Ergebnissen wie bei Algorithmus EAS-2.

Wird in Algorithmus EAS-1 die Berechnung der EAS-Parameter, wie aus Gleichung (6.4) ersichtlich, zuerst durch Auswertung des Klammerterms und anschließender Multiplikation vorgenommen, so wird damit für das Kragarm Beispiel ein zu Algorithmus EAS-2 äquivalentes Verhalten erreicht.

Allerdings ist das nicht immer der Fall: Bei der Untersuchung der Biegung einer dünnen quadratischen Platte (siehe [32]) kann mit dem verbesserten Algorithmus EAS-1 bereits im ersten Lastschritt keine Konvergenz mehr erzielt werden. Auch mit dem Algorithmus EAS-2 wird im fünften Lastschritt keine Konvergenz mehr erreicht, außer zur Lösung von (6.7) wird ein Algorithmus mit numerischer Pivotsuche eingesetzt, der sinnvollerweise ebenso bei der Invertierung von  $D^e$  für die Gleichungen (6.2) und (6.3) genutzt werden kann. Dann kann das vollständige Last-Verformungsverhalten auch mit EAS Elementen berechnet werden.

Hieraus kann geschlossen werden, daß die Kondensation der EAS-Parameter numerisch äußerst sensitiv ist. Um die globale Konvergenz nicht negativ zu beeinflussen, ist es unerläßlich, die Kondensation mit höchster Sorgfalt zu implementieren um numerische Ungenauigkeiten zu vermeiden. Dann liefert die EAS-Methode auch für "Solid Shell" Elemente effiziente Ergebnisse.

## 6.2 Verteilte Berechnung des Skalarproduktes

In Abschnitt 5.8.2 wurde die parallele Berechnung des Skalarproduktes zweier Vektoren beschrieben. Dabei wurde angestrebt, daß einer der beiden Vektoren *globalen*, der andere hingegen *lokalen* Charakter besitzt, da dann das Skalarprodukt besonders effizient parallel berechenbar ist.

Bei nichtlinearen parallelen Finiten Element Berechnungen wurde nun beobachtet, daß kurz vor der Konvergenz des Newton-Raphson Verfahrens, also bei der Lösung des letzten linearen Gleichungssystems, keine Konvergenz des iterativen Löser erzielt wurde. Dies steht im Widerspruch mit dem Verhalten des äquivalenten sequentiellen Verfahrens und ist unabhängig von der gewählten Vorkonditionierung. Die auffallende Eigenschaft der "letzten Lösung" ist die niedrige Norm der rechten Seite. D.h., die Norm der Ungleichgewichtskräfte ist sehr klein, da die Gleichgewichtsbeziehung nahezu erfüllt ist. In diesem Falle wird durch das Abbruchkriterium (4.94) eine große Genauigkeit der zu berechnenden Lösung verlangt, da das Residuum durch die Norm der rechten Seite dividiert wird. Numerische Einflüsse machen sich also hier besonders stark bemerkbar.

Da unter Verwendung der Jacobi-Vorkonditionierung das sequentielle und das parallele iterative Verfahren identisch ablaufen müßte, die parallele Variante jedoch im Gegensatz zur sequentiellen Berechnung nicht konvergiert, ist die Ursache in der parallelen Implementierung zu suchen. Bei der sequentiellen Berechnung des Skalarproduktes werden nacheinander die entsprechenden Komponenten der Vektoren miteinander multipliziert und anschließend addiert. Bei der in Abschnitt 5.8.2 beschriebenen parallelen Berechnung werden an den überlappenden Komponenten jeweils nur Teile miteinander multipliziert und anschließend in einem globalen Kommunikationsschritt aufsummiert. Die numerische Berechnung des parallelen Skalarprodukt ist also gegenüber der sequentiellen Berechnung geändert. Daher sind auch andere numerische Rundungsfehler zu erwarten. Dies kann, wie z.B. in [88] geschildert, gravierenden Einfluß auf das numerisch berechnete Ergebnis haben.

Die zur sequentiellen Berechnung äquivalente Vorgehensweise für die parallele Ermittlung des Skalarproduktes läßt sich dadurch erreichen, daß die überlappenden Vektorkomponenten zuerst auf einem Prozessor akkumuliert werden, der dann anschließend den Beitrag dieser Komponente berechnet und dem Skalarprodukt hinzufügt. Damit konvergiert das parallele iterative Verfahren in derselben Anzahl von Iterationen (bei Jacobi-Vorkonditionierung) wie das sequentielle Verfahren.

Die in Abschnitt 5.10.2 gezeigten Ergebnisse werden mit der hier geschilderten Vorgehensweise erzielt, da andernfalls im ersten Lastschritt keine Konvergenz erreicht werden konnte und als Folge keine parallele Bearbeitung möglich gewesen wäre.

## 7 Stabilitätsuntersuchungen mit Finiten Elementen

In Abschnitt 2 wurden Vorgehensweisen zur numerischen Berechnung des Gleichgewichts von Kontinua angegeben. Die Kenntnis einer Gleichgewichtslage eines mechanischen Systems erlaubt aber keine Aussage über deren physikalische Realisierbarkeit. Dieses Problem tritt auf bei Nichteindeutigkeit, d.h. dann, wenn bei einer Belastungssituation mehrere Gleichgewichtslagen möglich sind. In der Technik werden solche Probleme Verzweigungs- oder Durchschlagprobleme genannt. Dieser Sachverhalt kann auch auf Bewegungen erweitert werden. Über die Realisierbarkeit einer Gleichgewichtslage entscheidet deren Stabilität. In der klassischen Vorgehensweise wird hierbei das Verhalten eines mechanischen Systems bezüglich Störungen untersucht. Im einfachsten Fall sind dies momentane Störungen in den Koordinaten und Geschwindigkeiten. Handelt es sich um infinitesimale Störungen, dann spricht man von der Methode der kleinen Schwingungen. Ein System in einer stabilen Gleichgewichtslage reagiert unempfindlich auf solche Störungen, bei einer instabilen Gleichgewichtslage hat selbst eine kleine Störung große Auswirkungen auf das System. Damit kann eine Klassifizierung von Gleichgewichtslagen erfolgen. Der Nachteil dieses Stabilitätsbegriffes liegt aber gerade in der Tatsache, daß dabei mit infinitesimal kleinen Störungen gearbeitet wird und nicht mit realen, endlichen Störungen. Zudem wäre es wünschenswert, die Stabilität zu quantifizieren und so zu Aussagen über die Sensitivität zu gelangen. Mit dem Begriff der Sensitivität wird die Empfindlichkeit einer stabilen Gleichgewichtslage eines mechanischen Systems gegenüber Störungen bezeichnet.

In den folgenden Abschnitten soll zunächst der klassische Stabilitätsbegriff eingeführt werden. Ein wesentlicher Aspekt ist hierbei die Auswertung der Stabilität mittels der Methode der Finiten Elemente. Anschließend wird der Versuch unternommen, die Sensitivität einer stabilen Gleichgewichtslage zu bestimmen. Außerdem soll ein Vorgehen vorgestellt werden, mit dem sich systematisch benachbarte Gleichgewichtslagen ermitteln lassen, die ansonsten nicht durch Pfadverfolgungsalgorithmen erreichbar sind.

### 7.1 Stabilitätsbegriff nach Ljapunov

In [95] hat LJAPUNOV die in der Mechanik und auch allgemein in den Wissenschaften heute nach wie vor aktuelle Definition des Begriffes der Stabilität geprägt. LJAPUNOV betrachtet eine bekannte spezielle Bewegung und betrachtet sie als ungestört. Diese ungestörte Bewegung wird nun mit einer gestörten Bewegung verglichen. Dazu wird zu einem Zeitpunkt  $t_0$  eine Störung auf die Koordinaten und/oder auf die Geschwindigkeiten eingebracht. Die sich dann für entsprechend kleine Störungen einstellende Bewegung wird als gestörte Nachbarbewegung bezeichnet. Die ungestörte Bewegung gilt als stabil, wenn der Abstand der gestörten zur ungestörten Bewegung für alle Zeiten  $t > t_0$  kleiner als ein vorgegebener positiver Wert ist, sofern die Störung nur genügend klein gewählt wird.

Eine formalere Darstellung wird angelehnt an die Schreibweise in [59, 100] bzw. [58]. Eine

Lösung des Differentialgleichungssystems 1. Ordnung

$$\dot{\mathbf{x}} = f(\mathbf{x}, t), \quad t > t_0 \quad (7.1)$$

wird eindeutig durch ihre Anfangsbedingungen festgelegt und im Folgenden als  $\mathbf{x}(\mathbf{x}_0, t_0; t)$  bezeichnet. Die Existenz dieser Lösungen  $\mathbf{x}$  und ihre Eindeutigkeit in dem jeweils betrachteten Gebiet wird vorausgesetzt. Dabei ist auch die triviale Lösung  $\mathbf{x}(\mathbf{x}_0, t_0; t) = \mathbf{0}$ , also Gleichgewicht möglich. Sei  $\mathbf{x}(\mathbf{a}, t_0; t)$  eine auf Stabilität zu untersuchende Lösung. Sie wird als stabil bezeichnet, falls  $\|\mathbf{x}(\mathbf{x}_0, t_0; t) - \mathbf{x}(\mathbf{a}, t_0; t)\|$  für alle Zeiten kleiner als ein beliebiges kleines vorgegebenes  $\varepsilon$  bleibt, sofern man nur  $\mathbf{x}_0$  genügend nahe zu  $\mathbf{a}$  wählt.

Die Bewegung  $\mathbf{x}(\mathbf{a}, t_0; t)$  ist stabil, wenn zu jedem  $\varepsilon > 0$  mit  $\varepsilon \rightarrow 0$  ein  $\eta(\varepsilon) > 0$  existiert, sodaß

$$\|\mathbf{x}(\mathbf{x}_0, t_0; t) - \mathbf{x}(\mathbf{a}, t_0; t)\| < \varepsilon \quad \text{für } t > t_0 \quad (7.2)$$

gilt für

$$\|\mathbf{x}_0 - \mathbf{a}\| < \eta(\varepsilon). \quad (7.3)$$

Dabei bezeichnet  $\|\dots\|$  eine geeignete Vektornorm. Eine Bewegung, die nicht stabil ist, ist instabil.

Als asymptotisch stabil wird die Bewegung  $\mathbf{x}(\mathbf{a}, t_0; t)$  bezeichnet, wenn

$$\lim_{t \rightarrow \infty} \|\mathbf{x}(\mathbf{x}_0, t_0; t) - \mathbf{x}(\mathbf{a}, t_0; t)\| = 0 \quad (7.4)$$

gilt, die gestörte Bewegung also wieder in die ungestörte Bewegung übergeht (z.B. aufgrund von Dämpfung).

### 7.1.1 1. Methode nach Ljapunov

LJAPUNOV unterscheidet zwei Methoden zur Lösung des Stabilitätsproblems. Die 1. Methode beruht auf einer unmittelbaren Untersuchung der gestörten Bewegung – es müssen also Lösungen der entsprechenden Differentialgleichungen gefunden werden [97]. Oft wird hierbei eine Reihenlösung durchgeführt und höhere Terme vernachlässigt, die Stabilität dann anhand des linearisierten Problems diskutiert. Dies führt allerdings nicht immer auf eine korrekte Stabilitätsentscheidung, wie sich anhand einfacher Beispiele zeigen läßt, siehe [97].

### 7.1.2 2. oder direkte Methode nach Ljapunov

Bei der 2. Methode hingegen sucht man Funktionen  $V$ , sogenannte ‘‘Ljapunov-Funktionen’’, in Abhängigkeit von  $\mathbf{x}$  und  $t$  mit bestimmten Eigenschaften, die im Folgenden angegeben

werden. Dadurch kann auf die Stabilität geschlossen werden, ohne die Differentialgleichungen der gestörten Bewegungen zu lösen. Zur Untersuchung der Stabilität einer Bewegung  $\mathbf{x}(\mathbf{a}, t_0; t)$  ist es oft zweckmäßig, eine Koordinatentransformation

$$\mathbf{z}(t) = \mathbf{x}(\mathbf{x}_0, t_0; t) - \mathbf{x}(\mathbf{a}, t_0; t) \quad (7.5)$$

durchzuführen. Mit der Transformation (7.5) liefert Gleichung (7.1)

$$\dot{\mathbf{z}} = f(\mathbf{x}(\mathbf{x}_0, t_0; t)) - f(\mathbf{x}(\mathbf{a}, t_0; t)) = f(\mathbf{z} + \mathbf{x}(\mathbf{a}, t_0; t)) - f(\mathbf{x}(\mathbf{a}, t_0; t)), \quad (7.6)$$

d.h. eine Beziehung der Form

$$\dot{\mathbf{z}} = g(\mathbf{z}, t). \quad (7.7)$$

Die Untersuchung der Stabilität der Lösung  $\mathbf{x}(\mathbf{a}, t_0; t)$  ist somit gleichwertig zur Untersuchung der Stabilität der trivialen Lösung  $\mathbf{z}(t) = \mathbf{0}$ .

Der erste Stabilitätssatz für die Stabilität einer trivialen Lösung  $\mathbf{z} = \mathbf{0}$  von LJAPUNOV läßt sich folgendermaßen formulieren [58]:

*Wenn man zu (7.1) eine positiv definite Funktion  $V(\mathbf{z}, t)$  finden kann, sodaß*

$$\dot{V}(\mathbf{z}, t) = \frac{\partial V}{\partial t} + \text{grad } V \cdot f(\mathbf{z}, t)$$

*negativ semidefinit ist oder identisch verschwindet, dann ist die triviale Lösung  $\mathbf{z} = \mathbf{0}$  von (7.1) stabil.*

Ein Beweis für diesen Satz findet man z.B. in [97].

### 7.1.3 Sonderfall: Gleichgewichtslagen

Mit dem ersten Stabilitätssatz kann für konservative mechanische Systeme auch der Satz von LAGRANGE-DIRICHLET hergeleitet werden [58], wenn die Summe aus potentieller und kinetischer Energie als Ljapunov-Funktion verwendet wird. Dieser besagt:

*Besitzt die potentielle Energie an der Stelle  $\mathbf{x} = \mathbf{0}$  ein Minimum, so ist die Ruhelage stabil.*

Diese Aussage wird auch oft als das Prinzip vom Minimum des Gesamtpotentials bezeichnet.

## 7.2 Behandlung von Stabilitätsproblemen mit der Methode der Finiten Elementen

Die mittels der Methode der Finiten Elemente getroffenen Stabilitätsaussagen bezüglich statischer Gleichgewichtslagen, die im Folgenden kurz beschrieben werden, beruhen auf

dem Satz von LAGRANGE-DIRICHLET. Die gefundene Gleichgewichtslage  $\mathbf{d}$  des diskretisierten Systems erfüllt zunächst Gleichung (2.58), die der algebraischen Darstellung des Prinzips vom stationären Wert des Potentials (2.39) entspricht. Zu untersuchen bleibt, ob die potentielle Energie an dieser Gleichgewichtslage ein Minimum besitzt oder nicht. Dazu wird die 2. Variation des Potentials  $\delta^2\Pi$  benötigt. Gleichung (2.58) muß also nach den Verschiebungen  $\mathbf{d}$  differenziert werden. Im Rahmen des Newton-Raphson Verfahrens wurde dies bereits durchgeführt, der Term  $D\mathbf{G}(\bar{\mathbf{d}})$  ist die tangentielle Steifigkeitsmatrix  $\mathbf{K}_T$ . Die Untersuchung der gefundenen Gleichgewichtslage bezüglich Stabilität reduziert sich also auf die Untersuchung der Definitheit der tangentiellen Steifigkeitsmatrix. Ist Letztere positiv definit, ist die Gleichgewichtslage stabil, andernfalls ist sie instabil. Dies kann auch anschaulich interpretiert werden: Besitzt das mechanische System in der Gleichgewichtslage ausschließlich positive Eigenfrequenzen, so ist es schwingungsfähig. Bei einer kleinen Störung wird es also kleine Schwingungen um die Gleichgewichtslage ausführen. An der Stabilitätsgrenze, wenn mindestens ein Eigenwert Null ist, wird eine Translationsbewegung möglich, das System wird kinematisch. Eine kleine Störung kann also das unbeschränkte Anwachsen der Verschiebungen zur Folge haben. Bei mindestens einem negativen Eigenwert wachsen die Verschiebungen unbeschränkt an.

In den folgenden Abschnitten werden hier die bekannten Kriterien zur Feststellung der Definitheit, d.h. zur Stabilitätsuntersuchung mit Finiten Elementen angegeben.

### 7.2.1 Trägheit der tangentiellen Steifigkeitsmatrix

Wie bereits in Abschnitt 4.1 erwähnt, kann die Frage nach der Definitheit bei Verwendung eines direkten Gleichungslösers, bei der eine Zerlegung der Form  $\mathbf{K}_T = \mathbf{L}\mathbf{D}\mathbf{L}^T$  vorgenommen wird, direkt durch Betrachtung der Vorzeichen der Elemente der Diagonalmatrix  $\mathbf{D}$  beantwortet werden. Dies beruht auf dem sogenannten Trägheitssatz von Sylvester. Unter der Trägheit einer Matrix versteht man das Zahlentripel  $(m, z, p)$ , wobei  $m$  die Anzahl der Eigenwerte kleiner Null,  $z$  die Anzahl der Eigenwerte gleich Null und  $p$  die Anzahl der Eigenwerte größer Null ist. Der Begriff der Trägheit einer Matrix steht in keinerlei Zusammenhang zur Trägheit eines mechanischen Systems. Der Trägheitssatz von Sylvester besagt nun, daß die Trägheit (*inertia*) der symmetrischen Matrix  $\mathbf{A}$  gleich der Trägheit von  $\mathbf{X}^T\mathbf{A}\mathbf{X}$  ist, wenn  $\mathbf{X}$  nichtsingulär ist. Stellt man nun die Eigenwerte der tangentiellen Steifigkeitsmatrix  $\mathbf{K}_T$  als Diagonalmatrix  $\mathbf{\Lambda}$  dar und ordnet die dazugehörigen Eigenvektoren in der Matrix  $\mathbf{Q}$  an, so ergibt sich:

$$\begin{aligned} \text{inert}(\mathbf{\Lambda}) &= \text{inert}(\mathbf{Q}^T\mathbf{K}_T\mathbf{Q}) = \text{inert}(\mathbf{Q}^T\mathbf{L}\mathbf{D}\mathbf{L}^T\mathbf{Q}) = \text{inert}(\mathbf{X}^T\mathbf{D}\mathbf{X}) \\ &= \text{inert}(\mathbf{D}). \end{aligned} \tag{7.8}$$

Für nichtsinguläre Matrizen  $\mathbf{K}_T$  ist auch  $\mathbf{Q}^T\mathbf{L}$  nichtsingulär. Die Trägheit der Matrix der Eigenwerte  $\mathbf{\Lambda}$  ist gleich der Trägheit der Diagonalmatrix  $\mathbf{D}$ , und damit stimmt auch die Anzahl der negativen Eigenwerte überein. Da die Eigenwerte einer Diagonalmatrix identisch mit den Einträgen dieser Matrix sind, kann an der Anzahl der negativen Einträge in  $\mathbf{D}$  die Anzahl der negativen Eigenwerte von  $\mathbf{K}_T$  abgelesen werden.

Erfolgt die Lösung der linearen Gleichungssysteme iterativ, so kann diese Aussage nicht mehr so einfach getroffen werden. Unter Verwendung des Lanczos Algorithmus findet eine Transformation auf die Tridiagonalmatrix  $\mathbf{T}$  statt (4.60), bei der die Trägheit erhalten bleibt. Bei der Faktorisierung von  $\mathbf{T}$  erhält man wie beim direkten Löser dann die gewünschte Information. Dies gilt auch für den vorkonditionierten Algorithmus, da der Trägheitssatz ebenfalls auf die Transformation der Matrix in (4.85) angewendet werden kann. Allerdings ist erst nach  $n$  Iterationen sichergestellt, daß alle negativen Eigenwerte erkannt wurden. Bei allen durchgeführten Berechnungen wurde aber beobachtet, daß die zur Lösung der linearen Gleichungssysteme erforderliche Anzahl von Iterationen genügt, um etwaige negative Eigenwerte zu erkennen.

### 7.2.2 Determinantenkriterium

Ein weiteres Kriterium, dessen Auswertung ebenfalls insbesondere bei Einsatz eines direkten Lösungsverfahrens wenig aufwendig ist, stellt das sogenannte Determinantenkriterium dar. Die Determinante der symmetrischen tangentiellen Steifigkeitsmatrix kann bei einer  $\mathbf{LDL}^T$ -Zerlegung sehr einfach bestimmt werden durch

$$\det(\mathbf{K}_T) = \det(\mathbf{LDL}^T) = \det(\mathbf{L}) \det(\mathbf{D}) \det(\mathbf{L}^T) = \det(\mathbf{D}), \quad (7.9)$$

da die Determinante der unteren Dreiecksmatrix  $\mathbf{L}$  Eins ist. Die Determinante der Diagonalmatrix läßt sich nun einfach aus dem Produkt der Diagonalterme

$$\det(\mathbf{D}) = \prod_{i=1}^n d_i \quad (7.10)$$

ermitteln.

Da die Eigenwerte der tangentiellen Steifigkeitsmatrix sich als Lösung der charakteristischen Gleichung

$$\det(\mathbf{K}_T - \omega^2 \mathbf{I}) = 0 \quad (7.11)$$

ergeben, wird nach (7.11) im Falle der Stabilitätsgrenze, der Grenze der positiven Definitheit, mindestens ein Eigenwert gleich Null, d.h. es gilt

$$\det(\mathbf{K}_T) = 0. \quad (7.12)$$

Damit ist also der Wechsel von stabilen zu instabilen Zuständen einfach erkennbar. Während  $\det(\mathbf{K}_T) < 0$  eine hinreichende Bedingung für Instabilität darstellt, ist bei der Interpretation von  $\det(\mathbf{K}_T) > 0$  bezüglich Stabilität Vorsicht geboten, sofern negative Eigenwerte bzw. Diagonalterme vorliegen. Hier können sich zwei bzw.  $2m$  negative Einträge aufheben. Folglich ist eine positive Determinante lediglich eine notwendige Bedingung

für Stabilität. Eine Überprüfung der Trägheit der tangentiellen Steifigkeit, siehe Abschnitt 7.2.1, muß bei vorliegen negativer Eigenwerte zusätzlich erfolgen.

Dieses Kriterium ist auf iterative Löser nicht ohne zusätzlichen Rechenaufwand übertragbar.

Zwischen diesem hier erläuterten Determinantenkriterium und dem sogenannten HURWITZ Kriterium besteht nur insofern ein Zusammenhang, als daß in beiden Fällen das charakteristische Polynom untersucht wird. Beim HURWITZ Kriterium geschieht dies durch die Betrachtung von  $n$  Untermatrizen, siehe z.B. [58]. Es wird dadurch ein notwendiges und hinreichendes Stabilitätskriterium angegeben, wohingegen hier ein Kriterium zur Ermittlung des Systemwechsels von stabil nach instabil vorliegt.

### 7.2.3 Steifigkeitswert nach Bergan

Der sogenannte Steifigkeitswert nach BERGAN (*current stiffness parameter*) [10] besitzt eine ähnliche Aussagekraft wie das Determinantenkriterium. Es ist damit möglich, stabile von instabilen Bereichen des Lösungspfades zu unterscheiden. Die Definition nach [10] lautet:

$$S_P = \frac{\mathbf{d}_{0,\lambda}^T \mathbf{p}}{\mathbf{d}_{,\lambda} \mathbf{p}}, \quad (7.13)$$

wobei  $(\cdot)_{,\lambda}$  die partielle Ableitung  $\frac{\partial}{\partial \lambda}(\cdot)$  bezeichnet. Im Rahmen einer inkrementellen Lösungsprozedur stellt sich (7.13) als

$$S_P = \left( \frac{\Delta \lambda_i}{\Delta \lambda_1} \right)^2 \frac{\Delta \mathbf{d}_1^T \mathbf{K}_1 \Delta \mathbf{d}_1}{\Delta \mathbf{d}_i^T \mathbf{K}_i \Delta \mathbf{d}_i} \quad (7.14)$$

dar. Bei Berechnungsbeginn beträgt  $S_P = 1$ , erhöht sich für versteifende Systeme und verringert sich bei weicher werdenden Systemen. An der Stabilitätsgrenze ergibt sich für  $S_P$  der Wert Null. Der Vorteil des Steifigkeitswertes nach BERGAN ist, daß die Vektoren  $\mathbf{K}_i \Delta \mathbf{d}_i$  als jeweilige rechte Seiten vorliegen, d.h. der Steifigkeitswert mit jeweils einem Skalarprodukt ermittelbar ist. Die Ermittlung ist bei jedem Lösertyp gleich aufwendig.

### 7.2.4 Eigenwertanalyse

Die naheliegendste, allerdings aufwendigste Möglichkeit, die Definitheit der Steifigkeitsmatrix und damit die Stabilität einer Gleichgewichtslage zu überprüfen, besteht darin, alle Eigenwerte zu bestimmen, d.h. das Eigenwertproblem

$$(\mathbf{K}_T - \omega^2 \mathbf{I}) \Phi = \mathbf{0} \quad (7.15)$$

zu lösen. Für Systeme mit großen Anzahlen von Freiheitsgraden stellt dies aber einen bei weitem zu hohen Rechenaufwand dar, da diese Eigenwertanalyse bei einer inkrementellen Verfolgung des Lösungspfades an jeder Gleichgewichtslage durchgeführt werden muß.

Interessiert hingegen nur der betragskleinste Eigenwert  $\omega_1$ , so kann dieser mittels einer von Mises Vektoriteration [14] effizient bestimmt werden, sofern ein direkter Löser verwendet wird und daher die faktorisierte Steifigkeitsmatrix verfügbar ist. Für iterative Löser ist mit der von Mises Vektoriteration keine effiziente Bestimmung des kleinsten Eigenwertes möglich, eine Eigenwertermittlung unter Verwendung des Lanczos-Verfahrens erscheint hierfür sinnvoller, wurde jedoch im Rahmen dieser Arbeit nicht realisiert.

Bei einer nichtlinearen Pfadverfolgung kann dann die Entwicklung des betragskleinsten Eigenwertes untersucht werden. Eine Verringerung des betragskleinsten Eigenwertes zeigt die Annäherung an die Stabilitätsgrenze an. Zu Beachten ist aber, daß Instabilität nicht notwendigerweise angezeigt wird, da der betragskleinste Eigenwert positiv sein kann, und dennoch ein negativer Eigenwert mit größerem Betrag existiert.

Ein wesentlicher Vorteil bei der Lösung des Eigenwertproblems liegt darin, daß auch die Eigenvektoren vorliegen, die zusätzliche Information über das Strukturverhalten und mögliche Beulformen liefern.

### 7.3 Prognose, Bestimmung und Klassifizierung von singulären Punkten

Meist ist man nicht nur an der Stabilitätsinformation über eine gefundene Gleichgewichtslage auf einem Belastungspfad interessiert, sondern möchte das Verhalten an der Stabilitätsgrenze, d.h. den Übergang von stabilen zu instabilen Gleichgewichtszuständen und die Stabilitätsgrenze selbst – die sogenannte kritische Last – möglichst genau bestimmen oder sie zumindest abschätzen. Gleiches gilt für Verzweigungspunkte auf dem Lösungsast. Das unterschiedliche Verhalten am Übergang von einer stabilen zu einer instabilen Lage wird auch zu einer Klassifizierung im Rahmen einer nichtlinearen Berechnung des Last-Verformungspfades benötigt.

Die Berechnung solcher ausgezeichneten Punkte, d.h. von Maxima und Minima bzw. von Verzweigungspunkten entlang von Last-Verformungskurven wird als statische Stabilitätsuntersuchung bezeichnet.

#### 7.3.1 Bisektionsverfahren

Die einfachste Möglichkeit singuläre Punkte zwischen zwei Laststufen zu bestimmen, liegt in einer Bisektion, d.h. einer Intervallschachtelung. Zeigt eines der oben beschriebenen Kriterien einen Wechsel des Stabilitätsverhaltens an, so liegt (mindestens) eine Stabilitätsgrenze zwischen den beiden zuletzt bestimmten Gleichgewichtslagen. Durch wiederholte erneute Durchführung des letzten Lastschrittes mit reduzierter Schrittweite (Last- bzw. Verschiebungsinkremente) kann eine Iteration zum gesuchten singulären Punkt erfolgen.

Als Ausgangspunkt dient entweder die Gleichgewichtslage diesseits des singulären Punktes (einseitige Bisektion) oder auch die Gleichgewichtslage jenseits des singulären Punktes (beidseitige Bisektion). Handelt es sich beispielsweise um einen Übergang von einem stabilen zu einem instabilen Gleichgewichtszustand, so wird bei der einseitigen Bisektion ausgehend von der stabilen Gleichgewichtslage der Steuerparameter so lange verringert, bis die dann bestimmte Gleichgewichtslage erneut stabil ist. Mit dieser neuen stabilen Gleichgewichtslage wird der Vorgang wiederholt, bis der singuläre Punkt genügend genau bestimmt ist. Bei der beidseitigen Bisektion wird auch von der instabilen Gleichgewichtslage ausgehend der Steuerparameter reduziert und auf dem Last-Verformungspfad zurückgerechnet. Ist die dann gefundene Gleichgewichtslage stabil, wird mit nochmals reduzierter Schrittweite wieder vorwärts gerechnet, andernfalls weiter rückwärts. So findet eine Eingrenzung des singulären Punktes von beiden Seiten statt.

### 7.3.2 Lineare Eigenwertuntersuchungen

Wird die tangentielle Steifigkeitsmatrix  $\mathbf{K}_T$  an einer beliebig gewählten Laststufe in eine Reihe

$$\mathbf{K}_T = \xi^0 \mathbf{K}_0 + \xi^1 \mathbf{K}_1 + \xi^2 \mathbf{K}_2 + \dots \quad (7.16)$$

bezüglich eines Parameters  $\xi$ , z.B. des Lastparameters, entwickelt, so ergibt sich aus der Beziehung (7.12) bei Erreichen eines singulären Punktes

$$\det(\mathbf{K}_T) = \det(\mathbf{K}_0 + \xi \mathbf{K}_1) = 0, \quad (7.17)$$

sofern die Reihenentwicklung (7.16) nach dem linearen Glied abgebrochen wird. Die Erfüllung von Gleichung (7.17) ist die notwendige Bedingung für nichttriviale Lösungen des linearen allgemeinen Eigenwertproblems

$$(\mathbf{K}_0 + \xi \mathbf{K}_1)\phi = \mathbf{0}. \quad (7.18)$$

Für die Wahl der Matrizen  $\mathbf{K}_i$ ,  $i = 0, 1, \dots$  gibt es sehr unterschiedliche Vorschläge [113, 13], die im Rahmen der vorliegenden Arbeit nicht weiter diskutiert werden sollen. Für Finite Element Untersuchungen insbesondere großer Probleme ist die Wahl

$$\mathbf{K}_0 = \mathbf{K}_L \quad \text{und} \quad \mathbf{K}_1 = \mathbf{K}_{NL} \quad (7.19)$$

am sinnvollsten [31, 85, 119, 80], wobei  $\mathbf{K}_{NL}$  besonders einfach mittels

$$\mathbf{K}_{NL} = \mathbf{K}_T - \mathbf{K}_L \quad (7.20)$$

bestimmt werden kann. In Gleichung (7.20) ist  $\mathbf{K}_L$  die Steifigkeitsmatrix des linearen Problems. Unter der Annahme, daß die Steifigkeitsmatrix linear von den Verschiebungen abhängt, wird mit Gleichung (7.18) der Lastparameter  $\xi$  so bestimmt, daß

$$\mathbf{K} = \mathbf{K}_L + \xi \mathbf{K}_{NL} \quad (7.21)$$

singulär wird.

Wird das Eigenwertproblem für eine Last unterhalb des kritischen Lastniveaus ausgewertet, erhält man einen Näherungswert für die kritische Last  $\lambda_{kr} = \lambda\xi$ , der im Falle eines schwach nichtlinearen Vorbeulverhaltens der untersuchten mechanischen Struktur durchaus zufriedenstellend, d.h. nahe an der wirklichen Lösung sein kann. Wertet man Gleichung (7.18) hingegen begleitend zur inkrementellen Lösung aus, verbessert sich die Näherung des kritischen Lastfaktors  $\lambda_{kr} = \lambda\xi$  bei Annäherung an den kritischen Punkt. Es sei angemerkt, daß obiges nur für richtungstreue Lasten gilt, da alle Lasten durch den skalaren Parameter  $\lambda$  gleichmäßig gesteigert werden. Direkt am kritischen Punkt nimmt  $\xi$  dann den Wert Eins an. Alternativ kann mit (7.20) auch das Problem

$$(\mathbf{K}_L + \xi \mathbf{K}_{NL})\phi = (\mathbf{K}_T + (\xi - 1)\mathbf{K}_{NL})\phi = (\mathbf{K}_T + \bar{\xi}\mathbf{K}_{NL})\phi = 0 \quad (7.22)$$

gelöst werden, wobei  $\bar{\xi}$  im singulären Punkt den Wert Null annimmt.

Trägt man  $\lambda\xi$  und den Lastparameter  $\lambda$  über einer charakteristischen Verschiebung (oder einer Verschiebungsnorm) auf, schneiden sich die beiden Kurven im singulären Punkt.

Die Lösung des allgemeinen linearen Eigenwertproblems (7.18) ist numerisch sehr aufwendig. Zur Reduktion dieses Aufwandes wird im Schrifttum, siehe z.B. [148], vorgeschlagen, anstelle von (7.18) das numerisch effizienter lösbare Problem (7.15) zu lösen. Dies geschieht unter der Annahme, daß die Eigenvektoren  $\Phi$  und  $\phi$  bei Annäherung an den kritischen Punkt annähernd gleich sind. Am kritischen Punkt selbst sind die beiden Eigenwertprobleme (7.18) und (7.15) identisch. Daraus läßt sich die Beziehung für den angenäherten Wert  $\tilde{\xi}$

$$\tilde{\xi} = \frac{\Phi^T \mathbf{K}_L \Phi}{\Phi^T \mathbf{K}_L \Phi - \omega^2 \Phi^T \Phi} \quad (7.23)$$

gewinnen.

### 7.3.3 Direkte Berechnung von Stabilitätspunkten

Von WRIGGERS ET AL. [155, 154] wurde ein Verfahren vorgeschlagen, bei dem als Nebenbedingung das Verschwinden des kleinsten Eigenwertes direkt in das Newton Verfahren eingebracht wird. Für  $\omega^2 = 0$  ergibt sich aus (7.15)

$$\mathbf{K}_T \Phi = \mathbf{0}, \quad (7.24)$$

sodaß sich mit einer zusätzlich erforderlichen Normierungsbedingung  $l(\Phi) = 0$  das erweiterte algebraische nichtlineare Gleichungssystem

$$\hat{\mathbf{G}}(\mathbf{d}, \lambda, \Phi) = \begin{Bmatrix} \mathbf{G}(\mathbf{d}, \lambda) \\ \mathbf{K}_T(\mathbf{d}, \lambda)\Phi \\ l(\Phi) \end{Bmatrix} = \mathbf{0} \quad (7.25)$$

ergibt. Dieses erweiterte System wird dann mit dem Newton Verfahren gelöst. Zwar ergeben sich  $2n + 1$  Unbekannte, durch eine geschickte Formulierung müssen aber je Newton Iteration (ähnlich zum Bogenlängenverfahren) nur zwei lineare Gleichungssysteme mit derselben Koeffizientenmatrix  $\mathbf{K}_T$  gelöst werden.

Als Variante dieses Verfahrens kann anstelle (7.24) Bedingung (7.12) verwendet werden, wodurch laut [111] ein größerer Konvergenzradius erreicht werden soll.

Damit ist eine elegante Möglichkeit der direkten Berechnung von singulären Punkten gegeben, ohne das vollständige Last-Verformungsverhalten bestimmen zu müssen. Allerdings ist dabei nicht gewährleistet, daß der maßgebende singuläre Punkt, d.h. derjenige mit dem niedrigsten Lastparameter, bestimmt wird. Insbesondere, wenn mehrere singuläre Punkte entlang des Last-Verformungspfades nahe beieinander liegen, kann die direkte Berechnung des singulären Punktes mit dem kleinsten Lastparameter  $\lambda$  nicht sichergestellt werden, siehe z.B. [136]. Ähnliches wurde in [80] bei der Berechnung geometrisch imperfekter Zylinder mit größerer Anzahl an Freiheitsgraden und vielen möglichen Beulformen in Kombination mit einer Berechnung des Last-Verformungspfades beobachtet, sodaß dort das zwar weniger elegante, dafür aber robustere Bisektionsverfahren angewandt wurde.

### 7.3.4 Klassifizierung von singulären Punkten

Die Lösung des algebraischen, nichtlinearen Gleichungssystems (2.58) beschreibt eine Kurve im  $(n + 1)$ -dimensionalen Raum. Dabei sind  $n$  die diskreten Knotenfreiheitsgrade. Der Lastparameter  $\lambda$  ergibt die Erweiterung um eine zusätzliche Dimension. Die Lösungskurve kann in natürlichen Koordinaten mit einer Bogenlänge  $s$  beschrieben werden. Die Ableitung von (2.58) nach dieser Bogenlänge besitzt die Gestalt:

$$\frac{d\mathbf{G}}{ds} = \frac{\partial\mathbf{G}}{\partial\mathbf{d}} \frac{d\mathbf{d}}{ds} + \frac{\partial\mathbf{G}}{\partial\lambda} \frac{d\lambda}{ds} = \mathbf{K}_T \frac{d\mathbf{d}}{ds} + \frac{\partial\mathbf{G}}{\partial\lambda} \frac{d\lambda}{ds} = \mathbf{0}. \quad (7.26)$$

An einem singulären Punkt gilt

$$\mathbf{K}_T \Phi_1 = \mathbf{0}, \quad (7.27)$$

wobei mit  $\Phi_1$  der zu  $\omega_1 = 0$  gehörende Eigenvektor bezeichnet wird. Um (7.26) zu erfüllen, muß also auch

$$\Phi_1^T \frac{\partial\mathbf{G}}{\partial\lambda} \frac{d\lambda}{ds} = \mathbf{0} \quad (7.28)$$

gelten. Die Multiplikation mit  $\Phi_1^T$  ist durch die Singularität von  $K_T$  begründet, siehe [120]. Gleichung (7.28) kann also durch

$$\Phi_1^T \frac{\partial \mathbf{G}}{\partial \lambda} = \mathbf{0} \quad \text{oder} \quad \frac{d\lambda}{ds} = 0 \quad (7.29)$$

erreicht werden. Im ersten Fall spricht man von einem Verzweigungspunkt, im Letzteren von einem Durchschlagpunkt. Anschaulich ist im zweiten Fall die Steigung der Tangente an die Lösungskurve bezüglich des Lastparameters Null, an diesem Punkt findet also keine Laststeigerung statt.

Unter Berücksichtigung von Gleichung (2.58) wird aus (7.29)<sub>1</sub> für einen Verzweigungspunkt

$$\Phi_1^T \frac{\partial \mathbf{G}}{\partial \lambda} = -\Phi_1^T \mathbf{p} = 0. \quad (7.30)$$

Ist (7.30) nicht erfüllt, aber  $\frac{d\lambda}{ds} = 0$ , gilt also

$$\Phi_1^T \mathbf{p} \neq 0, \quad (7.31)$$

handelt es sich um einen Durchschlagpunkt.

### 7.3.5 Behandlung von Verzweigungsproblemen

Ist ein Verzweigungspunkt als solcher identifiziert, ist man daran interessiert, neben dem natürlichen auch den abzweigenden Lösungspfad zu berechnen. A priori ist nicht bekannt, ob der natürliche oder der abzweigende Lösungspfad zur niedrigsten Nachbeullast führt, weswegen beide Pfade berechnet werden müssen.

Im Schrifttum hierzu behandelte Algorithmen [73, 70, 83, 116, 120] gehen auf die von KOITER entwickelte Anfangs-Nachbeul Theorie zurück [82]. Diese beruht auf einem Störungsansatz, der lehrbuchmäßig in [101] zu finden ist. Anwendungen auf die Elastostatik finden sich z.B. in [143]. Die hier gegebene Darstellung beruht im Wesentlichen auf [16]. Der Grundgedanke ist, daß in einem Bereich des Lastparameters  $\lambda$  neben dem natürlichen Lösungspfad  $\mathbf{d}_0$  ein weiterer Lösungspfad

$$\mathbf{d} = \mathbf{d}_0(\lambda) + \mathbf{v}(\lambda) \quad (7.32)$$

existiert, der den natürlichen Lösungspfad im kritischen Punkt schneidet, d.h.

$$\lim_{\lambda \rightarrow \lambda_{kr}} \mathbf{v}(\lambda) = \mathbf{0}. \quad (7.33)$$

Gleichung (7.32) stellt einen Störungsansatz in der Umgebung des kritischen Punktes dar, siehe Gleichung (7.33). Ziel ist nun die Bestimmung von  $\mathbf{v}(\lambda)$ . Da bekanntermaßen die zum kritischen Eigenwert gehörende Eigenform maßgebend in der Beulform, d.h. der Verschiebungsfigur auf dem abzweigenden Lösungspfad, enthalten ist, wird  $\mathbf{v}$  als

$$\mathbf{v} = \xi \Phi_1 + \tilde{\mathbf{v}} \quad (7.34)$$

geschrieben, wobei nun der Parameter  $\xi$  als Anteil der ersten Eigenform an der Beulform zu verstehen ist. Als Ausgangspunkt des weiteren Vorgehens dient der Satz vom stationären Wert des Potentials  $\delta\Pi = 0$ , der für die gesuchten Gleichgewichtslagen  $\mathbf{d}$  aus (7.32) für einen festen Lastwert gelten muß:

$$\delta\Pi(\mathbf{d}, \lambda) = \Pi_{,\mathbf{d}}(\mathbf{d}, \lambda) \delta\mathbf{d} = 0. \quad (7.35)$$

Im Folgenden wird die partielle Ableitung nach den Verschiebungen durch  $\frac{\partial}{\partial \mathbf{d}}(\cdot) = (\cdot)_{,\mathbf{d}}$  und die partielle Ableitung nach dem Lastparameter  $\lambda$  durch  $\frac{\partial}{\partial \lambda}(\cdot) = (\cdot)_{,\lambda}$  gekennzeichnet. In der Umgebung des kritischen Punktes liefert eine Taylor Entwicklung von (7.35)

$$\Pi_{,\mathbf{d}}(\mathbf{d}_0, \lambda) \delta\mathbf{d} + \Pi_{,dd}(\mathbf{d}_0, \lambda) \mathbf{v} \delta\mathbf{d} + \frac{1}{2} \Pi_{,ddd}(\mathbf{d}_0, \lambda) \mathbf{v}^2 \delta\mathbf{d} + \dots = 0, \quad (7.36)$$

wobei der erste Term aufgrund des Gleichgewichtes bei  $\mathbf{d}_0$  verschwindet. Anschließend kann eine weitere Taylor Entwicklung nach  $\lambda$  auf (7.36) um den kritischen Punkt angewendet werden:

$$\begin{aligned} & \left\{ \Pi_{,dd}^{kr} + (\lambda - \lambda_{kr}) \Pi_{,dd\lambda}^{kr} + \frac{1}{2} (\lambda - \lambda_{kr})^2 \Pi_{,dd\lambda\lambda}^{kr} + \dots \right\} \mathbf{v} \delta\mathbf{d} + \\ & \frac{1}{2} \left\{ \Pi_{,ddd}^{kr} + (\lambda - \lambda_{kr}) \Pi_{,ddd\lambda}^{kr} + \frac{1}{2} (\lambda - \lambda_{kr})^2 + \dots \right\} \mathbf{v}^2 \delta\mathbf{d} + \dots = 0. \end{aligned} \quad (7.37)$$

Als nächsten Schritt führt man für  $\lambda$  und  $\mathbf{d}$  eine Potenzreihenentwicklung ein:

$$\lambda = \lambda_{kr} + \xi \lambda_1 + \xi^2 \lambda_2 + \xi^3 \lambda_3 + \dots \quad (7.38)$$

$$\mathbf{d} = \mathbf{d}_0(\lambda) + \xi \mathbf{d}_1 + \xi^2 \mathbf{d}_2 + \xi^3 \mathbf{d}_3 + \dots \quad \text{bzw. mit (7.32)} \quad (7.39)$$

$$\mathbf{v} = \xi \mathbf{d}_1 + \xi^2 \mathbf{d}_2 + \xi^3 \mathbf{d}_3 + \dots \quad (7.40)$$

Die Gleichungen (7.38) bis (7.40) stellen einen Ansatz nach der Methode LINDSTEDT [101] gemäß der regulären Störungsrechnung dar. Nach den Voraussetzungen der Störungsrechnung müssen die Größen  $\lambda_{kr}$  und  $\mathbf{d}$  bekannt sein. Ebenso ist  $\mathbf{d}_1$  mit Gleichung (7.34) festgelegt. Bei dem Parameter  $\xi \ll 1$  handelt es sich im Sinne der Störungsrechnung um einen willkürlichen Größenordnungsparameter. Die resultierende Lösung gilt demnach nur in der unmittelbaren Umgebung des kritischen Punktes. Ebenso sei angemerkt, daß bei diesem Ansatz die Konvergenz nicht gesichert ist.

Setzt man nun Gleichungen (7.38) und (7.40) in Gleichung (7.37) ein und sortiert die entsprechenden Terme nach Potenzen von  $\xi$ , so ergibt sich

$$\begin{aligned} & \xi \left( \Pi_{,dd}^{kr} \mathbf{d}_1 \right) \delta \mathbf{d} + \\ & \xi^2 \left( \Pi_{,dd}^{kr} \mathbf{d}_2 + \lambda_1 \Pi_{,dd\lambda}^{kr} \mathbf{d}_1 + \frac{1}{2} \Pi_{,ddd}^{kr} \mathbf{d}_1^2 \right) \delta \mathbf{d} + \\ & \xi^3 \left( \Pi_{,dd}^{kr} \mathbf{d}_3 + \lambda_1 \Pi_{,dd\lambda}^{kr} \mathbf{d}_2 + \lambda_2 \Pi_{,dd\lambda}^{kr} \mathbf{d}_1 + \frac{1}{2} \lambda_1 \Pi_{,dd\lambda\lambda}^{kr} \mathbf{d}_1 + \dots \right) \delta \mathbf{d} + \dots = 0. \end{aligned} \quad (7.41)$$

Damit (7.41) erfüllt ist, müssen alle Klammerterme für die entsprechenden Potenzen von  $\xi$  verschwinden, woraus sich die  $\lambda_i$  bestimmen lassen. Setzt man  $\delta \mathbf{d} = \mathbf{d}_1$  und berücksichtigt  $\Pi_{,dd}^{kr} \mathbf{d}_2 \mathbf{d}_1 = \Pi_{,dd}^{kr} \mathbf{d}_1 \mathbf{d}_2 = 0$  (dies entspricht dem Eigenwertproblem (7.15) – die zweite Variation verschwindet im singulären Punkt) erhält man schließlich für die ersten beiden Terme  $\lambda_1$  und  $\lambda_2$

$$\lambda_1 = -\frac{1}{2} \frac{\Pi_{,ddd}^{kr} \mathbf{d}_1^3}{\Pi_{,dd\lambda}^{kr} \mathbf{d}_1^2} \quad \text{und} \quad (7.42)$$

$$\lambda_2 = -\frac{\frac{1}{6} \Pi_{,dddd}^{kr} \mathbf{d}_1^4 + \Pi_{,ddd}^{kr} \mathbf{d}_1^2 \mathbf{d}_2 + \lambda_1 (\Pi_{,dd\lambda}^{kr} \mathbf{d}_2 \mathbf{d}_1 + \frac{1}{2} \Pi_{,ddd\lambda}^{kr} \mathbf{d}_1^2) + \frac{1}{2} \lambda_1^2 \Pi_{,dd\lambda\lambda}^{kr} \mathbf{d}_1^3}{\Pi_{,dd\lambda}^{kr} \mathbf{d}_1^2}. \quad (7.43)$$

Mit diesen beiden Werten kann der Typ der Verzweigung bestimmt werden. In Abbildung 7.1 sind verschiedene Verzweigungstypen dargestellt; die Lösungen nach Gleichungen (7.42) und (7.43) gelten nur in der unmittelbaren Nachbarschaft des kritischen Punktes. Stabile Äste sind in Abbildung 7.1 durchgezogen, instabile gestrichelt dargestellt. Diese Information entstammt nicht dem in diesem Abschnitt behandelten Verfahren. Für  $\lambda_1 = 0$  ist die Verzweigung symmetrisch, für  $\lambda_1 \neq 0$  ist sie unsymmetrisch. Bei symmetrischen Verzweigungen zeigt  $\lambda_2 > 0$  eine stabile,  $\lambda_2 < 0$  eine instabile Verzweigung an. Allerdings setzt die Berechnung von  $\lambda_2$  die Kenntnis von  $\mathbf{d}_2$  voraus. Letzterer Vektor kann mit Hilfe von Gleichung (7.41) bestimmt werden, hierzu sei auf [16] verwiesen. Dann ist der Typ der Verzweigung und die Beulform in naher Umgebung des Verzweigungspunktes bekannt.

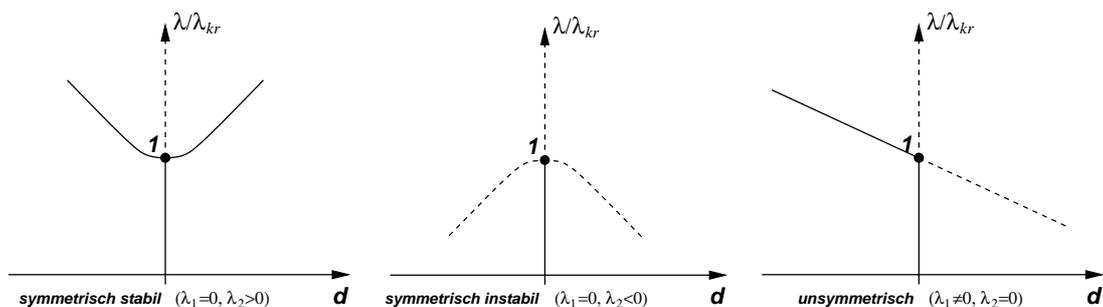


Abbildung 7.1: Klassifikation von Verzweigungen

Es bleibt jedoch anzumerken, daß die zur Bestimmung von  $\lambda_1$  und  $\lambda_2$  notwendigen höheren partiellen Ableitungen des Potentials nur mit hohem Aufwand bestimmbar sind.

In der "Berechnungspraxis" wird deshalb beim sogenannten *branch-switching*, also dem Wechseln der Lösungsäste an einem Verzweigungspunkt, meist nach einem einfacheren Schema vorgegangen als oben beschrieben, siehe z.B. [149, 85, 150]. Wird ein Verzweigen in einen anderen als den natürlichen Lösungspfad gewünscht, so wird zur Lösung am kritischen Punkt  $\mathbf{d}_{kr}$  ein Vielfaches der ersten Eigenform (Beulform)  $\xi \Phi_1$  addiert und als Startvektor für eine Gleichgewichtsiteration verwendet. Dabei wird für  $\xi$  ein kleiner Wert vorgegeben. Die erste Eigenform  $\Phi_1$  ist nach Gleichung (7.30) orthogonal zur Last  $\mathbf{p}$  und es ist deshalb im Verzweigungspunkt eine Verschiebung in diese Richtung ohne Energiezufuhr möglich. Dadurch sollte im Allgemeinen ein Lösungspunkt auf dem Sekundärast aniteriert werden. Zur Klassifizierung des Verzweigungspunktes wird in [150] vorgeschlagen, diesen Vorgang mit  $-\xi$  zu wiederholen. Dadurch erhält man zwei Lösungspunkte auf dem Sekundärast – auf beiden Seiten des kritischen Punktes – und es kann hiermit auf den Typ der Verzweigung geschlossen werden.

Treten bei einer zu untersuchenden mechanischen Struktur mehrere Verzweigungen mit jeweils unterschiedlicher Last auf<sup>10</sup>, deren Lösungsäste sich möglicherweise weiter verzweigen, ist es mit vertretbarem Aufwand bei mechanischen Systemen mit hohen Anzahlen von Freiheitsgraden nicht möglich, alle Lösungsäste zu verfolgen, um letztendlich denjenigen Ast zu ermitteln, der die geringste Nachbeullast aufweist. Darüberhinaus besteht auch die Gefahr, Verzweigungspunkte unbemerkt zu passieren, wenn nicht eine begleitende Eigenwertanalyse durchgeführt wird, da z.B. die Trägheit der Matrix und das Determinantenkriterium nur an einzelnen Punkten des Lösungspfades ausgewertet werden.

### 7.3.6 Imperfektionsempfindlichkeit

Ein reales System kann nie exakt so hergestellt werden wie geplant. Stets treten kleine Abweichungen vom idealen Zustand, sogenannte Imperfektionen, wie z.B. Abweichungen von der geometrischen Form oder Abweichungen vom idealen Lastzustand auf. Obwohl diese Abweichungen vom Idealzustand der Struktur klein sind, können sie ihr Verhalten jedoch stark beeinflussen [139]. In Abhängigkeit der Imperfektionen verändern sich charakteristische Systemgrößen wie z.B. Grenzlaster. Bei Verzweigungsproblemen verändert sich aber auch das Verzweigungsverhalten. Bei der Wahl entsprechender Imperfektionen verschwinden die Verzweigungspunkte.

Abbildung 7.2 zeigt die Diagramme aus Abbildung 7.1 bei der Anwesenheit von Imperfektionen, deren Amplituden durch den skalaren Wert  $\varepsilon$  bezeichnet werden. An einfachen mechanischen Systemen können sie mittels der Theorie der gestörten Verzweigung bestimmt werden, siehe z.B. [98, 144]. Die darin enthaltene Stabilitätsinformation entstammt der Kenntnis der Gesamtlösung der jeweils betrachteten Probleme.

Für die symmetrisch stabile und die symmetrisch instabile Verzweigung ist das Verhal-

<sup>10</sup>Mehrfache Nulleigenwerte werden im Rahmen der vorliegenden Arbeit nicht betrachtet.

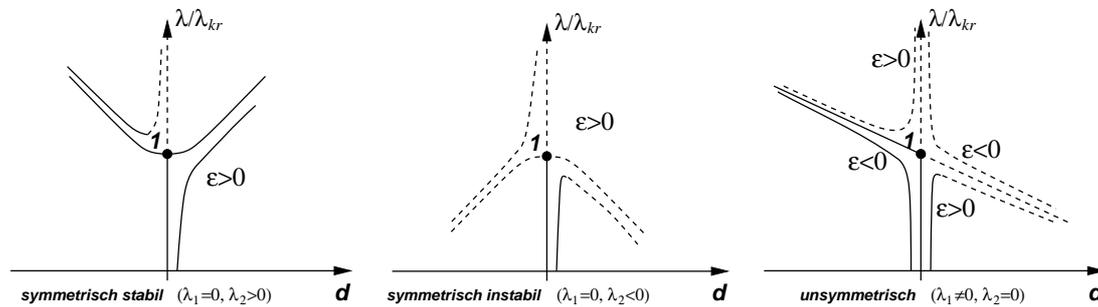


Abbildung 7.2: Einfluß der Imperfektion auf das Verzweigungsverhalten

ten nur für  $\varepsilon > 0$  dargestellt, die entsprechenden Diagramme für  $\varepsilon < 0$  ergeben sich durch Spiegelung an der Lastachse. Im dritten Diagramm sind beide Fälle dargestellt und die einzelnen Kurvenäste gekennzeichnet. Bei allen Verzweigungsdiagrammen verschwindet der Verzweigungspunkt und es entstehen zwei unabhängige Lösungsäste. Im Falle der symmetrisch stabilen Verzweigung ist der natürliche Lösungspfad stabil und oberhalb der Verzweigungslast der perfekten Struktur existiert ein weiterer Lösungsast. Bei der unsymmetrischen Verzweigung entsteht als Folge der Imperfektion eine Grenzlast, ab der der stabile Lösungsast in einen abfallenden instabilen Lösungsast übergeht. Die Höhe dieser Grenzlast ist abhängig von der Größe der Imperfektion  $\varepsilon$ . Unter dem Begriff der Imperfektionsempfindlichkeit versteht man die Abhängigkeit der Grenzlast  $\lambda_{Gr}(\varepsilon)$  von der Imperfektionsgröße. Auch bei der unsymmetrischen Verzweigung stellt sich für  $\varepsilon > 0$  eine Grenzlast ein, für  $\varepsilon < 0$  ergibt sich aber ein vollständig stabiler Lösungsast ohne Grenzlast. Hier ist also das Vorzeichen der Imperfektion auch entscheidend für das Strukturverhalten.

Um im Rahmen der Methode der Finiten Elemente die Imperfektionsempfindlichkeit zu ermitteln, sind entweder die unterschiedlichen mechanischen Strukturen mit dem vorgegebenen Imperfektionsmuster zu modellieren, und anschließend ist die Beullast durch eine nichtlineare Analyse zu ermitteln. Alternativ stehen unter dem Begriff der Parametersensitivität Algorithmen zur Verfügung [39, 79, 103], die es erlauben, die Änderung des Systemverhaltens aufgrund von Parameteränderungen (wie z.B. die Größe der Imperfektion  $\varepsilon$ , aber auch geometrische Größen) zu berechnen. Die Vorgehensweise gleicht dabei der direkten Berechnung von singulären Punkten, auch hier werden zu dem nichtlinearen Gleichungssystem Bedingungen hinzugefügt. Aufgrund der Vielfalt der möglichen Parameter kann die Linearisierung für eine Lösung mit dem Newton Verfahren dann nur noch numerisch erfolgen. Mit dieser Lösung kann dann die Abhängigkeit  $\lambda_{Gr}(\varepsilon)$  direkt bestimmt werden, ohne eine Vielzahl von Systemen zu modellieren.

### 7.3.7 Zwischenbilanz zu statischen Stabilitätsanalysen mittels Finiten Elemente

Mittels der Methode der Finiten Elemente ist es in einfacher Weise möglich, die Stabilität oder Instabilität von berechneten Gleichgewichtslagen konservativer Systeme zu bestimmen. Einen größeren Aufwand stellt es dar, singuläre Punkte mit hoher Genauigkeit zu

berechnen, da sich im Bereich singulärer Tangentensteifigkeiten die Konvergenzordnung des Newton-Raphson Verfahrens von quadratisch auf linear reduziert. Auch begleitend durchgeführte Eigenwertanalysen stellen einen zusätzlichen Aufwand dar. Die gefundenen singulären Punkte können klassifiziert werden, d.h. es kann zwischen Durchschlag- und Verzweigungspunkten unterschieden werden. Selbst der Typ des Verzweigungspunktes kann ermittelt werden.

Der dazu notwendige Berechnungsaufwand indes ist beträchtlich, wenn auf diese Art und Weise das vollständige Strukturverhalten ermittelt werden soll, da jeder der gefundenen Äste verfolgt werden muß. Dabei muß auch sichergestellt sein, daß kein Ast übersprungen wird und kein Lösungspfad an einer – möglicherweise mehrfachen – Verzweigung unberücksichtigt bleibt. Weiter ist zu beachten, daß z.B. durch die Modellierung unter Ausnutzung von Symmetrien unsymmetrische Beulformen ausgeschlossen werden. Zu Letzterem ist in [19] ein Verfahren angegeben, mit dem unter Ausnutzung von Symmetrieeigenschaften auch unsymmetrische Beulformen bestimmt werden können. Allerdings ist durch die Modellierung der jeweils symmetrischen/antisymmetrischen Randbedingungen der Teilstrukturen die Lösung des Problems vorweggenommen.

Dennoch kann die Pfadverfolgung bei komplexen Systemen äußerst mühsam sein, da bei häufigem Richtungswechsel des Lastparameters automatische Verfahren oft versagen und meist ein Prädiktorschritt und eine neue Bogenlänge von Hand vorgegeben werden müssen, um weiterhin Konvergenz innerhalb des Newton Verfahrens zu erzielen. In Abschnitt 8 soll auf solche Probleme anhand eines konkreten Beispiels genauer eingegangen werden.

Hingegen ist es mittels statischer Finite Element Untersuchungen nicht möglich, Lösungsäste zu ermitteln, die nicht mit dem natürlichen Last-Verformungspfad zusammenhängen, d.h. keinen gemeinsamen Punkt mit ihm haben. Dies ist häufig bei imperfekten Strukturen der Fall; als einfaches Beispiel sei hier der 1. Eulerfall genannt, der im Folgenden noch diskutiert wird. Das zugehörige Last-Verformungsdiagramm der imperfekten Struktur ist in Abbildung 7.2<sub>1</sub> dargestellt.

### 7.3.8 Perfekte und imperfekte Systeme

Es ist hinlänglich bekannt, daß hochoptimierte Schalentragwerke, bei denen die Last hauptsächlich über die Membranwirkung abgetragen wird, besonders empfindlich gegenüber geometrischen Imperfektionen sind. Daher sollten die Imperfektionen bei der numerischen Untersuchung berücksichtigt werden.

Gegen die Berücksichtigung der geometrischen Imperfektionen spricht aber, daß die tatsächlichen Imperfektionen im Planungszustand noch nicht bekannt sind. Außerdem ist ihre Erfassung nur mit großem Meßaufwand realisierbar, sodaß Untersuchungen an der perfekten Struktur wünschenswert sind. Bei der Verwendung künstlicher Imperfektionen hingegen besteht keine Sicherheit, daß diese ausgewählten Formen das tatsächliche Verhalten des realen Systems beschreiben.

Für die Berücksichtigung spricht aber vor allem, daß nur damit die Systemeigenschaften

ten richtig erfaßt werden können und Beullasten nicht überschätzt werden. Ein weiteres gewünschtes Verhalten bei der numerischen Lösung ist das Verschwinden von Verzweigungspunkten, die bei geeigneter Wahl der Imperfektionen eliminiert werden. Dadurch wird der Lösungsprozess stark vereinfacht. Insbesondere bei der Anhäufung von Nulleigenwerten bei perfekten Strukturen, sogenannten *mode-jumping* Phänomenen, folgen nach der ersten Verzweigung weitere sekundäre Verzweigungen, die ein statisches Verfolgen der Lösungspfade zwar nicht unmöglich machen [124, 52], aber doch mit einem großen numerischen und algorithmischen Aufwand verbinden, sodaß dies für die Praxis nur begrenzt attraktiv ist. In einigen Fällen ist es sogar erst mit Berücksichtigung von geometrischen Imperfektionen möglich, das im Experiment beobachtete qualitative Systemverhalten in der Simulation korrekt abzubilden [120].

Ein Ziel der Forschung besteht seit langem darin, die ungünstigste Imperfektionsform vorab zu ermitteln. In neueren Arbeiten [28, 29] werden auf Elementebene die Imperfektionen als zusätzliche Freiheitsgrade eingeführt, alle Linearisierungen konsistent durchgeführt und in das nichtlineare Gleichungssystem mit dem Ziel eingebracht, diejenigen Imperfektionen zu ermitteln, die zu einer minimalen Beullast der Struktur führen. Damit kann vorab eine ungünstige Imperfektionsform bestimmt werden, mit der dann Berechnungen an einer imperfekten Geometrie durchgeführt werden können. Die einzigen notwendigen Vorgaben bestehen in einer maximalen Imperfektionsamplitude und Startwerten für die Imperfektionsform. Allerdings ist mit diesem Verfahren nicht sichergestellt, daß wirklich die ungünstigste Imperfektionsform bestimmt wurde, da die Lösung von den Startwerten abhängt, siehe [28].

Insgesamt läßt sich vom Standpunkt der praktischen Anwendung feststellen, daß Simulationen unter Berücksichtigung von geometrischen Imperfektionen realistischere Ergebnisse liefern und zudem oft numerisch weniger problematisch zu behandeln sind. Die geometrischen Imperfektionen können sowohl aus Messungen oder aus numerischen Untersuchungen gewonnen werden. Bei Letzterem hat sich allerdings die Wahl von Imperfektionsformen in Gestalt der niedrigsten Eigenform nicht als zuverlässig gezeigt, nicht zuletzt, weil mit dieser meist symmetrischen Form damit etwaige Symmetrien eines Systems nicht entfernt werden.

Grundsätzlich sei festgestellt, daß mit der Ermittlung der Imperfektionsform, die zur niedrigsten Versagenslast führt, noch nicht bekannt ist, wie empfindlich das betrachtete Tragwerk in der Gleichgewichtslage auf Störungen reagiert, d.h. wie sensitiv die Gleichgewichtslage ist.

## 7.4 Transiente Analysen

Die statische Pfadverfolgung von Lösungsästen liefert keinen Hinweis darauf, welche der möglichen Pfade in Wirklichkeit die wahrscheinlichsten sind. Da der reale Beulvorgang transient abläuft und die vorhandenen Trägheiten das Beulverhalten beeinflussen, ist es naheliegend, die Simulation unter Berücksichtigung dieser Gegebenheiten durchzuführen. Für praktische Anwendungen ist nämlich nicht die Vielzahl von möglichen Gleichge-

wichtslagen und deren Stabilität, sondern vielmehr das tatsächliche Strukturverhalten und die Sensitivität gegenüber endlichen Störungen von Interesse. Durchgeführte Simulationen sollten also das reale Strukturverhalten möglichst genau wiedergeben können, weshalb transiente Analysen unter Berücksichtigung der Massenträgheit zur Berechnung des Beulverhaltens – wie z.B. in [120, 64, 80, 130] vorgeschlagen und durchgeführt – naheliegend sind.

Im Rahmen der räumlichen Diskretisierung mit der Methode der Finiten Elemente werden dabei die Bewegungsgleichungen (2.72) oft mittels eines Verfahrens vom Newmark Typ (siehe Abschnitt 2.3.3) gelöst. Dabei ergibt sich eine modifizierte Steifigkeitsmatrix (2.82), die im Zuge einer Lösung mit direkten Lösungsverfahren faktorisiert wird. Informationen über die Steifigkeitsmatrix selbst können während des Lösungsprozesses nur auf Umwegen erhalten werden.

Zur Untersuchung der Bewegung ist es zweckmäßig, die gekoppelten Differentialgleichungen zweiter Ordnung (2.72) in Differentialgleichungen 1. Ordnung zu transformieren. Dies geschieht durch Einführung des Vektors  $\mathbf{x} = [\mathbf{d} \ \dot{\mathbf{d}}]^T$ . Man erhält dann aus (2.72)

$$\dot{\mathbf{x}} = \left\{ \begin{array}{c} \dot{\mathbf{d}} \\ -\mathbf{M}^{-1}[\mathbf{r}(\mathbf{d}) + \mathbf{C}\dot{\mathbf{d}} - \mathbf{p}(t)] \end{array} \right\}, \quad (7.44)$$

und die im Rahmen einer nichtlinearen Lösung linearisierten Bewegungsgleichungen ergeben sich zu

$$\begin{aligned} \dot{\mathbf{x}} &= \begin{bmatrix} \dot{\mathbf{d}} \\ \ddot{\mathbf{d}} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K}_T & -\mathbf{M}^{-1}\mathbf{C}_T \end{bmatrix} \begin{bmatrix} \mathbf{d} \\ \dot{\mathbf{d}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{p} \end{bmatrix} \\ &= \mathbf{A}_T \mathbf{x} + \mathbf{b}. \end{aligned} \quad (7.45)$$

In den folgenden Abschnitten sollen die in Kombination mit der Methode der Finiten Elemente üblicherweise angewandten Verfahren zur Beurteilung der Stabilität kurz eingeführt werden. Das erste Verfahren, das auf der 1. Methode Ljapunovs beruht, ist ausführlich in [17, 18, 85] beschrieben, das zweite ist [100] und [37] entnommen.

#### 7.4.1 Gleichung der ersten Näherung

Mit der 1. Methode nach LJAPUNOV versucht man, Aussagen über die Stabilität einer Lösung  $\mathbf{x}(\mathbf{a}, t_0; t)$  von

$$\dot{\mathbf{x}} = \mathbf{A}_T \mathbf{x} + \mathbf{h}(\mathbf{x}, t) \quad (7.46)$$

anhand der linearisierten Differentialgleichung zu gewinnen. Analog zu Abschnitt 7.1.2 kann auch hier die Koordinatentransformation

$$\mathbf{z}(t) = \mathbf{x}(\mathbf{x}_0, t_0; t) - \mathbf{x}(\mathbf{a}, t_0; t) \quad (7.47)$$

durchgeführt werden, sodaß dann die triviale Lösung  $z(t) = \mathbf{0}$  des Differentialgleichungssystems

$$\dot{z} = \bar{\mathbf{A}}_T z + \bar{\mathbf{h}}(z, t) \quad (7.48)$$

untersucht wird. Die Stabilität dieser Bewegung soll dann anhand der linearisierten Differentialgleichung

$$\dot{z} = \bar{\mathbf{A}}_T z \quad (7.49)$$

gewonnen werden [97, 58]. In  $\bar{\mathbf{h}}(z, t)$  werden alle nichtlinearen Glieder zusammengefaßt. Dabei sollen Zahlen  $\alpha \geq 0$  und  $\beta > 1$  existieren, sodaß

$$|\bar{\mathbf{h}}(z, t)| \leq \alpha |z|^\beta \quad (7.50)$$

in einer Umgebung von  $z = \mathbf{0}$  gilt. Zur Anwendung dieses Kriteriums wird die Bewegungsgleichung der Differenzbewegung benötigt. Die zu untersuchende Bewegung stellt dann die triviale Lösung dar.

Für eine Bewegung mit der Koordinate

$$\mathbf{d}_s = \mathbf{d} + \tilde{\mathbf{d}}, \quad (7.51)$$

wobei der Differenzvektor zwischen der Nachbarbewegung  $\mathbf{d}_s$  und der zu untersuchenden Bewegung  $\mathbf{d}$  mit  $\tilde{\mathbf{d}}$  bezeichnet wird, gilt analog zu Gleichung (2.72)

$$\mathbf{M} \ddot{\mathbf{d}}_s + \mathbf{C} \dot{\mathbf{d}}_s + \mathbf{r}(\mathbf{d}_s) = \bar{\mathbf{p}}. \quad (7.52)$$

Für beide Bewegungen gelte der gleiche Belastungsverlauf  $\mathbf{p}(t)$ . Eine Taylorreihenentwicklung von  $\mathbf{r}(\mathbf{d}_s)$  liefert

$$\mathbf{r}(\mathbf{d}_s) = \mathbf{r}(\mathbf{d} + \tilde{\mathbf{d}}) = \mathbf{r}(\mathbf{d}) + \frac{\partial \mathbf{r}(\mathbf{d})}{\partial \mathbf{d}} \tilde{\mathbf{d}} + \dots = \mathbf{r}(\mathbf{d}) + \mathbf{K}_T^s \tilde{\mathbf{d}} + \dots \quad (7.53)$$

Bricht man die Reihenentwicklung (7.53) nach dem linearen Glied ab, setzt dies in (7.52) unter Berücksichtigung von (7.51) und subtrahiert (2.72), so erhält man

$$\mathbf{M} \ddot{\tilde{\mathbf{d}}} + \mathbf{C} \dot{\tilde{\mathbf{d}}} + \mathbf{K}_T^s \tilde{\mathbf{d}} = \mathbf{0}. \quad (7.54)$$

Nimmt man zudem an, daß die Steifigkeitsmatrix  $\mathbf{K}_T^s$  mit  $\mathbf{K}_T$  übereinstimmt, erhält man (7.49) als linearisierte Differentialgleichung 1. Ordnung mit  $\mathbf{A}_T$  aus (7.45). Mit einem Exponentialansatz erhält man daraus die charakteristische Gleichung

$$\det(\mathbf{A}_T - \lambda \mathbf{I}) = 0, \quad (7.55)$$

anhand deren Wurzeln die Stabilitätsfrage entschieden werden kann. Sind alle Wurzeln  $\lambda$  negativ, ist die Bewegung stabil, ist mindestens eine Wurzel Null oder positiv, liegt Instabilität vor. Mit  $\mathbf{A}_T$  aus (7.45) ergibt sich die Beziehung

$$\det(\lambda^2 \mathbf{M} + \lambda \mathbf{C}_T + \mathbf{K}_T) = 0, \quad (7.56)$$

unter Vernachlässigung von Dämpfung vereinfacht sich (7.56) zu

$$\det(\lambda^2 \mathbf{M} + \mathbf{K}_T) = 0. \quad (7.57)$$

Da die Dämpfung das Stabilitätsverhalten maßgeblich mitbestimmt, indem sie den Einzugsbereich von Gleichgewichtslagen vergrößert, befindet man sich durch deren Vernachlässigung auf der sicheren Seite. Der Übergang zu instabilem Systemverhalten erfolgt dann für  $\lambda^2 = 0$ , wenn das System seine Schwingungsfähigkeit verliert, d.h. negative oder Null-eigenfrequenzen existieren und unbegrenzt anwachsende Bewegungen möglich sind. Aus (7.57) ergibt sich also für den transienten Fall das gleiche Kriterium

$$\det(\mathbf{K}_T) = 0, \quad (7.58)$$

wie für den statischen Fall, siehe Gleichung (7.12). BURMEISTER [17] deutet dies als ein "Einfrieren im Inkrement", d.h. der Bewegungszustand zur Zeit  $t$  wird festgehalten und dann die Bewegung aufgrund von Störungen untersucht. Damit gilt die getroffene Aussage auch nur für diese ausgewählte Laststufe. Es ist klar, daß dieses Verfahren, das durch die Linearisierung und die Annahme  $\mathbf{K}_T^s = \mathbf{K}_T$  geprägt ist, nicht das vollständige nichtlineare Verhalten beschreiben kann. Einigkeit besteht jedoch darüber, daß es einen Indikator für Instabilität darstellt. In [20] wird dieses Kriterium der statischen Stabilität auch als Insensitivität (und analog statische Instabilität als Sensitivität) für nichtlineare dynamische Systeme bezeichnet.

Im Rahmen des Newmark Verfahrens ist es, im Gegensatz zu statischen Berechnungen, nicht mehr möglich, die Information über die Determinante der tangentiellen Steifigkeitsmatrix ohne zusätzlichen Aufwand aus dem Lösungsprozess zu erhalten, da nun ein lineares Gleichungssystem mit einer modifizierten Koeffizientenmatrix (2.82) gelöst wird. Entweder muß die tangentielle Steifigkeitsmatrix in jedem Lastinkrement zusätzlich faktorisiert werden, was den Lösungsaufwand verdoppeln würde, oder der kleinste Eigenwert der Steifigkeitsmatrix wird auf anderem Wege, wie z.B. in [153] vorgeschlagen, bestimmt. Im letzten Fall wird der kleinste Eigenwert für das lineare Problem bestimmt und während der nichtlinearen Zeitverlaufsrechnung in jedem Inkrement iterativ berechnet, wobei der Wert aus dem vorherigen Inkrement als Näherungswert verwendet wird.

#### 7.4.2 Dynamische Stabilitätsuntersuchungen mittels Ljapunov-Exponenten

In [100] und [37] wird vorgeschlagen, die Untersuchung der Stabilität einer numerisch ermittelten Lösung unter Verwendung der sogenannten Ljapunov-Exponenten durchzuführen

ren. Grundlage hierfür ist die 1. Methode nach Ljapunov. Nach Berechnung der ungestörten Lösung  $\mathbf{x}(\mathbf{a}, t_0; t)$  werden zusätzlich Nachbarbewegungen  $\mathbf{x}(\mathbf{x}_0, t_0; t)$  berechnet; die Differenz beider Lösungen wird mit  $\tilde{\mathbf{x}} = \mathbf{x}(\mathbf{x}_0, t_0; t) - \mathbf{x}(\mathbf{a}, t_0; t)$  bezeichnet. Die Stabilität von  $\mathbf{x}(\mathbf{a}, t_0; t)$  wird direkt mit (7.2) und (7.3) entschieden. Dazu wird der Ljapunov-Exponent als

$$\vartheta(\mathbf{x}, \mathbf{x}_0 - \mathbf{a}) = \lim_{t \rightarrow \infty} \frac{1}{t - t_0} \ln \frac{\|\tilde{\mathbf{x}}\|}{\|\mathbf{x}_0 - \mathbf{a}\|} \quad (7.59)$$

definiert [86]. Wächst die Entfernung zwischen  $\mathbf{x}(\mathbf{x}_0, t_0; t)$  und  $\mathbf{x}(\mathbf{a}, t_0; t)$  mit der Zeit an (wird also größer als  $\|\mathbf{x}_0 - \mathbf{a}\|$ ), ist der Ljapunov-Exponent  $\vartheta$  größer Null, andernfalls kleiner Null.

Dieser Vorgang wird dann für jede linear unabhängige Richtung des Phasenraumes wiederholt, über die Stabilität entscheidet die Nachbarlösung mit dem größten Exponenten. Darin liegt aber auch der Nachteil dieses Vorgehens; für Systeme mit wenigen Freiheitsgraden ist der Aufwand vertretbar. Für Systeme mit vielen Freiheitsgraden ist der Aufwand hingegen immens, das Verfahren dann nicht praktikabel.

## 7.5 Weitere Beurteilung stabiler oder instabiler Gleichgewichtslagen

Der Begriff der Stabilität erlaubt die Angabe einer Qualität einer Gleichgewichtslage (oder auch einer Bewegung, was im Folgenden nicht betrachtet werden soll). Ist die Gleichgewichtslage stabil, so kann sie physikalisch realisiert werden. Ist sie aber instabil, kann die Gleichgewichtslage nicht realisiert werden. Die mechanische Struktur wird im Falle von Instabilität eine andere, stabile Gleichgewichtslage einnehmen.

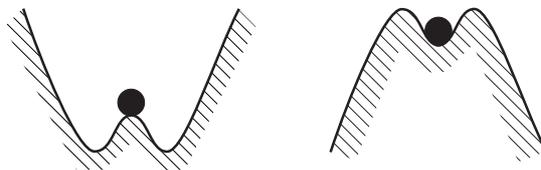


Abbildung 7.3: Praktische Stabilität

Die Stabilitätsaussage nach LJAPUNOV sagt aber nichts über das Verhalten der Struktur bei endlichen Störungen aus, sondern gilt nur für infinitesimal kleine Störungen. Zur Veranschaulichung der 2. Methode nach LJAPUNOV mit der sogenannten Kugelanalogie betrachtet man die Potentialfläche im Schwerfeld als materielle Bahn eines Massenpunktes. Man erkennt damit leicht, daß die in Abbildung 7.3 links dargestellte Gleichgewichtslage zwar instabil ist, für praktische Anwendung aber durchaus interessant sein kann, da eine kleine Störung nur zu einer begrenzten Verschiebung führt. Umgekehrt verhält es sich mit der

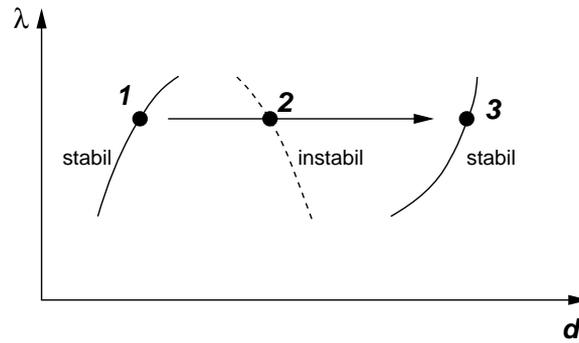


Abbildung 7.4: Sensitivität im Last-Verformungsdiagramm

rechts dargestellten, stabilen Gleichgewichtslage. Infinitesimal kleine Störungen verursachen kleine Bewegungen um die Gleichgewichtslage. Reale, endliche Störungen hingegen führen zu großen Verschiebungen. Aus praktischen Gesichtspunkten ist diese Gleichgewichtslage als “praktisch instabil” zu bezeichnen.

Wünschenswert wäre eine Quantifizierung des Stabilitätsbegriffs, um die Sicherheit, d.h. Sensitivität einer Gleichgewichtslage zu bewerten. LEIPHOLZ definiert hierzu den Stabilitätsgrad [92] als minimalen Radiusvektor zur nächsten Separatrix, das bedeutet die kleinste mögliche Störung, die zum Verlassen des stabilen Orbits führt. In der Darstellung in einem Last-Verformungsdiagramm, siehe Abbildung 7.4, führt dies zu der Frage, wie man von einer stabilen Gleichgewichtslage “1” über eine instabile Gleichgewichtslage “2” hinweg zu einer weiteren stabilen Gleichgewichtslage “3” gelangt, *ohne* der Last-Verformungskurve zwischen den Gleichgewichtslagen zu folgen.

VIELSACK ET AL. [145] definiert Sensitivität als reziproken Wert einer Potentialdifferenz, die als invariante Größe die Eigenschaften einer stabilen Gleichgewichtslage beschreibt. Die Potentialdifferenz zwischen einer stabilen und einer benachbarten instabilen Gleichgewichtslage charakterisiert die maximal ertragbare endliche Störung. Eine solche Potentialdifferenz kann für dynamische Vorgänge durch einen Impuls überwunden werden.

In ähnlicher Weise wird in [87] und nachfolgenden Arbeiten [65, 31, 146, 140, 137] ein Störenergiekonzept entwickelt, das die Qualität einer stabilen Gleichgewichtslage mittels einer sogenannten Störenergie bewertet. Dazu wird zunächst zur Beurteilung der Gleichgewichtslage  $\hat{\mathbf{d}}$  eine Störform  $\mathbf{d}_{St}$  festgelegt und anschließend die Störenergie  $\Pi_{St}$  berechnet, die notwendig ist, um Instabilität zu erreichen [65]. Es wird angenommen, daß die erste Eigenform des Eigenwertproblems (7.22) die Richtung angibt, für die das Tragwerk einer Störung den geringsten Widerstand entgegensetzt. Dann wird die Störenergie definiert als

$$\Pi^{St} = \Delta \mathbf{d} \cdot \mathbf{K}_{NL} \phi_1 = \Delta \mathbf{d} \cdot \mathbf{p}_{St}, \quad (7.60)$$

wobei sich  $\Delta \mathbf{d}$  aus einem Vielfachen der Störform  $\phi_1$  und einem dazu senkrechten Anteil  $\Delta \mathbf{d}_\perp$  zusammensetzt, der mit dem Störvektor  $\mathbf{p}_{St}$  keine Arbeit verrichtet. Es ist

$$\Delta \mathbf{d} = \gamma \phi_1 + \Delta \mathbf{d}_\perp. \quad (7.61)$$

Bei einer Normierung  $\phi_1^T \mathbf{K}_{NL} \phi_1 = 1$  ergibt sich somit für die Störenergie

$$\Pi^{St} = \gamma. \quad (7.62)$$

Das System geht also bei einer Störung, die größer als  $\gamma\phi_1$  ist, in eine andere Beulform über, sofern Linearität vorausgesetzt wird. Der Faktor  $\gamma$  wird nach [146] aus dem Eigenwertproblem

$$(\mathbf{K}_T + \gamma\mathbf{K}_{NL}(\phi_1))\Phi = \mathbf{0} \quad (7.63)$$

bestimmt, d.h. der Störvektor  $\phi_1$  wird so lange vergrößert, bis die Tangentensteifigkeit singular wird.

Numerisch wird also in der Gleichgewichtslage  $\hat{\mathbf{d}}$  der Vektor  $\gamma\phi_1$  addiert und versucht, die benachbarte Gleichgewichtslage anzusteuern. Dies kann mißlingen, da über  $\Delta\mathbf{d}_\perp$  keine Aussage gemacht wurde und außerdem  $\phi_1$  aus einer linearen Abschätzung stammt. Es ist demnach nicht abgesichert, daß die vermutete Gleichgewichtslage sich dort befindet. Aus ebendiesen Gründen ist es auch von großer Wichtigkeit, ein überaus robustes Iterationsverfahren zur Ansteuerung der benachbarten Gleichgewichtslage zu verwenden, worauf in [87] ausdrücklich hingewiesen wird.

## 7.6 Auffinden nichteindeutiger Gleichgewichtslagen und Einführung des Sensitivitätsbegriffes

Für beide im vorherigen Abschnitt dargestellten Verfahren ist es von zentraler Bedeutung, die in Abbildung 7.4 mit "1" bis "3" gekennzeichneten Gleichgewichtslagen zu kennen bzw. zu bestimmen. Bei dem in [145] gewählten Problem ist es aufgrund der expliziten Kenntnis der Potentialfunktion möglich, alle Gleichgewichtslagen zu bestimmen. Dies ist bei allgemeinen Untersuchungen komplexerer Systeme mit der Methode der Finiten Elemente nicht der Fall. Es können die Gleichgewichtslagen nur iterativ entlang eines Lösungspfad es ermittelt werden, der sich eventuell auch verzweigen kann. Existieren jedoch unzusammenhängende Lösungspfade, so kann ein Lösungspfad vom anderen nicht mittels statischer Berechnungen durch einfache Lastveränderung erreicht werden.

Ausgehend von bekannten Gleichgewichtslagen auf dem natürlichen Lösungsast soll anhand von Stabilitätsproblemen mit unterschiedlichem Verzweigungsverhalten, wie symmetrische stabile und symmetrische instabile Verzweigung sowie für ein Durchschlagproblem eine Vorgehensweise zum Auffinden von weiteren Gleichgewichtslagen beim gleichen Lastniveau vorgeschlagen werden. Darüber hinaus wird der Begriff der Sensitivität zur quantitativen Beurteilung einer stabilen Gleichgewichtslage definiert.

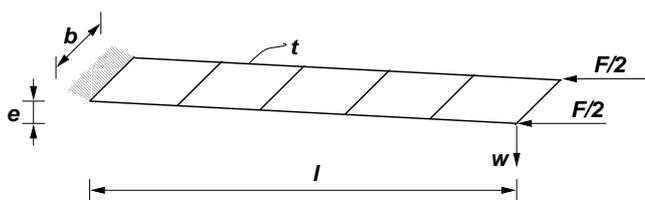
### 7.6.1 Symmetrisch stabiles Systemverhalten

Ein einfaches Beispiel für den Fall der symmetrisch stabilen Verzweigung ist der Eulerstab [139]. Es soll also im Folgenden ein imperfekter Kragarm unter Normalkraftbelastung, siehe Abbildung 7.5, betrachtet werden. Die Imperfektion besteht in einer Schrägstellung  $e$  des Kragarms. Als charakteristische Verschiebung wird die Endverschiebung  $w$  gewählt. Dies ist aufgrund des bekannten Verhaltens des Eulerstabes sinnvoll. Im Falle einer perfekten Geometrie ( $e = 0$ ) tritt bei steigender Belastung keine Querverschiebung  $w$  auf. Beim Erreichen eines – hier symmetrisch stabilen – Verzweigungspunktes erfolgt dann ein Ausweichen des Kragarmes nach oben oder unten. Zwar ist auch für  $w = 0$  weiterhin Gleichgewicht möglich, dieses ist aber ab der Verzweigungslast instabil, siehe Abbildung 7.1<sub>1</sub>.

Für die in Abbildung 7.5 angegebenen Daten ist in Abbildung 7.6 das Last-Verformungsverhalten gegeben. Bei imperfekter Geometrie verschwindet der Verzweigungspunkt. Die Lösungsäste sind unzusammenhängend, vergleiche Abbildung 7.2<sub>1</sub>. Bei einer statischen Untersuchung mit einfacher Laststeigerung erhält man nur Verschiebungen  $w > 0$  als Lösung. Keine der berechneten Gleichgewichtslagen ist instabil. Der Schluß, es handele sich hierbei nicht um ein Stabilitätsproblem liegt nahe (und wird auch getroffen, siehe [146]). Es existieren jedoch ab einer gewissen Last weitere Lösungen mit  $w < 0$  und es stellt sich die Frage, wie diese Gleichgewichtslagen systematisch z.B. mit der Methode der Finiten Elemente bestimmt werden können.

Mittels statischer Berechnungen ist dies nur möglich, indem das gewählte Modell verändert wird, z.B. durch Einführung einer zusätzlichen Last senkrecht zu  $F$ . Diese könnte z.B. in der negativen  $w$ -Richtung aufgebracht werden und nach Erreichen des zweiten Lösungsastes wieder entfernt werden. Damit wird aber hier im Spezialfall die bekannte Lösung genutzt, d.h. dies stellt keine allgemein durchführbare Vorgehensweise dar.

Im Folgenden wird eine Vorgehensweise zum Auffinden nichteindeutiger Gleichgewichtslagen vorgeschlagen. Dazu wird zunächst mittels einer statischen Analyse eine statische Gleichgewichtslage bei einem vorgegebenen Lastniveau bzw. Lastparameter bestimmt.



#### Geometrie:

$$l = 100 \text{ mm}$$

$$b = 10 \text{ mm}$$

$$e = 0.1 \text{ mm}$$

$$t = 1 \text{ mm}$$

#### Material: linear elastisch

$$E = 2.1 \cdot 10^5 \text{ N/mm}^2$$

$$\rho = 7.87 \cdot 10^{-6} \text{ kg/mm}^3$$

#### Last:

$$F = 1 \text{ kN}$$

Abbildung 7.5: Geometrie und Materialdaten des imperfekten Kragarmes

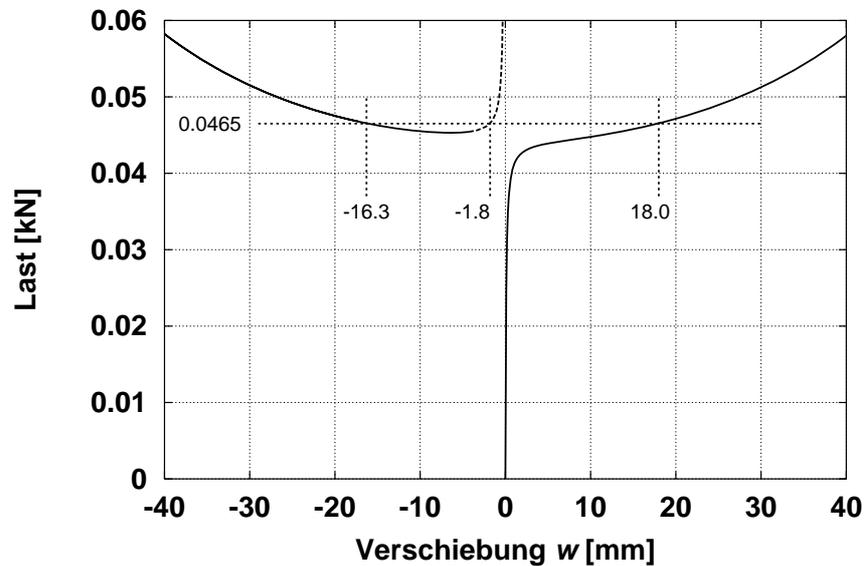


Abbildung 7.6: Imperfekter Kragarm – Last-Verformungskurve

Diese Belastung bleibt im Weiteren unverändert. In dieser Gleichgewichtslage wird dem mechanischen System kinetische Energie  $W_{kin}$  zugeführt, d.h. es werden Anfangsbedingungen für die nachfolgende numerische Integration der Bewegungsgleichungen gewählt. Als Anfangsverschiebungen  $\mathbf{d}_0$  werden dabei die aus dem statischen Gleichgewicht resultierenden Verschiebungen gewählt. Zusätzlich werden Anfangsgeschwindigkeiten  $\dot{\mathbf{d}}_0 = \mathbf{v}_0$  gewählt. Die zugeführte kinetische Energie ist dann durch  $W_{kin} = \frac{1}{2} \mathbf{v}_0^T \mathbf{M} \mathbf{v}_0$  gegeben. Die Wahl der Anfangsgeschwindigkeiten  $\mathbf{v}_0$  ist zunächst frei, verschiedene Möglichkeiten werden im Anschluß am konkreten Beispiel diskutiert. Damit ist ein Systemzustand definiert, der auch als momentaner Systemzustand während eines Schwingungsvorganges aufgefaßt werden kann. Mittels numerischer Integration kann dann das Anfangsrandwertproblem gelöst werden.

Zur Auswertung der berechneten Bewegung dienen zunächst Phasenportraits ausgewählter, charakteristischer Koordinaten. Im Falle einer gedämpften Bewegung kann im Phasenportrait eine stabile Gleichgewichtslage als attraktiver Punkt erkannt werden. Dies kann einerseits die bereits bekannte Gleichgewichtslage sein, aber andererseits auch eine weitere stabile Gleichgewichtslage, sofern eine solche existiert. Wenn mindestens eine weitere stabile Gleichgewichtslage existiert, so ist es nicht vorherbestimmbar, welche der möglichen Gleichgewichtslagen berechnet wird. Dies wird durch die Vorgabe der Anfangsgeschwindigkeit  $\mathbf{v}_0$  beeinflusst. In [135] wird das Rückkehren des Systems in die ursprüngliche Gleichgewichtslage bei vorhandener Dämpfung näher diskutiert. Im Falle einer dämpfungsfreien Bewegung können Gleichgewichtslagen dem Phasenportrait nicht ohne weiteres entnommen werden. Daher ist die Einführung einer Systemdämpfung für diese Berechnung sinnvoll. Es sei aber angemerkt, daß die Dämpfung den Einzugsbereich stabiler Gleichgewichtslagen verändert. Ein weiterer Vorteil der Dämpfung ist darin zu sehen, daß insbesondere höherfrequente Lösungsanteile, die aufgrund der Diskretisierung nicht gut abgebildet

werden, herausgefiltert werden können. Dies ist insbesondere durch numerische Dämpfung möglich und soll daher im Folgenden Verwendung finden.

Durch die bislang angegebene Vorgehensweise ist das Auffinden weiterer Gleichgewichtslagen eher zufällig und im Wesentlichen von der Vorgabe der Anfangsgeschwindigkeiten  $v_0$  abhängig. Dies betrifft sowohl die zugeführte Energie  $W_{kin}$ , d.h. den Betrag der Geschwindigkeiten, als auch die Form, d.h. das Verhältnis der Beträge der Komponenten von  $v_0$  zueinander. Es fehlt ein Kriterium, das die mögliche Erreichbarkeit einer Gleichgewichtslage anzeigt. Als solches wird die Norm des statischen Residuums verwendet, d.h. den Vektor der statischen Ungleichgewichtskräfte  $-r(\mathbf{d}) + \mathbf{p}$ , vergleiche Gleichung (2.81). Dieser Vektor muß im Newmark Verfahren berechnet werden, es entsteht also nur der zusätzliche Aufwand zur Bestimmung einer Norm.

Die obige Vorgehensweise wird nachfolgend anhand des Problems des 1. Eulerfalles veranschaulicht und diskutiert. Es wird zunächst eine statische Lastveränderung der Horizontallast  $F$  bis zu der zu untersuchenden Gleichgewichtslage durchgeführt. Diese Gleichgewichtslage wird bei  $\lambda = 0.0465$  gewählt. Dem System wird dann durch Vorgabe von Anfangsgeschwindigkeiten  $v_0$  kinetische Energie  $W_{kin}$  eingeprägt. Als Geschwindigkeitsverteilung wird die erste Schwingungseigenform  $\varphi_1$  der Struktur, d.h. der ersten Eigenvektor aus

$$(\mathbf{K}_L - \omega^2 \mathbf{M})\boldsymbol{\varphi} = \mathbf{0} \quad (7.64)$$

gewählt. Das System wird dadurch zu Schwingungen angeregt, die statische Last bleibt unterdessen konstant. In Abbildung 7.7 ist das Phasendiagramm für zwei unterschiedliche kinetische Energien dargestellt. Die stabilen Gleichgewichtslagen sind darin durch ein  $\bullet$ ,

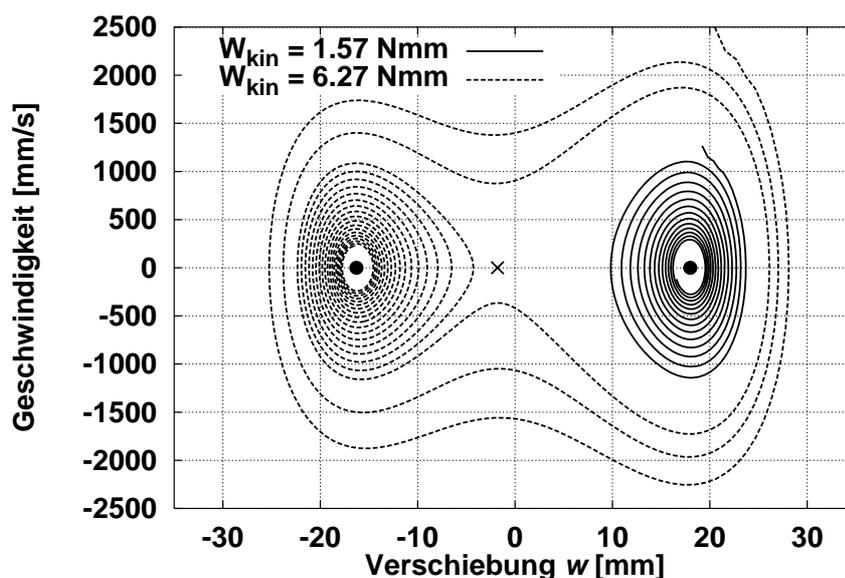


Abbildung 7.7: Imperfekter Kragarm – Phasendiagramm bei  $\lambda = 0.0465$ , Anfangsgeschwindigkeit in 1. Eigenform  $\varphi_1$

die instabile durch ein  $\times$  gekennzeichnet. Wie bei der statischen Last-Verformungskurve in Abbildung 7.6, wird auch hier die Endverschiebung  $w$  als charakteristische Koordinate gewählt. Die Größe der zugeführten kinetischen Energie wird durch den Skalar  $W_{kin} = \frac{1}{2} \mathbf{v}_0^T \mathbf{M} \mathbf{v}_0$  charakterisiert. Die Zeitintegration erfolgt mit dem Newmark Verfahren unter Einsatz numerischer Dämpfung<sup>11</sup>, wodurch die zugeführte kinetische Energie wieder dissipiert wird. Deutlich erkennbar ist in Abbildung 7.7, daß für die Energie  $W_{kin} = 1.57 \text{ Nmm}$  eine Schwingung um die mittels statischer Berechnung gefundene Gleichgewichtslage bei  $w = 18.0 \text{ mm}$  einsetzt. Oberschwingungen sind keine erkennbar, die erste Eigenform ist dominant. Dies ist eine Folge der Vorgabe von  $\mathbf{v}_0$ . Die Vorgabe der Geschwindigkeitsverteilung kann auch als momentaner Systemzustand einer Schwingung gedeutet werden, quasi als "Momentaufnahme". Beim Durchgang durch die Gleichgewichtslage, d.h. der Beginn der transienten Analyse, ist die Geschwindigkeit in der Form  $\varphi_1$  verteilt, und dominiert daher die nachfolgende Bewegung. Eventuell enthaltene höhere Eigenformen werden außerdem numerisch gedämpft. Nach Dissipation der gesamten kinetischen Energie befindet sich das System wieder in seiner Ausgangslage.

Bei einer höheren zugeführten kinetischen Energie  $W_{kin} = 6.27 \text{ Nmm}$  hingegen kommt das System bei einer zweiten, ebenfalls statischen Gleichgewichtslage bei  $w = -16.3 \text{ mm}$  zur Ruhe. Damit liegt eine zweite Gleichgewichtslage bei dem selben Lastniveau vor. Bei einer weiteren Vergrößerung der kinetischen Energie  $W_{kin}$  kann auch wieder die ursprüngliche Gleichgewichtslage bei  $w = 18 \text{ mm}$  berechnet werden. Für dieses Verhalten, d.h. für welche zugeführte Energie welche Gleichgewichtslage erhalten wird, ist die Dämpfung, die die Einzugsgebiete der Gleichgewichtslagen verändert, maßgeblich. Im dämpfungsfreien Fall entspricht der Übergang von einer Schwingung um die Gleichgewichtslage zum Durchschlagen dem Überschreiten der sogenannten Separatrix. Die Separatrix trennt in der Phasenebene einen Bereich periodischer Lösungen von einem Bereich, in dem aperiodische Lösungen oder periodische Lösungen von einem anderen Typ existieren. Bei nur schwacher Dämpfung unterscheidet sich daher die kinetische Energie, die zum Erreichen der zweiten Gleichgewichtslage mindestens notwendig ist, nur wenig von der Energie, die im dämpfungsfreien Fall zum Überschreiten der Separatrix benötigt wird. Unter der Voraussetzung schwacher Dämpfung kann also die Energie, die mindestens notwendig ist, um eine zweite Gleichgewichtslage zu erreichen, zur Bewertung der Stabilität einer stabilen Gleichgewichtslage – und damit der Sensitivität – verwendet werden.

In Abbildung 7.8 ist die Norm des statischen Residuums für die in Abbildung 7.7 bereits gezeigte Bewegung mit  $W_{kin} = 6.27 \text{ Nmm}$  dargestellt. Die Kurve besitzt drei deutliche Minima bei  $w = -16.3 \text{ mm}$ ,  $w = -1.8 \text{ mm}$  und  $w = 18.0 \text{ mm}$ . Dies entspricht den drei möglichen statischen Gleichgewichtslagen bei  $\lambda = 0.0465$ . Die instabile Gleichgewichtslage bei  $w = -1.8 \text{ mm}$  konnte vorher nicht lokalisiert werden.

Mit dieser zusätzlichen Information, die innerhalb des Newmark Verfahrens mittels geringem zusätzlichem Aufwand durch zusätzliche Bildung einer Norm erhalten wird, kann im Allgemeinen die Existenz einer Gleichgewichtslage zuerst nur vermutet werden. Eine statische Gleichgewichtslage wird für den Wert Null angezeigt. Es ist aber nicht zu erwarten, daß diese Situation während des Schwingungsvorganges genau erreicht wird. Daher

<sup>11</sup>Hierzu werden die Newmark Parameter  $\gamma = 0.9$  und  $\beta = 0.49$  gewählt.

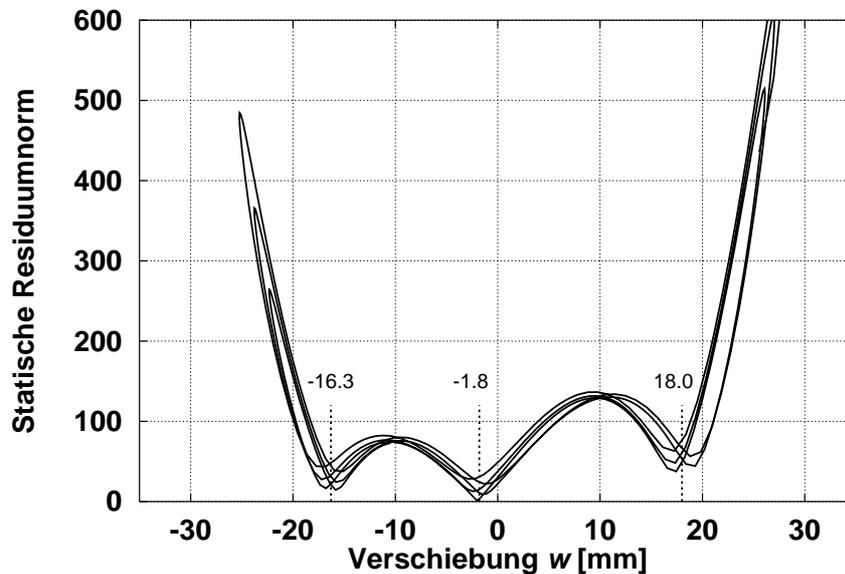


Abbildung 7.8: Imperfekter Kragarm – statisches Residuum, Anfangsgeschwindigkeit in 1. Eigenform,  $W_{kin} = 6.27 \text{ Nmm}$

sind kleine Werte des Residuums als Indikator für die Nähe einer Gleichgewichtslage zu interpretieren. Diese Information kann aber zu einer beschleunigten Bestimmung der benachbarten Gleichgewichtslage gut genutzt werden, indem in der Nähe der vermuteten Gleichgewichtslage stärker gedämpft, und so der Einzugsbereich dieser Gleichgewichtslage vergrößert wird. Die kinetische Energie wird dann schneller dissipiert und das System kommt in der statischen Gleichgewichtslage zur Ruhe. Es sei allerdings angemerkt, daß aufgrund der Änderung des Einzugsgebietes der Gleichgewichtslage durch die Rayleigh Dämpfung diese nicht mit Sicherheit bestimmt werden kann. In Abbildung 7.9 ist dieser Vorgang dargestellt. Zuerst wird dem System die kinetische Energie  $W_{kin} = 6.27 \text{ Nmm}$  in der ersten Eigenschwingungsform  $\varphi_1$  eingeprägt. Kurz vor Erreichen des Minimums des statischen Residuums wird Rayleigh Dämpfung eingeführt und so die kinetische Energie wieder dissipiert. Der Beginn der starken Dämpfung ist an den Knicken der Lösungskurven im Phasenportrait in Abbildung 7.9 deutlich erkennbar. Die Geschwindigkeit der Bewegung wird bei starker Dämpfung schnell reduziert. Für die beiden stabilen Gleichgewichtslagen bei  $w = -16.3 \text{ mm}$  und  $w = 18.0 \text{ mm}$ , wie aus Abbildung 7.7 ersichtlich, wird so der statische Gleichgewichtszustand problemlos erreicht. Bei der instabilen Gleichgewichtslage bei  $w = -1.8 \text{ mm}$  gelingt dies nicht, da das System stets in eine der beiden benachbarten stabilen Gleichgewichtslagen ausweicht. Nur wenn in unmittelbarer Nähe der instabilen Gleichgewichtslage auf statische Berechnung umgeschaltet wird, kann auch diese bestimmt werden.

Sind damit Lösungspunkte auf dem mit dem natürlichen Last-Verformungspfad nicht zusammenhängenden Lösungsästen bekannt, so können diese weiter mittels statischer Untersuchungen bestimmt werden, siehe Abbildung 7.6. Auch die Stabilitätsfrage der Gleichgewichtslagen auf diesen Lösungsästen kann mittels der in Abschnitt 7.2 angegebenen Krite-

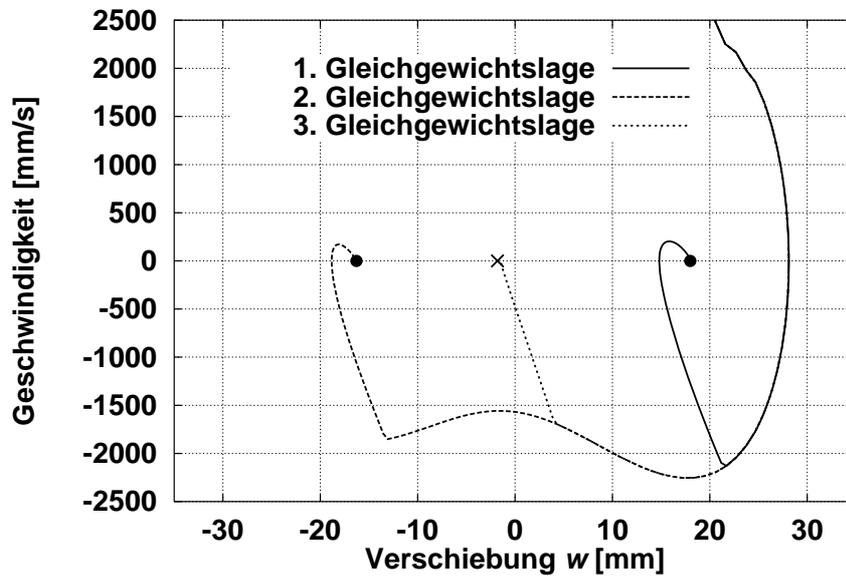


Abbildung 7.9: Imperfekter Kragarm – Phasendiagramm, Berechnung der statischen Gleichgewichtslagen

rien entschieden werden. Für das betrachtete Beispiel des imperfekten Kragarmes sind in Abbildung 7.6 die stabilen Gleichgewichtslagen mit einer durchgezogenen, die instabilen mit einer gestrichelten Linie gekennzeichnet.

Noch nicht näher diskutiert wurde die Form der Anfangsgeschwindigkeitsverteilung  $v_0$ ,

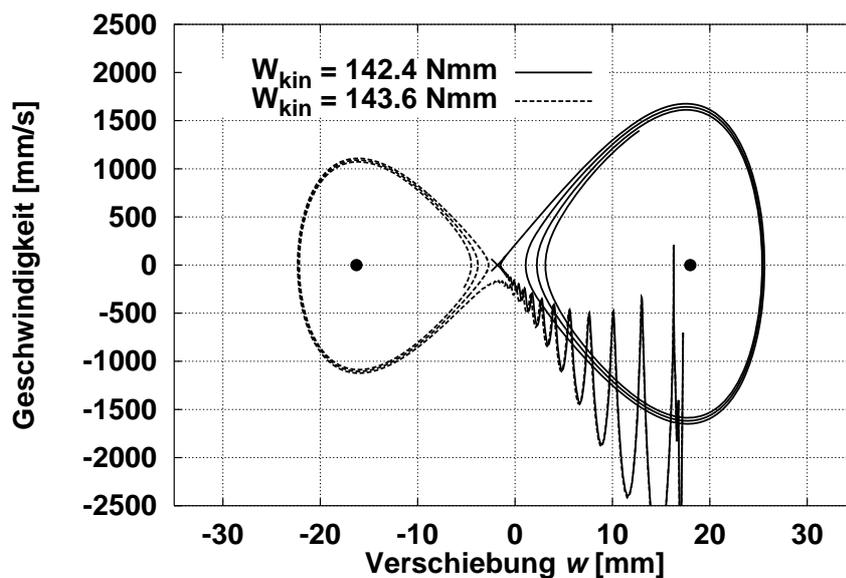


Abbildung 7.10: Imperfekter Kragarm – Phasendiagramm bei  $\lambda = 0.0465$ , Anfangsgeschwindigkeit in Lastrichtung

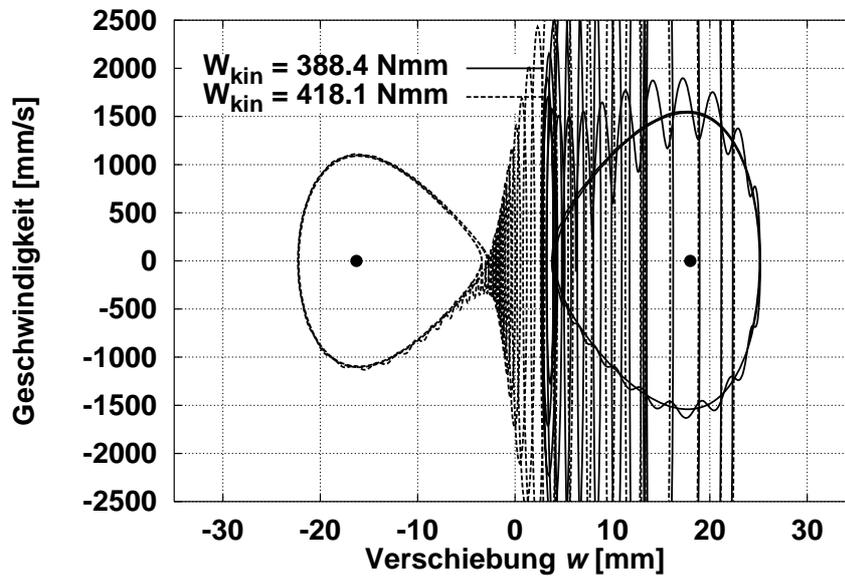


Abbildung 7.11: Imperfekter Kragarm – Phasendiagramm bei  $\lambda = 0.0465$ , Verteilung der Anfangsgeschwindigkeit in der 2. Eigenform (I)

mit der die kinetische Energie eingepreßt wird. Außer der gewählten 1. Schwingungseigenform können im Prinzip beliebige Verteilungen, sofern sie den Randbedingungen genügen, gewählt werden. So kann z.B. auch eine Anfangsgeschwindigkeitsverteilung vorgegeben werden, bei der nur die belasteten Freiheitsgrade eine Anfangsgeschwindigkeit ungleich Null erhalten. Dann ergibt sich das in Abbildung 7.10 gezeigte Phasendiagramm für unterschiedliche Größen der kinetischen Energie. Auch hiermit kann ein Übergang in die zweite stabile Gleichgewichtslage erreicht werden. Es ist aber offensichtlich, daß nun eine höhere Energie insbesondere wegen der Anregung von Längsschwingungen erforderlich ist, die aber zum Durchschlagen in Querrichtung kaum beitragen kann. Auch sind höherfrequente Anteile sichtbar, die durch die numerische Dämpfung reduziert werden. Die Schwingungsform geht dann in eine Schwingung in der ersten Eigenform über. Für hohe Werte von  $W_{kin}$ , verglichen mit der Anfangsgeschwindigkeitsverteilung in der ersten Eigenschwingungsform, wird der Übergang in die zweite stabile Gleichgewichtslage erreicht.

Ein ähnliches Verhalten ist für eine Anfangsgeschwindigkeitsverteilung in der 2. Schwingungseigenform  $\varphi_2$ , d.h. dem zum zweiten Eigenwert von (7.64) gehörenden Eigenvektor, zu beobachten, siehe Abbildung 7.11 und 7.12. In Abbildung 7.11 ist das Phasenportrait mit der gleichen Skalierung dargestellt wie in den vorherigen Abbildungen. Dabei treten bei der sich einstellenden Schwingung in der zweiten Eigenform hohe Geschwindigkeiten auf. Daher ist das Phasenportrait in Abbildung 7.12 mit einer anderen Skalierung erneut gezeigt. Hier ist der Übergang von der zweiten in die erste Schwingungseigenform gut erkennbar. Zum Übergang in die zweite stabile Gleichgewichtslage ist bei dieser Anfangsgeschwindigkeitsverteilung eine deutlich höhere kinetische Energie erforderlich als bei den zuvor untersuchten Anfangsgeschwindigkeitsverteilungen. Dieses kann dadurch begründet werden, daß zur Anregung von Schwingungen in niedrigen Eigenformen weniger Energie

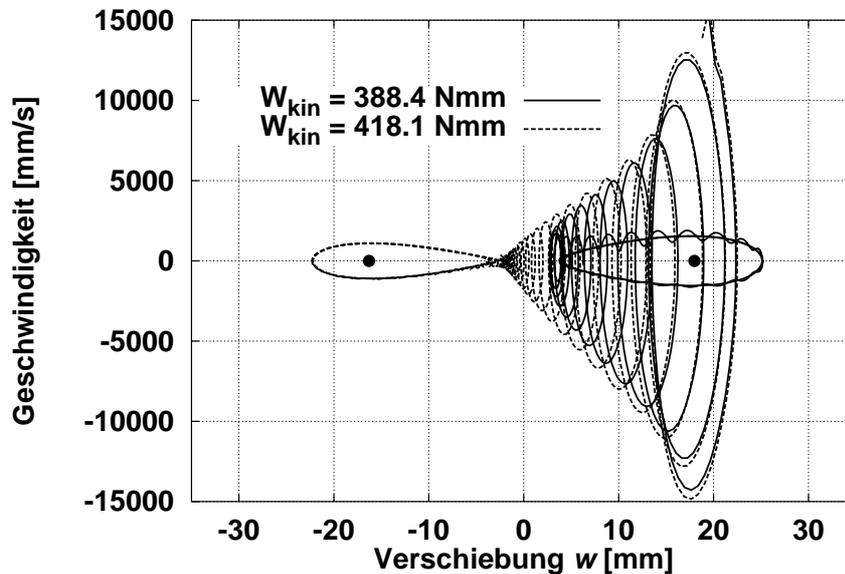


Abbildung 7.12: Imperfekter Kragarm – Phasendiagramm bei  $\lambda = 0.0465$ , Verteilung der Anfangsgeschwindigkeit in der 2. Eigenform (II) (Skala gegenüber 7.11 geändert)

erforderlich ist, als zur Anregung in höheren Eigenformen. Bei einer allgemeinen Anregung, d.h. durch eine Anregung, die nicht ausschließlich aus einzelnen Eigenformen besteht, ist im allgemeinen Fall auch die erste Eigenform enthalten. Daher ist zu erwarten, daß zur Anregung in einer einzelnen höheren Eigenform mehr Energie notwendig ist.

Zur Beurteilung der Sensitivität der stabilen Gleichgewichtslage wird nun die Anfangsgeschwindigkeitsverteilung  $v_0$  herangezogen, bei der das Erreichen der zweiten stabilen Gleichgewichtslage mit minimaler Zuführung an kinetischer Energie gelingt. Dies ist für  $v_0 = \varphi_1$  der Fall, wie aus den vorangegangenen Untersuchungen hervorgeht. Für alle anderen Anfangsgeschwindigkeitsverteilungen sind deutlich höhere Energien  $W_{kin}$  erforderlich. In Abbildung 7.13 ist der kritische Fall dargestellt. Mit der kinetischen Energie  $W_{kin} = 3.48 \text{ Nmm}$  wird die zweite stabile Gleichgewichtslage gerade erreicht. Dies ist also die kleinstmögliche Energie, die dem System zugeführt werden muß, um diese stabile Gleichgewichtslage zu verlassen. Der Einfluß der eingeführten Dämpfung ist dabei vernachlässigt, es ist aber aus Abbildung 7.13 ersichtlich, daß die Dämpfung klein ist; die Schwingung klingt nur sehr langsam ab. Damit stellt die Größe  $W_{kin} = 3.48 \text{ Nmm}$  einen quantitativen Wert zur Beurteilung der Stabilität der untersuchten Gleichgewichtslage dar, bzw. der reziproke Wert ein Maß für die Sensitivität der Gleichgewichtslage.

### 7.6.2 Symmetrisch instabiles Systemverhalten

Ein System mit symmetrisch instabilem Verzweungsverhalten (vergleiche z.B. Abbildung 7.2<sub>2</sub>) zeigt Abbildung 7.14. Der abgebildete Balken ist am einen Ende frei drehbar gelagert und am anderen Ende durch linear elastische Federn abgestützt. Zur Finiten Ele-

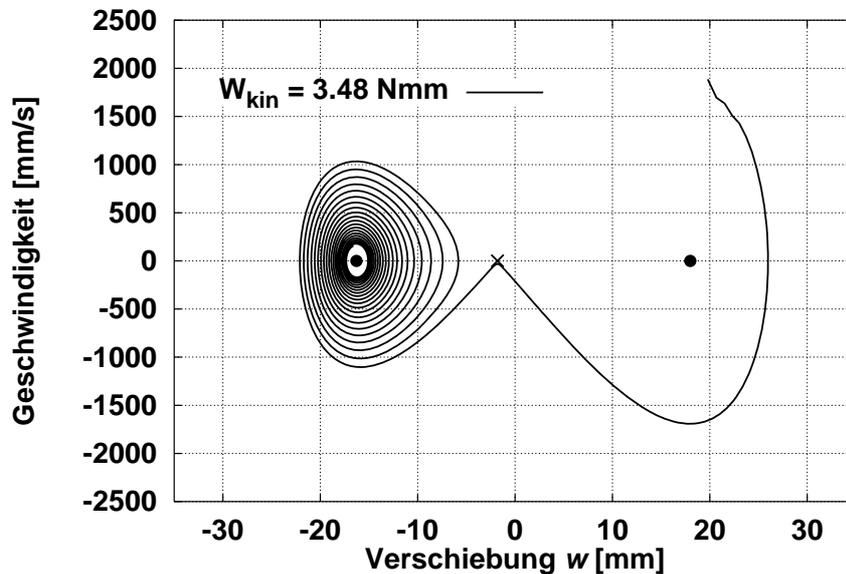


Abbildung 7.13: Imperfekter Kragarm – Phasendiagramm bei  $\lambda = 0.0465$ , Sensitivität

ment Berechnung wurde die elastische Abstützung durch ein drehbar verbundenes Schalelement mit dem Elastizitätsmodul  $E_2 = 1.0 \cdot 10^1 \text{ N/mm}^2$  realisiert. Um, vergleichbar zu dem in [139] behandelten System, einen starren Balken zu realisieren, wurde der Elastizitätsmodul mit  $E_1 = 2.1 \cdot 10^7 \text{ N/mm}^2$  sehr hoch gewählt. Die Last-Verformungskurve der Verschiebung  $w$  des Lastangriffspunktes senkrecht zur Lastrichtung ist in Abbildung 7.15 für die imperfekte Struktur dargestellt. Als Imperfektion dient wie zuvor eine Schrägstellung  $e$  des Balkens. Die stabilen Lösungsbereiche sind mit einer durchgezogenen, die instabilen mit einer gestrichelten Linie gekennzeichnet.

Anhand dieses Systems soll zuerst die Vorgehensweise zum Auffinden benachbarter Gleich-

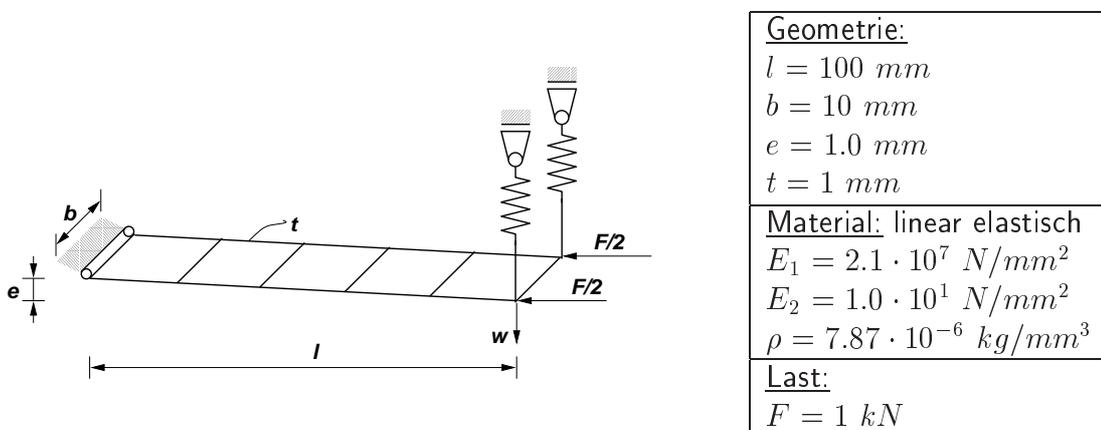


Abbildung 7.14: Geometrie und Materialdaten des elastisch abgestützten, imperfekten Balkens

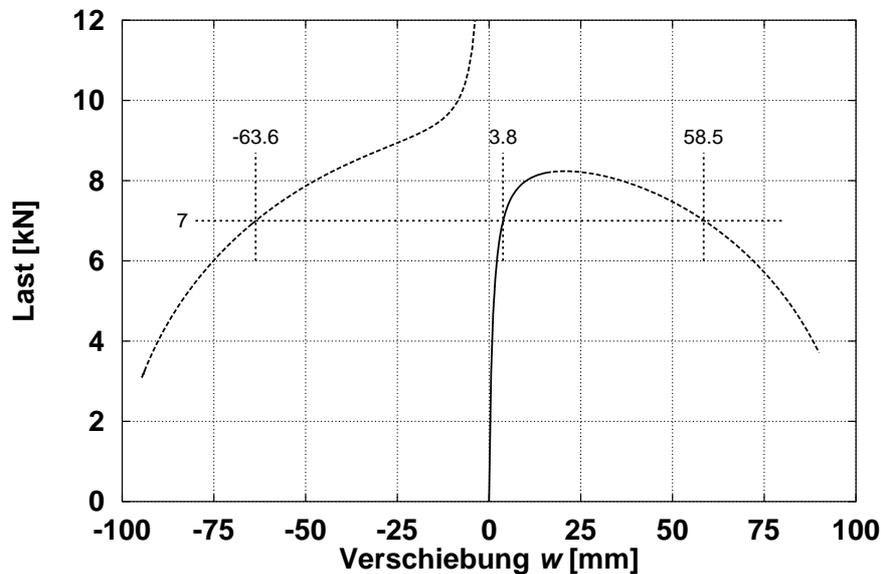


Abbildung 7.15: Elastisch abgestützter, imperfekter Balken – Last-Verformungskurve

gewichtslagen auf gleichem Lastniveau wiederholt werden. Aufgrund des geänderten Systemverhaltens wird eine Anpassung der Definition des Begriffes der Sensitivität erforderlich. Wie aus Abbildung 7.15 ersichtlich, existiert z.B. für das Lastniveau bei  $7 \text{ kN}$  neben der stabilen Gleichgewichtslage auf dem natürlichen Lösungspfad keine weitere stabile Gleichgewichtslage. Eine Definition der Sensitivität als reziproker Wert der kinetischen Energie, die mindestens benötigt wird, um eine zweite, stabile Gleichgewichtslage zu erreichen, greift hier nicht. Daher wird bei diesem Systemverhalten unter Sensitivität der reziproke Wert der kinetischen Energie verstanden, die mindestens benötigt wird, um ein unbegrenztes Anwachsen der Verschiebungen zu erreichen. Unter unbegrenzt anwachsenden Verschiebungen sind bei dem untersuchten elastisch abgestützten Balken vollständige Umdrehungen um die frei drehbare Lagerung zu verstehen, bzw. Verschiebungen des Lastangriffspunktes in Lastrichtung, die der Balkenlänge entsprechen. Alle gezeigten Ergebnisse sind auf den letzteren Bereich beschränkt.

Aufgrund der Resultate aus dem vorher betrachteten fest eingespannten Kragarm wird hier nur die Anfangsgeschwindigkeitsverteilung in der ersten Schwingungseigenform untersucht. Bei dem gewählten Lastniveau von  $7 \text{ kN}$  wird dem mechanischen System kinetische Energie  $W_{kin}$  eingeprägt. Für  $W_{kin} = 28.7 \text{ Nm}$  sind in Abbildung 7.16 zwei Phasendiagramme dargestellt. Bei Kurve (b) wurde die Anfangsgeschwindigkeitsverteilung gegenüber Kurve (a) mit dem Faktor  $-1$  multipliziert, d.h. die Richtung umgekehrt. Die Symbole  $\bullet$  und  $\times$  kennzeichnen erneut stabile und instabile Gleichgewichtslagen. In Kurve (a) wird der Kraftangriffspunkt in positive Richtung  $w$ , bei Kurve (b) dagegen in negative Richtung  $w$  bewegt. Ein unbeschränktes Anwachsen der Verschiebungen wird in beiden Fällen erreicht, der Balken schlägt bei (a) nach unten, bei (b) nach oben durch. Die zu beiden Bewegungen zugehörige statische Residuumsnorm zeigt Abbildung 7.17. Minima der beiden Kurven (a) und (b) bei ca.  $w = -63 \text{ mm}$  und  $w = 58 \text{ mm}$  lassen wieder statische

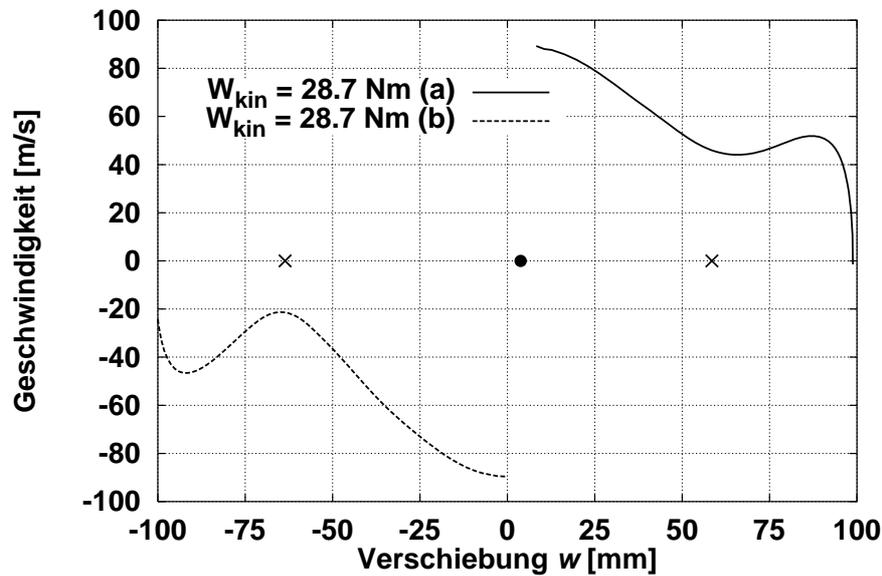


Abbildung 7.16: Elastisch abgestützter, imperfekter Balken – Phasendiagramm bei  $7 \text{ kN}$ ,  $W_{kin} = 28.7 \text{ Nm}$

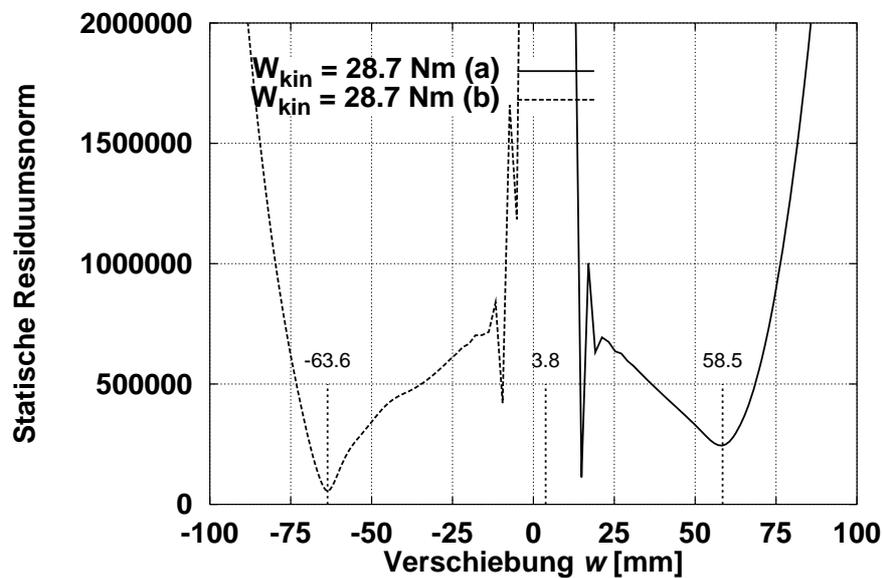


Abbildung 7.17: Elastisch abgestützter, imperfekter Kragarm – statische Residuumsnorm bei  $7 \text{ kN}$ ,  $W_{kin} = 28.7 \text{ Nm}$

Gleichgewichtslagen vermuten. Mittels der Einführung von Dämpfung sind diese Gleichgewichtslagen nicht ansteuerbar, beide sind instabil. Aufgrund der statischen Pfadverfolgung des natürlichen Lösungsastes ist die instabile Gleichgewichtslage bei  $w = 58.5 \text{ mm}$  bereits bekannt. Die zunächst nur vermutete Gleichgewichtslage bei  $w = -63 \text{ mm}$  kann durch Umschalten von der transienten auf eine statische Berechnung in unmittelbarer Nähe bestimmt werden. Somit ist eine Gleichgewichtslage bei  $w = -63.6 \text{ mm}$  an dem unzu-

sammenhängenden Lösungsast bekannt und dieser kann mittels weiterer statischer Pfadverfolgung vollständig bestimmt werden. Auch die Stabilitätsfrage kann dann entschieden werden.

Das Aufbringen einer Anfangsgeschwindigkeit mit positivem und negativem Vorzeichen ist notwendig, da bei Überschreiten des Einzugsbereiches der stabilen Ausgangsgleichgewichtslage das System nicht mehr zurückschwingt. Gleichgewichtslagen in der anderen Richtung können daher nicht gefunden werden. Auf die Bestimmung der Sensitivität hat dies jedoch keine Auswirkung. Dazu wird die Sensitivität beim Lastniveau von  $7\text{ kN}$  bestimmt und die Anfangsgeschwindigkeit jeweils in beide Richtungen aufgebracht. Für  $W_{kin} = 15.4\text{ kNm}$  sind die beiden zugehörigen Phasendiagramme in Abbildung 7.18 dargestellt. Der Durchgang der Phasenkurve (b) zwischen der instabilen und der stabilen Gleichgewichtslage ist eine Folge der Nichtlinearität des Problems, siehe z.B. [99]. In beiden Fällen führt die eingebrachte kinetische Energie zu einem unbeschränkten Anwachsen der Verschiebung in Lastrichtung. Als Folge der bei den Berechnungen eingesetzten numerischen Dämpfung wäre bei Kurve (a) eine geringfügig kleinere Energie ausreichend. Der Unterschied ist mit ca. 3% vernachlässigbar klein. Erst bei einem längerem Bewegungsvorgang vor Erreichen und Überschreiten der instabilen Gleichgewichtslage und damit zu großem Energieverlust durch Dämpfung müßte dies berücksichtigt werden. Wenn ausschließlich die Sensitivität bestimmt werden soll, ist ein vollständiger Verzicht auf Dämpfung möglich. Diese bezweckt ein Herausfiltern höherer Frequenzen um insbesondere die Kurve der statischen Residuumsnorm zum Auffinden weiterer Gleichgewichtslagen zu glätten.

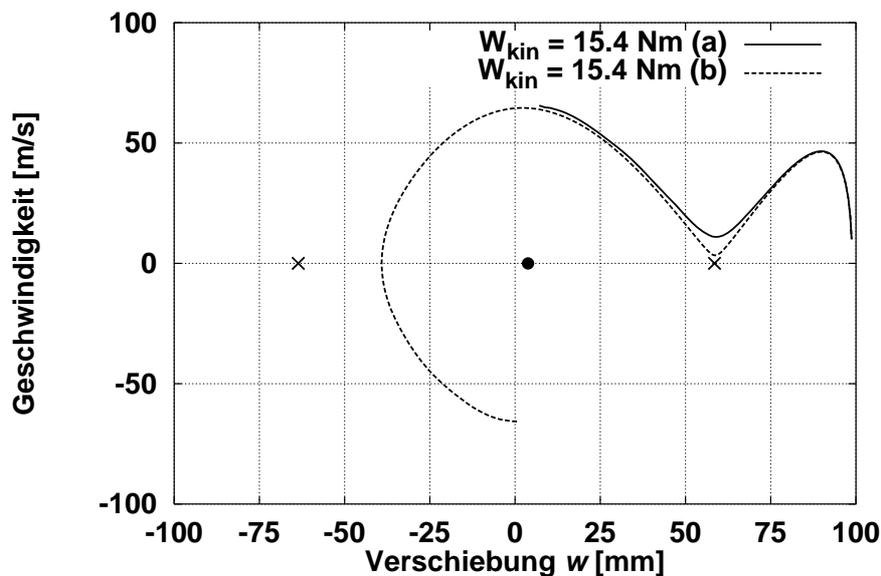


Abbildung 7.18: Elastisch abgestützter, imperfekter Kragarm – Phasendiagramm bei  $7\text{ kN}$ ,  $W_{kin} = 15.4\text{ Nm}$

### 7.6.3 Durchschlagproblem

Als weiteres Beispiel soll die in Abschnitt 3.2 eingeführte flache Zylinderschale betrachtet werden. Dieses zusätzliche Beispiel soll zum Einen zeigen, daß Durchschlagprobleme analog wie die vorher untersuchten Probleme mit unzusammenhängenden Lösungsästen behandelt werden können. Zum Anderen soll aber auch gezeigt werden, daß bei Systemen mit einer höheren Anzahl von Freiheitsgraden die Ergebnisse schwieriger zu interpretieren sind. Weiterhin soll untersucht werden, inwieweit auch die Eigenform  $\phi_1$  aus dem linearen Eigenwertproblem (7.22) als Anfangsgeschwindigkeitsverteilung sinnvoll einzusetzen ist, da diese Eigenform bei der Berechnung von benachbarten Gleichgewichtslagen in [31] eine wesentliche Rolle spielt.

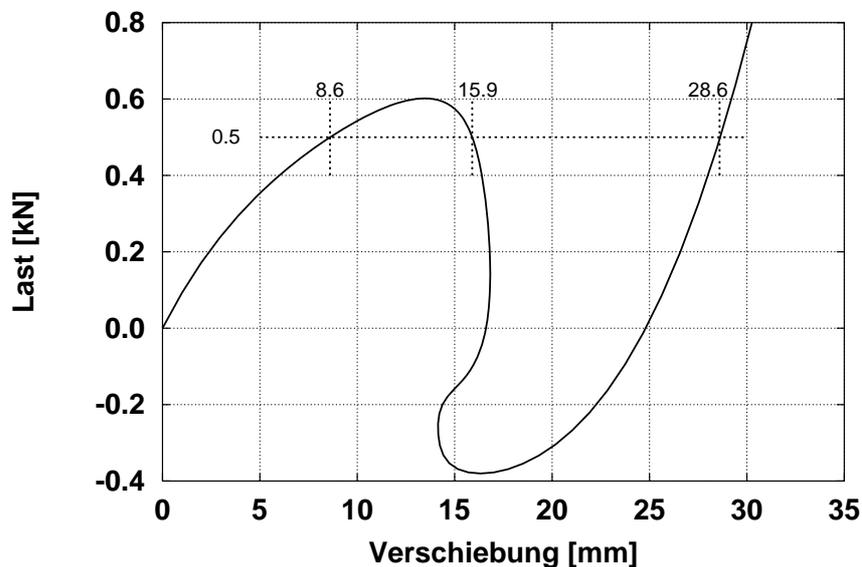


Abbildung 7.19: Flache Zylinderschale – Last-Verformungskurve

In Abbildung 7.19 ist die mittels Pfadverfolgungsalgorithmen ermittelte Last-Verformungskurve dargestellt. Für die Last-Verformungskurve wie auch für die nachfolgend gezeigten Phasenportraits wurde der Lastangriffspunkt als charakteristische Koordinate ausgewählt. Es soll die Gleichgewichtslage bei der Last  $0.5 \text{ kN}$  mit der Verschiebung  $8.6 \text{ mm}$  auf ihr Verhalten gegenüber den Anfangsgeschwindigkeiten  $v_0 = \varphi_1$  und  $v_0 = \phi_1$  untersucht werden. Dazu wird wiederum eine Geschwindigkeitsverteilung in der ersten Schwingungseigenform  $\varphi_1$  mit der kinetischen Energie  $W_{kin} = 34.1 \text{ Nm}$  aufgebracht. Die zugehörige Phasenkurve zeigt Abbildung 7.20, stabile Gleichgewichtslagen sind darin erneut durch  $\bullet$ , die instabile durch  $\times$  gekennzeichnet. Die Norm des dazugehörigen statischen Residuums ist in Abbildung 7.21 dargestellt.

Auch hier kann über das statische Residuum eine Aussage über die Nähe zu den drei Gleichgewichtslagen getroffen werden. Die instabile Gleichgewichtslage ist aber nicht so deutlich ausgeprägt. Da die Steifigkeit der Struktur nichtlinear von den Verschiebungen abhängt, ist auch bei der Vorgabe von  $v_0 = \varphi_1$  als Anfangsgeschwindigkeitsverteilung keine

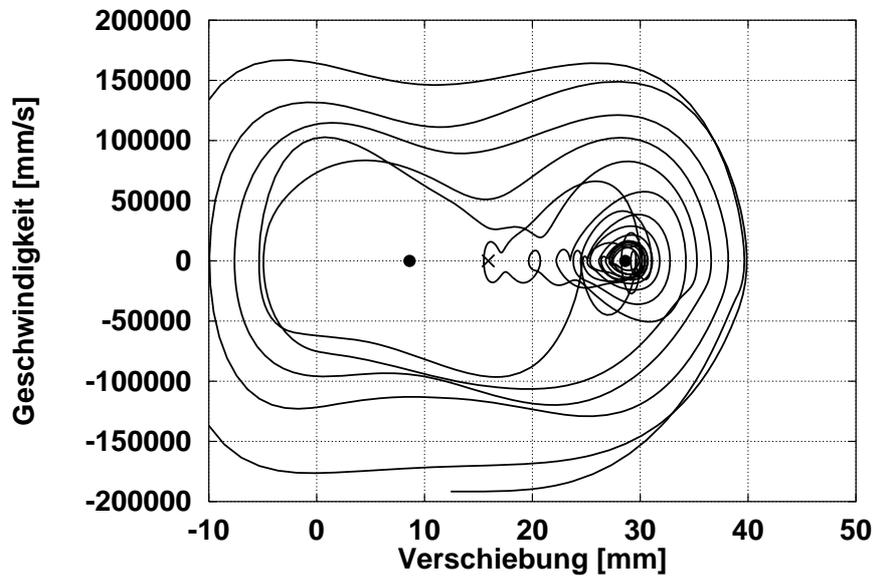


Abbildung 7.20: Flache Zylinderschale – Phasendiagramm (1. Eigenform  $\varphi_1$ ),  $W_{kin} = 34.1 Nm$

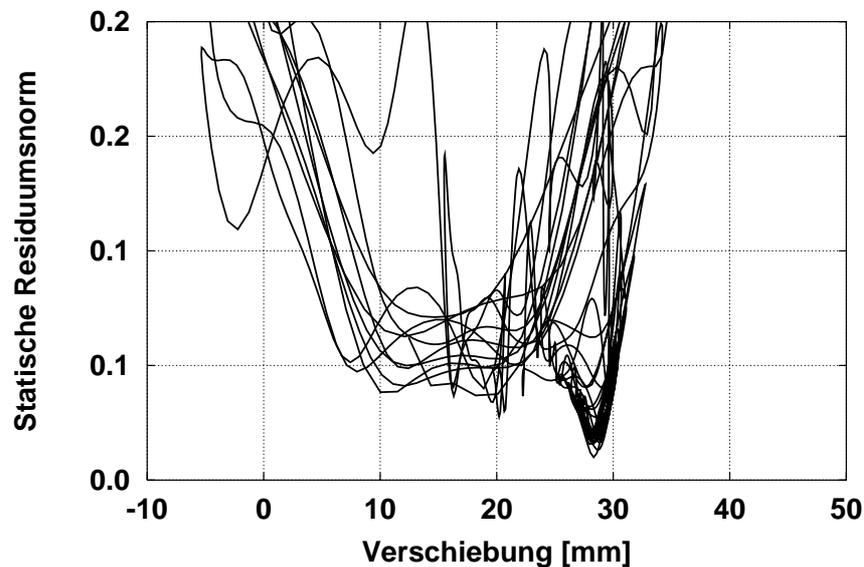


Abbildung 7.21: Flache Zylinderschale – statisches Residuum (1. Eigenform  $\varphi_1$ ),  $W_{kin} = 34.1 Nm$

reine Schwingung in der ersten Schwingungseigenform zu beobachten, zumal die Auslenkungen groß sind, wie ein Vergleich mit den statischen Verschiebungen in Abbildung 7.19 zeigt.

In den Abbildungen 7.22 und 7.23 sind das Phasendiagramm und die Residuumsnorm für eine Anfangsgeschwindigkeitsverteilung in der ersten Eigenform  $\phi_1$  des linearen Ei-

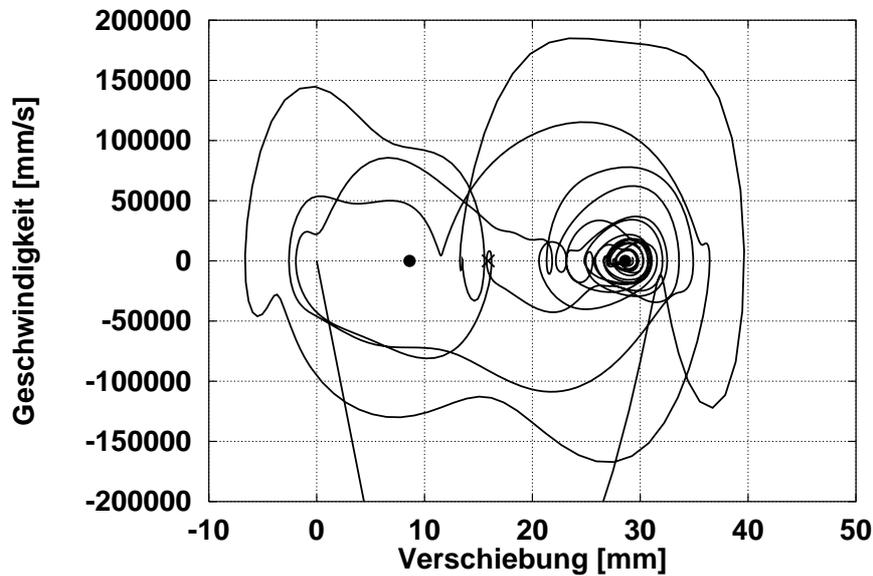


Abbildung 7.22: Flache Zylinderschale – Phasendiagramm (1. Eigenform  $\phi_1$ ),  $W_{kin} = 34.1 Nm$

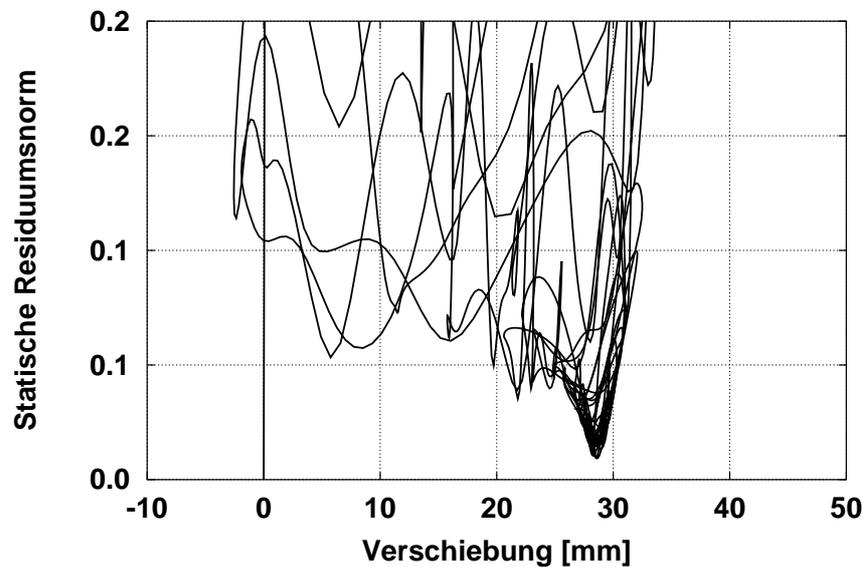


Abbildung 7.23: Flache Zylinderschale – statisches Residuum (1. Eigenform  $\phi_1$ ),  $W_{kin} = 34.1 Nm$

genwertproblem aus (7.22) angegeben. Die kinetische Energie der Anfangsgeschwindigkeit beträgt ebenfalls  $W_{kin} = 34.1 Nm$ . Insbesondere im Phasendiagramm in Abbildung 7.22 sind deutlich höherfrequente Schwingungsanteile als bei der Störung in der ersten Schwingungseigenform  $\varphi_1$  zu erkennen. Dies wirkt sich dann entsprechend auf das zugehörige statische Residuum in Abbildung 7.23 aus.

Es ist daher vorzuziehen, die Störung in Form der ersten Schwingungseigenform  $\varphi_1$  aufzubringen, da dies auch die vom System bevorzugte Schwingungsform darstellt. In dieser Eigenform kann das System mit dem geringsten Energieaufwand zu Schwingungen angeregt werden. Dementsprechend ist hier auch der geringste zusätzliche Energieaufwand zum Erreichen eines möglichen Durchschlagens der Struktur in eine andere stabile Gleichgewichtslage oder auch zu einer unbegrenzten Bewegung zu erwarten.

## 7.7 Zusammenfassung und Diskussion

Bei statischen Stabilitätsproblemen sind mit der Methode der Finiten Elemente prinzipiell alle Lösungsäste ermittelbar [28, 38]. Dabei kann die Stabilitätsfrage der ermittelten Gleichgewichtslage mittels der Definitheit der tangentiellen Steifigkeitsmatrix in effizienter Weise entschieden werden. Bei Verzweigungsproblemen existieren sogenannte Pfadwechselprozeduren, die es ermöglichen, auch verzweigende Lösungspfade zu berechnen. Der Aufwand bei diesem Vorgehen kann allerdings enorm sein, für große Probleme ist er oft zu hoch.

Bei transienten Betrachtungen beruht die Stabilitätsaussage entweder auf einem linearisierten Kriterium, das keine Aussage über die Bewegung als Ganzes liefert, oder das Kriterium ist aufgrund des Aufwandes für reale Problemstellungen ungeeignet. Bei Verwendung des Newmark Verfahrens zur Integration der Bewegungsgleichungen ist das linearisierte Kriterium nur mit beträchtlichem zusätzlichem Aufwand auswertbar. Dennoch liefert diese Aussage einen Hinweis für den Zustand des Systems an der entsprechenden Laststufe und bei einer quasistatischen Bewegung, z.B. im Vorbeulbereich, kann sogar die statische Stabilitätsaussage erhalten werden.

Unzusammenhängende Lösungsäste können aufgrund der lokalen Konvergenzeigenschaften des Newton Verfahrens meist nicht durch rein statische Methoden erreicht werden. Zwar können zusätzliche statische Lasten in das Modell eingebracht und nach Erreichen des unzusammenhängenden Lösungsastes wieder entfernt werden. Dieses Vorgehen bedingt aber ein Kenntnis des Systemverhaltens und kann nicht immer vorausgesetzt werden. Daher empfehlen sich in solchen Fällen transiente Berechnungsmethoden. So kann z.B. durch Aufbringen einer endlichen Anfangsgeschwindigkeit eine Struktur in einer beliebigen Gleichgewichtslage zu Schwingungen angeregt werden. Durch die Beobachtung des Schwingungsvorganges selbst und des statischen Residuums kann auf die Existenz von weiteren benachbarten Gleichgewichtslagen geschlossen werden, die auch numerisch angesteuert werden können, was jedoch nur bei stabilen Gleichgewichtslagen zuverlässig gelingt. Die Beobachtung des Schwingungsvorganges selbst wird aber insbesondere bei größeren und daher nicht einfach übersehbaren Strukturen schwierig, da nicht immer eine charakteristische Verschiebung angebbar ist. Hier ist die statische Residuumsnorm als skalare Größe von Vorteil. Allerdings erweist sich diese Norm nur dann als zuverlässiges Kriterium für statische Gleichgewichtslagen, wenn die Schwingung bereits im Einzugsbereich einer statischen Gleichgewichtslage abläuft. Führt die aufgebrachte Anfangsgeschwindigkeit nicht zu einer Schwingung, sondern wachsen die Verschiebungen unbegrenzt an, so versagt die Struktur. Der Vorgang kann mit einer kleineren eingepägten

kinetischen Energie wiederholt werden, um entweder eine eventuell vorhandene weitere Gleichgewichtslage aufzuspüren und anzusteuern, oder die Grenze zwischen einer Bewegung um die Gleichgewichtslage und dem unbegrenzten Anwachsen der Verschiebungen zu ermitteln.

Letzteres führt zum Begriff der Sensitivität: die Struktur reagiert am empfindlichsten gegenüber einer Störung, die bei minimaler zugeführter kinetischer Energie zum Versagen bzw. zum Übergang in eine weitere Gleichgewichtslage bei gleichem Lastniveau führt. Meist sollte die Form der Anfangsgeschwindigkeitsverteilung, die zu minimalen Energien führt (insbesondere in der Nähe singulärer Punkte), die erste Schwingungseigenform sein. Weitere Eigenformen sowie technisch realistische Formen sollten aber ebenfalls Berücksichtigung finden, da hiermit tatsächlich auftretende Situationen abgesichert werden und um zu überprüfen, ob nicht andere Anfangsgeschwindigkeitsverteilungen neben der ersten Schwingungseigenform zu kleineren Energien führen.

Der hier verwendete Begriff der Sensitivität einer Gleichgewichtslage gegenüber endlichen Störungen ist streng abzugrenzen von der Empfindlichkeit einer Struktur gegenüber Systemparametern wie geometrischen Kenngrößen, geometrischen und Lastimperfektionen und Materialparametern [39, 79, 103].

Die hier vorgeschlagene systematische Vorgehensweise zur Untersuchung der Sensitivität einer stabilen Gleichgewichtslage kommt zwar ohne Eigenwertberechnungen aus (zur Berechnung der ersten Schwingungseigenform genügt eine grobe Näherung, die z.B. über eine von Mises Vektoriteration erfolgen kann), doch erfordert die Zeitintegration einen hohen Rechenaufwand. Für Letzteres ist der Einsatz paralleler Methoden von großem Vorteil. Wegen der verbesserten Kondition der effektiven Steifigkeitsmatrix  $\mathbf{K}^*$  in (2.82), die durch die Addition eines Vielfachen der positiv definiten Massenmatrix zur tangentiellen Steifigkeitsmatrix entsteht, eignet sich das einfach zu parallelisierende cg-Verfahren hervorragend zu Lösung solcher Probleme.

## 8 Untersuchung eines Stahlzylinders mit imperfekter Geometrie

In diesem Abschnitt werden die in Kapitel 7 vorgestellten Vorgehensweisen an einem größeren, komplexeren Problem durchgeführt. Für eine mechanische Struktur erfolgen sowohl statische als auch transiente Untersuchungen. Dabei werden die jeweiligen Stabilitätskriterien ausgewertet. Auch soll die Imperfektionsempfindlichkeit gegenüber geometrischen Imperfektionen und die Sensitivität, d.h. die Empfindlichkeit gegenüber endlichen Störungen an ausgewählten Gleichgewichtslagen betrachtet werden.

Wesentlich dabei ist, daß die gewählte mechanische Struktur eines axial belasteten Stahlzylinders nicht eine perfekt symmetrische, sondern eine imperfekte Geometrie aufweist, die aber dennoch kein “gutmütiges”, d.h. einfaches nichtlineares Verhalten aufweist.

### 8.1 Problembeschreibung

Silos sind Großraumbehälter zur Lagerung von Gütern, die sowohl fest, flüssig oder auch gasförmig sein können. Auf kleiner Grundfläche, die üblicherweise rund oder quadratisch ist, können darin große Mengen gelagert werden. Allerdings sind diese Bauwerke bei Ausführung als dünnwandige Stahlbehälter oft stark beulgefährdet.

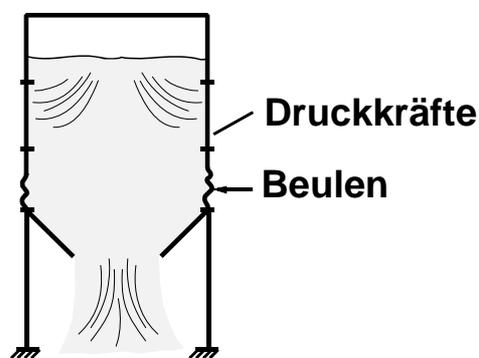


Abbildung 8.1: Schüttgutgefülltes Stahlsilo

Abbildung 8.1 zeigt ein solches Silo während des Entleerungsvorganges. Es treten in Längsrichtung Druckspannungen aufgrund des Gewichtes des oberen Teiles des Silos und aufgrund der Wandreibung zwischen Schüttgut und Siloschale auf. Außerdem entstehen Membranspannungen durch das Schüttgut.

Bei der Bemessung der Wandstärke eines Silos ist in aller Regel der Beulsicherheitsnachweis für die Axialbelastung maßgebend. In Regelwerken wie DIN 18800 Teil 4 [1] oder ECCS [34] beruht der zu führende Nachweis auf der Bestimmung einer idealen, klassischen Beulspannung der ungefüllten Zylinderschale. Über Abminderungsfaktoren, die üblicherweise auf der statistischen Auswertung einer großen Anzahl von Beulversuchen basieren, wird der Abfall der Beullast durch geometrische Imperfektionseinflüsse berücksichtigt. Der Einfluß der Bettung durch das Schüttgut und die damit verbundene Steigerung

der Traglast darf aber nicht berücksichtigt werden. In einem Sonderforschungsbereich der Deutschen Forschungsgesellschaft (DFG) [81, 80, 142] wurde das Stabilitätsverhalten von Siloschüssen unter Berücksichtigung des Schüttguteinflusses experimentell und numerisch untersucht. Dazu wurden unter anderem leere und wassergefüllte Kreiszylinderschalen in einer Prüfapparatur axial belastet und das Last-Verformungsverhalten gemessen. Die imperfekte Geometrie des Prüfkörpers wurde zuvor bezüglich radialer Abweichung vom Sollradius hochgenau vermessen, sodaß diese zu den nachfolgenden numerischen Untersuchungen verfügbar war. Auch die Materialdaten wurden für jeden Zylinder gesondert bestimmt.

Ein solcher Zylinder mit imperfekter Geometrie soll im Folgenden exemplarisch bezüglich Stabilität und Sensitivität untersucht werden.

## 8.2 Modellierung mit Finiten Elementen

In Abbildung 8.2 sind das System und die Materialdaten des untersuchten Siloschusses dargestellt. Dieser entspricht dem unteren, beulgefährdeten Teil des in Abbildung 8.1 dargestellten Silos. Der Zylinder ist an seiner Unterseite konstruktionsbedingt unverschieblich aber frei drehbar gelagert. An seiner Oberseite sind nur Verschiebungen in Zylinderlängsachsenrichtung zugelassen. Die Unverschieblichkeit in Umfangsrichtung am oberen Ende ist durch die bei den Versuchen angebrachten Flansche [142] begründet. Das Verhältnis des Radius  $r$  zur Schalendicke  $t$  beträgt ca. 1100, daher die Versuchsbezeichnung "AL-1100", wobei "A" für den ersten Teil der Versuchsserie und "L" für "leer" steht. Bei der Auswahl des Blechs für die Versuchszylinder wurde ein Material mit hoher Streckgrenze gewählt, um plastische Einflüsse zu vermeiden. Daher wurde für die numerische Berechnung ausschließlich linear elastisches Materialverhalten angenommen.

Der Zylinder wurde mit einem annähernd quadratischen Raster vermessen, für den gewählten Zylinder ergeben sich bei direkter Übernahme der Meßpunkte in das FE-Netz 9400

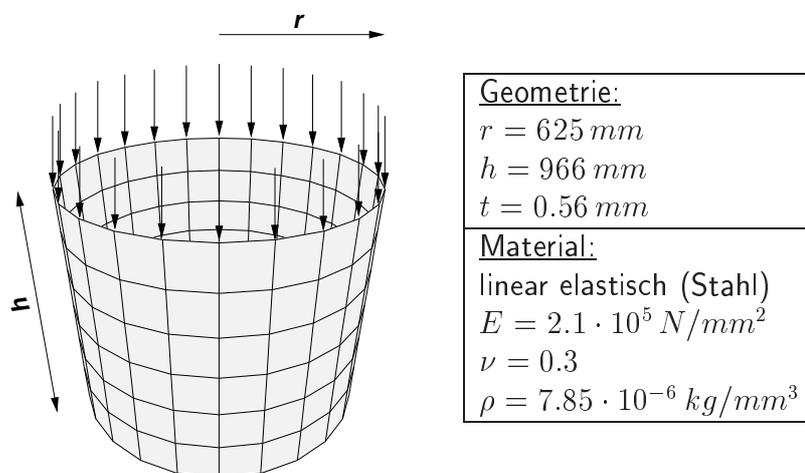


Abbildung 8.2: Imperfekter Zylinder AL-1100 – System

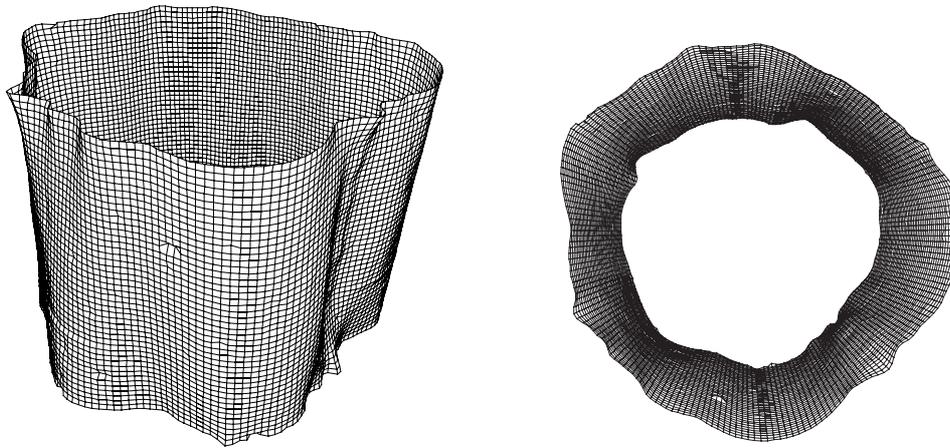


Abbildung 8.3: Imperfekte Geometrie, 50-fach überhöht dargestellt

Knotenpunkte und 9200 Elemente. Das zugehörige Finite Elemente Netz ist in Abbildung 8.3 gezeigt, wobei die Imperfektionen, d.h. die Abweichung von der perfekten Zylindergeometrie, mit dem Faktor 50 überhöht dargestellt sind. Auffällig an den Imperfektionen ist insbesondere eine deutlich ausgeprägte “Delle” in der linken Hälfte auf ca.  $2/3$  der Zylinderhöhe, siehe Abbildung 8.3 links. Weiterhin ist, deutlicher sichtbar in Abbildung 8.3 rechts vorn, eine sich über die Zylinderhöhe erstreckende Abplattung erkennbar. Die maximale Abweichung von der perfekten Geometrie beträgt  $1.52\text{ mm}$ , ca. das 2.7-fache der Schalendicke.

Zur Abbildung der Belastung werden die Freiheitsgrade der Knoten am oberen Zylindertrand in Zylinderlängsachsenrichtung über eine Zwangsbedingung zu einem einzigen Freiheitsgrad zusammengefaßt. Damit wird eine vollständig starre und ebene Lasteinleitung wie in der Prüfapparatur erreicht.

### 8.3 Beschreibung der Versuchsergebnisse

Die im Folgenden kurz wiedergegebenen Versuchsergebnisse für den im Rahmen der vorliegenden Arbeit numerisch untersuchten Versuchszylinder entstammen [142].

Für den Beulversuch wurde der Testzylinder in die Prüfapparatur zwischen der oberen und unteren Druckplatte eingespannt, siehe Abbildung 8.4. Die Lasteinleitung erfolgte über eine Lasteinleitungskugel aus gehärtetem Stahl, um die Entstehung von Biegemomenten durch den Hydraulikkolben zu vermeiden. Das Last-Verformungsverhalten wurde mittels einer Kraftmeßdose und vier induktiven Wegaufnehmern an den Viertelpunkten des Zylinderumfangs gemessen. Die Last wurde langsam bis zum Versagen aufgebracht. Nach jedem inkrementellen Lastschritt wurde das jeweilige Lastniveau konstant gehalten, um die Meßwerte abzulesen. Das Versagen des Zylinders setzte schlagartig ein und wurde von einem lauten Knall begleitet. Ein vorheriges, lokales Beulen konnte nicht beobachtet werden. Bedingt durch das hohe Gewicht des oberen Versuchsaufbaus traten große Verformungen auf. Das beobachtete Nachbeulmuster war rautenförmig, 2-3 reihig und befand

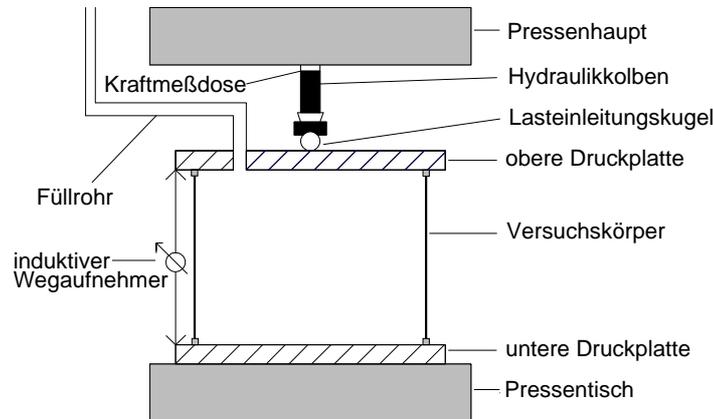


Abbildung 8.4: Versuchsaufbau

sich zwischen dem unteren Zylinderrand und der Zylindermitte. In Umfangsrichtung befanden sich ca. 16 Beulen. Die Last, bei der das schlagartige Versagen des Zylinders eintritt, wird im Folgenden als Beullast bezeichnet. Für den Zylinder AL-1100 betrug sie  $135\text{ kN}$ . Nach dem Versagen wurde der Zylinder erneut belastet, um die sogenannte Nachbeullast zu bestimmen, sie wurde für den untersuchten Zylinder zu  $48\text{ kN}$  bestimmt.

## 8.4 Imperfektionsempfindlichkeit

In Abbildung 8.5 ist die Imperfektionsempfindlichkeit, d.h. die Abhängigkeit der Beullast von der Imperfektionsamplitude des untersuchten Zylinders, dargestellt. Das gemessene Imperfektionsmuster, d.h. die Abweichung der tatsächlichen Geometrie von der idealen Zylindergeometrie wurde mit einem skalaren Parameter  $\varepsilon$  multipliziert und zur idealen Zylindergeometrie addiert. Für  $\varepsilon = 0$  wird so der perfekte Zylinder erhalten, bei  $\varepsilon = 1$  die tatsächlich gemessene Geometrie. Variationen an der Imperfektionsform zur Ermittlung einer ungünstigeren Form wurden nicht vorgenommen, hierzu wird z.B. auf DEML [28] verwiesen. Dort wird die Imperfektionsform so bestimmt, daß bei vorgegebener Maximalabweichung der imperfekten von der perfekten Geometrie eine minimale Beullast erreicht wird. Dazu werden die geometrischen Imperfektionen als zusätzliche Freiheitsgrade eingeführt und mit der Schalenformulierung konsistent linearisiert.

Die Imperfektionsempfindlichkeit ist größer für positive Werte von  $\varepsilon$ , bei  $\varepsilon = 1$  ist die Beullast von  $250\text{ kN}$  auf ca.  $2/3$  dieses Wertes d.h.  $165\text{ kN}$  gesunken. Bei  $\varepsilon = -1$  beträgt sie hingegen noch  $206\text{ kN}$ . Der numerisch ermittelte Wert für den perfekten Zylinder stimmt sehr gut mit dem von LORENZ angegebenen Wert der Beulspannung (siehe z.B. [125, 110])

$$\sigma_{kr} = \frac{E}{\sqrt{3(1-\nu^2)}} \frac{t}{r}, \quad (8.1)$$

die auf eine Beullast von  $253\text{ kN}$  führt, überein.

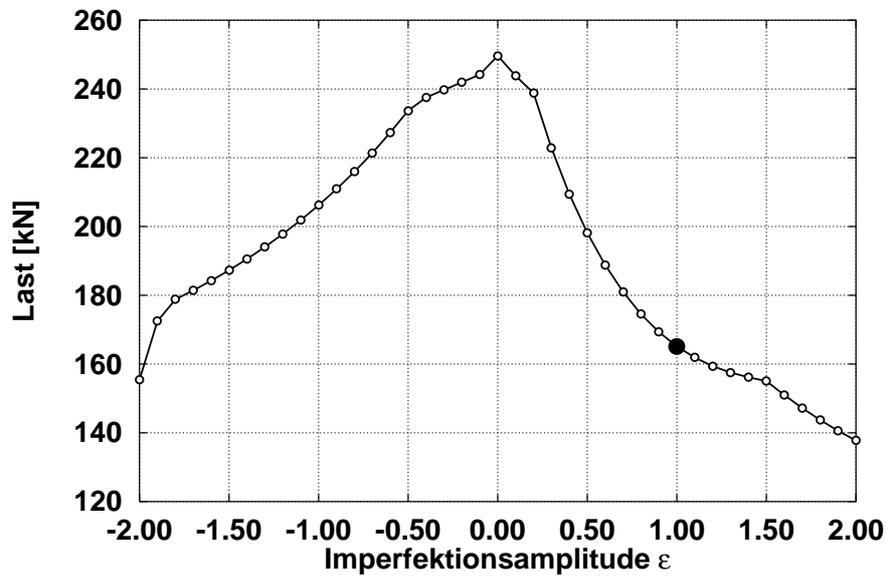


Abbildung 8.5: Imperfektionsempfindlichkeit des Zylinders AL-1100

In Abbildung 8.6 sind die Grenzlasten für die unterschiedlichen Verzweigungstypen, siehe Abbildung 7.1 und 7.2, angegeben. Die Maximallasten sind darin in Abhängigkeit der Imperfektionsgröße  $\varepsilon$  aufgetragen. Die größte Übereinstimmung der Imperfektionsempfindlichkeit des imperfekten Zylinders ergibt sich mit dem symmetrischen instabilen Verzweigungstyp. Auch in [125] wird dieser Typ dem axial belasteten Zylinder zugeordnet. Die Unsymmetrie in Abbildung 8.5 ist durch das allgemeine, unsymmetrische Imperfektionsmuster verursacht, für betragsmäßig gleiche  $\varepsilon$ -Werte ergeben sich unterschiedliche Imperfektionsamplituden nach innen bzw. außen.

Alle weiteren Berechnungen werden mit der tatsächlich gemessenen Imperfektion (d.h.  $\varepsilon = 1$ ) durchgeführt, in Abbildung 8.5 ist dies mit einem ● gekennzeichnet.

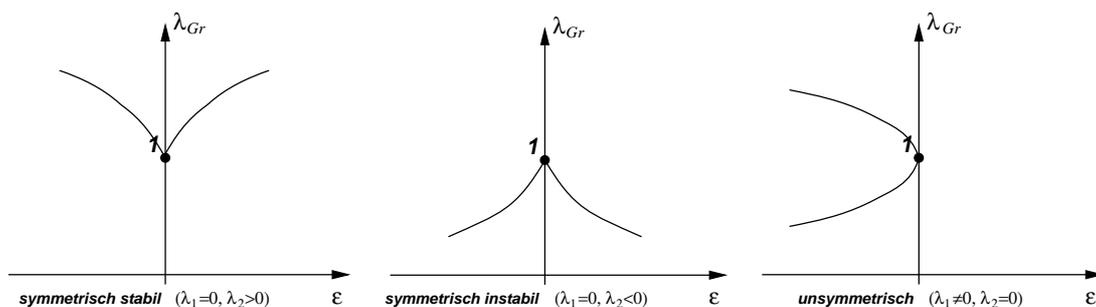


Abbildung 8.6: Grenzlasten der unterschiedlichen Verzweigungstypen

## 8.5 Statische Stabilitätsuntersuchung

Zur Bestimmung der kritischen Last einer mechanischen Struktur erfolgt üblicherweise eine Berechnung des nichtlinearen Last-Verformungsverhaltens. Dabei werden begleitend die Determinante der tangentiellen Steifigkeitsmatrix bzw. deren Trägheit beobachtet. Zeigen diese Kriterien eine singuläre Steifigkeitsmatrix an, so ist ein "Beulpunkt" erreicht. Mit Hilfe von Bisektionsverfahren kann dieser iterativ noch genauer ermittelt werden. Eine direkte Berechnung des Beulpunktes durch das in Abschnitt 7.3.3 Verfahren führt für die hier untersuchte Struktur zu Divergenz [80]. Die numerisch mittels Pfadverfolgung ermittelte Beullast beträgt  $165 \text{ kN}$  (siehe auch Abbildung 8.5) und weicht damit um ca. 22% von der experimentell bestimmten Beullast von  $135 \text{ kN}$  ab. Diese Abweichung ist im Wesentlichen auf im Rechenmodell nicht berücksichtigte zusätzliche Modellabweichungen zurückzuführen. Diese bestehen in eventuell vorhandenen Materialinhomogenitäten und vor allem in Randaufstandsimperfektionen, deren Einfluß in [80] genauer untersucht wurden. Desweiteren ist zu bedenken, daß der experimentell bestimmte Wert auf einen einzigen Versuch zurückgeht und damit sonstige, versuchsbedingte Abweichungen bzw. Streuungen (z.B. Ablesefehler, Abgleichungenauigkeiten usw.) nicht zur Verfügung stehen. Die numerische Abschätzung des Beulpunktes mit Hilfe des linearen Eigenwertproblems nach Gleichung (7.22) ergibt ein kritisches Lastniveau von  $226 \text{ kN}$  und zeigt die Notwendigkeit der Berechnung des nichtlinearen Strukturverhalten in diesem Fall.

Unter Verwendung von Bogenlängenverfahren kann der Last-Verformungspfad auch über den singulären Punkt hinweg weiterverfolgt werden. In Abbildung 8.7 ist die berechnete Last über der Verschiebung des oberen Zylinderrandes aufgetragen, eine Ausschnittsver-

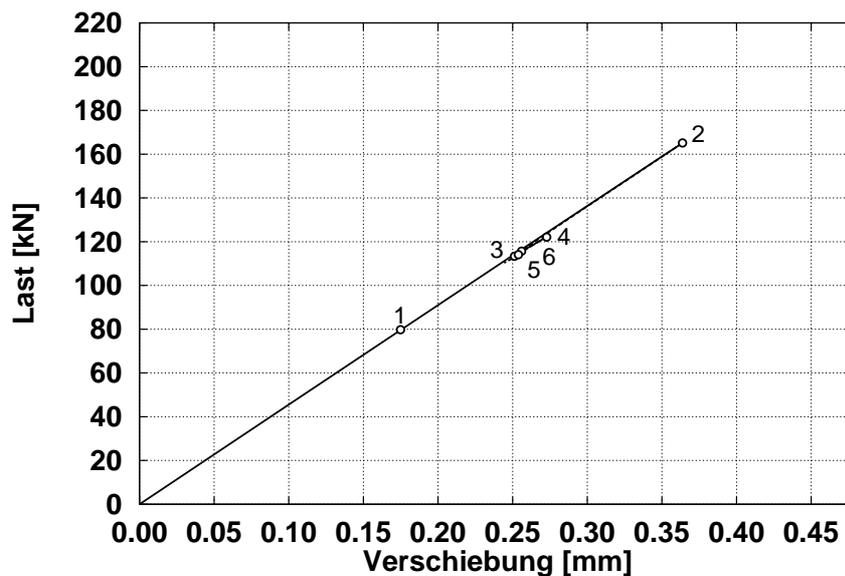


Abbildung 8.7: Last-Verformungskurve, statische Berechnung mit Hilfe von Kurvenverfolgungsalgorithmen

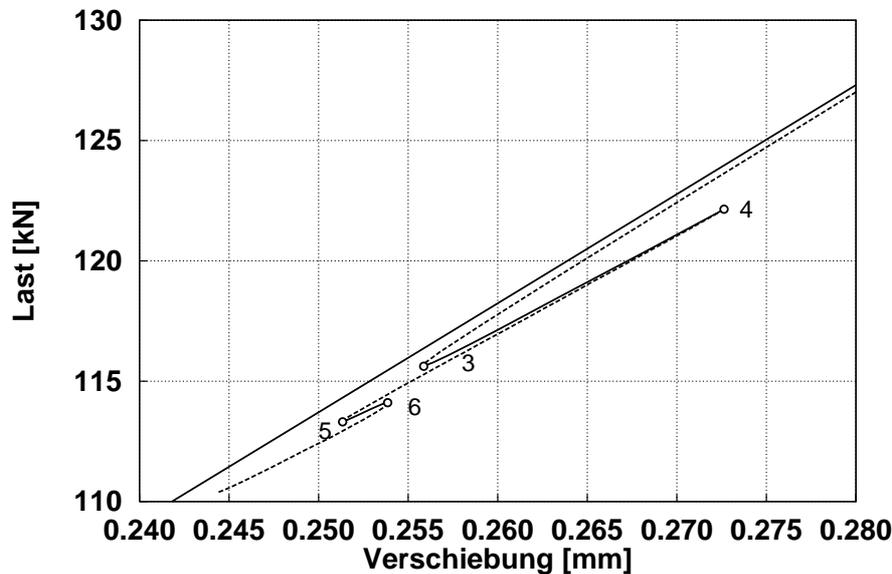


Abbildung 8.8: Last-Verformungskurve, statische Berechnung mit Hilfe von Kurvenverfolgungsalgorithmen – Ausschnittsvergrößerung

größerung zeigt Abbildung 8.8. Nach Erreichen des kritischen Punktes, in Abbildung 8.7 mit “2” bezeichnet, fällt die Last abrupt unter gleichzeitigem Rückgang der Verschiebung ab. In der vergrößerten Darstellung des Nachbeulbereiches in Abbildung 8.8 kann noch ein mehrfaches lokales Ansteigen und Abfallen der Last beobachtet werden. Die abfallenden Bereiche der Kurve sind jeweils instabil. In den Abbildungen 8.7 und 8.8 sind die stabilen Bereiche mit durchgezogenen, die instabilen Bereiche mit gestrichelten Linien gekennzeichnet.<sup>12</sup> Innerhalb einer numerischen Untersuchung muß hier meist eine manuelle Modifikation des Steuerparameters der Berechnung erfolgen. Automatische Steuerungen mit Hilfe des Steifigkeitsparameters aus Abschnitt 7.2.3 oder der Trägheit der Steifigkeitsmatrix versagen hier oft bzw. erfordern dann extremen Aufwand. Desweiteren ist die Konvergenz des Newton-Raphson Verfahrens als Folge der schlechten Kondition der Steifigkeitsmatrizen in diesen Bereichen sehr schlecht. Es muß dann mit sehr kleinen Schrittweiten gearbeitet werden. Daher wird die Berechnung der kompletten Last-Verformungskurve bis hin zur Nachbeullast, wenngleich prinzipiell möglich, mit vertretbarem Aufwand nicht mehr durchführbar, insbesondere wenn man bedenkt, daß im weiteren Verlauf auch Verzweigungspunkte auftreten können. Die sich verzweigenden Lösungsäste müßten ihrerseits wieder verfolgt werden.

Die Deformation des Zylinders ist in Abbildung 8.9 dargestellt, die jeweils zugehörigen Last-Verformungszustände sind in Abbildung 8.7 und 8.8 durch Punkte gekennzeichnet. An den Deformationsfiguren ist an allen Zuständen nur eine einzige Beule zu erkennen, deren Ausprägung variiert. Sie tritt in der Nähe der auffälligen Delle im Ausgangsimper-

<sup>12</sup>Diese Information kann mittels der in Abschnitt 7.2.1 bzw. 7.2.2 angegebenen Kriterien in einfacher Weise ermittelt werden.

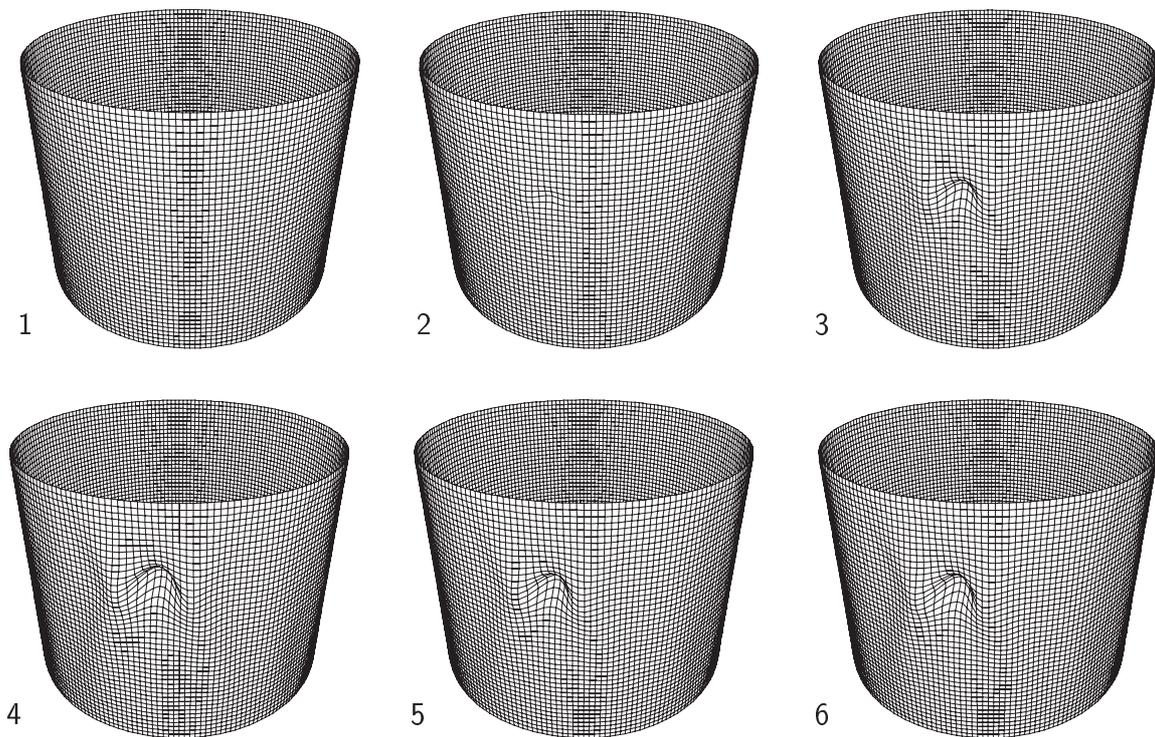


Abbildung 8.9: Verformungsfiguren, statische Belastung, 30-fach überhöht, zugehörige Last-Verformungskurve siehe Abbildung 8.7

fektionsmuster in Abbildung 8.3 auf. Von dieser Stelle scheint die Beulentwicklung auszugehen. Einen Hinweis auf ein sich einstellendes Nachbeulmuster analog zum Versuch kann aus Abbildung 8.9 nicht entnommen werden.

Die mit Abstand schnellste Lösung auf sequentiellen Rechnern wird hierbei mit der in Abschnitt 4.4.3 beschriebenen Kombination aus iterativen und direktem Verfahren (SQMR-LU) erzielt, wobei festzustellen ist, daß der verfügbare Speicherplatz für die vollständige Abspeicherung im Hauptspeicher ausreichte. Rein iterative Lösungsverfahren sind hier in Bezug auf Rechenzeit sehr unvorteilhaft, da die Steifigkeitsmatrix über weite Bereiche sehr schlecht konditioniert ist. Da bei der parallelen Berechnung iterative Lösungsverfahren eingesetzt werden, können nur sehr schlechte *speedup* Zahlen erzielt werden, wenn ein Vergleich zum besten sequentiellen Verfahren (hier SQMR-LU) zugrunde gelegt wird [123].

Die hier exemplarisch dargestellte Last-Verformungskurve hat für andere Zylinder eine ähnliche Gestalt.

## 8.6 Transiente Belastung

Im Gegensatz zur statischen Analyse kann die axiale Lasteinleitung für transiente Analysen verschiedenartig erfolgen. Zur Simulation des weggesteuerten Beulversuches wird eine

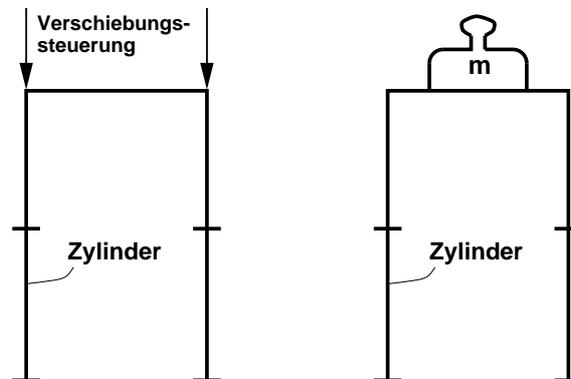


Abbildung 8.10: Modellierung der Axialbelastung für transiente Untersuchungen

verschiebungs- bzw. geschwindigkeitsgesteuerte Lasteinleitung verwendet.

Wird aber das ursprüngliche Problem des Silobeulens in Abbildung 8.1 genau betrachtet, wird deutlich, daß zumindest ein Teil der Axiallast durch das Eigengewicht des Oberbaus verursacht wird. Mit diesem Gewicht gehen bei transienter Betrachtung als Folge der Beschleunigung der Masse des Oberbaus beim Beulvorgang Trägheitskräfte einher, die nicht vernachlässigt werden können. Die Axiallast kann also auch als Gewichtsbelastung des Silos betrachtet werden. In Abbildung 8.10 sind beide Modelle der Axialbelastung, die in den folgenden Abschnitten diskutiert werden, skizziert.

### 8.6.1 Verschiebungsgesteuerter Prozess

Um das Experiment des Zylinderbeulens in der Prüfmaschine zu simulieren, bei dem die Belastung zwar quasistatisch aufgebracht wird, der Beulvorgang dann jedoch schlagartig einsetzt und somit dynamisch abläuft, wurde eine transiente Berechnung durchgeführt. Dabei wurde die Verschiebung des oberen Zylinderrandes so vorgegeben, daß dieser sich mit der konstanten Geschwindigkeit von  $v = 0.01 \text{ mm/s}$  nach unten bewegt, gleichsam quasistatisch. Im Vorbeulbereich ergibt sich daher auch das gleiche Verhalten wie bei der klassischen statischen Simulation, siehe die Last-Verformungskurve des oberen Zylinderrandes in Abbildung 8.11. In diesem Bereich zeigt dann das Kriterium aus Gleichung (7.58) Stabilität an. Dieser Kurvenabschnitt ist in Abbildung 8.11 mit einer durchgezogenen Linie gekennzeichnet; die berechneten Deformationen wären also im statischen Sinne stabil. Der Beulvorgang selbst findet in einem sehr kurzen Zeit- bzw. kleinen Deformationsbereich statt und ist instabil im Sinne von Gleichung (7.58). In Abbildung 8.11 ist dieser Bereich mit einer gestrichelten Linie gekennzeichnet; aus statischer Sicht sind diese Deformationen nicht stabil.

Die Last-Verformungskurve des oberen Zylinderrandes des Beulvorgang ist in Abbildung 8.12 noch einmal vergrößert dargestellt. Zusätzlich zur durchgezogenen Last-Verformungskurve ist der Verlauf der kinetischen Energie  $\frac{1}{2} \dot{d} M \dot{d}$  des Systems mit einer gestrichelten Linie wiedergegeben. Der eigentliche Beulvorgang findet in einem Verschiebungsbereich von  $0.0003 \text{ mm}$  statt, in diesem Bereich steigt die kinetische Energie von nahe Null bis

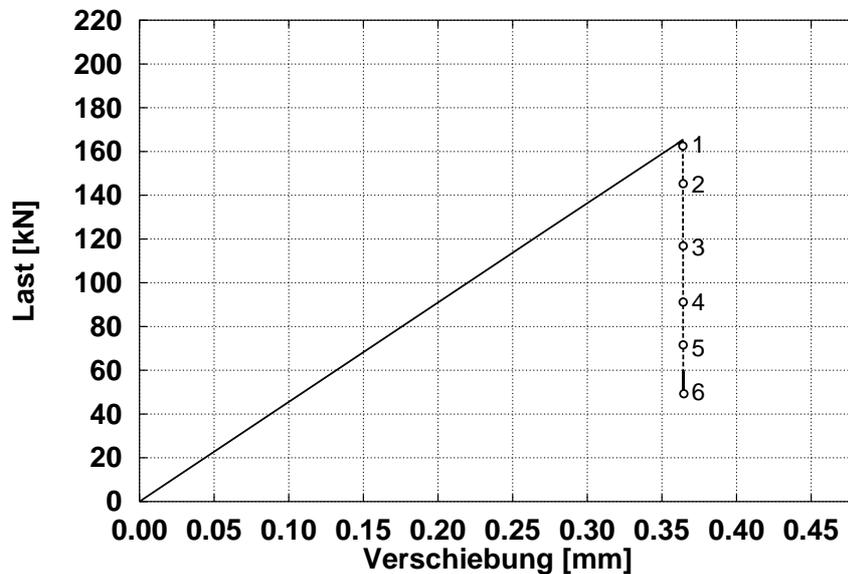


Abbildung 8.11: Last-Verformungskurve, transiente Belastung

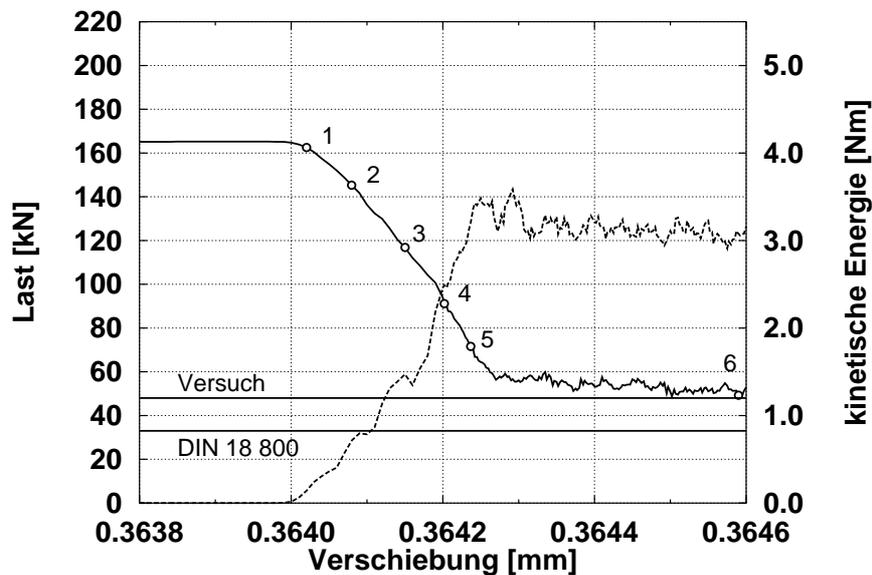


Abbildung 8.12: Last-Verformungskurve (Endbereich), transiente Belastung

ca.  $3 \text{ Nm}$  an, der Vorgang läuft also wie erwartet nicht statisch sondern dynamisch ab, wenngleich die kinetische Energie verglichen mit der inneren Energie am Beulpunkt von ca.  $60 \text{ Nm}$  sehr klein bleibt. Nach dem Beulvorgang stellt sich eine fast konstante Nachbeullast bei einem Niveau von etwa  $53 \text{ kN}$  ein. Dies stimmt überraschend gut mit dem tatsächlich im Versuch gemessenen Wert von  $48 \text{ kN}$  überein. Die DIN 18800, Teil 4 gibt eine Bemessungsbeullast für diesen Zylinder von  $33 \text{ kN}$  an, wobei zu beachten ist, daß hierzu Sicherheitsfaktoren eingehen. Im Nachbeulbereich setzt der Zylinder der aufgebrauchten Verschiebung einen konstanten Widerstand entgegen. In diesem Verformungszustand be-

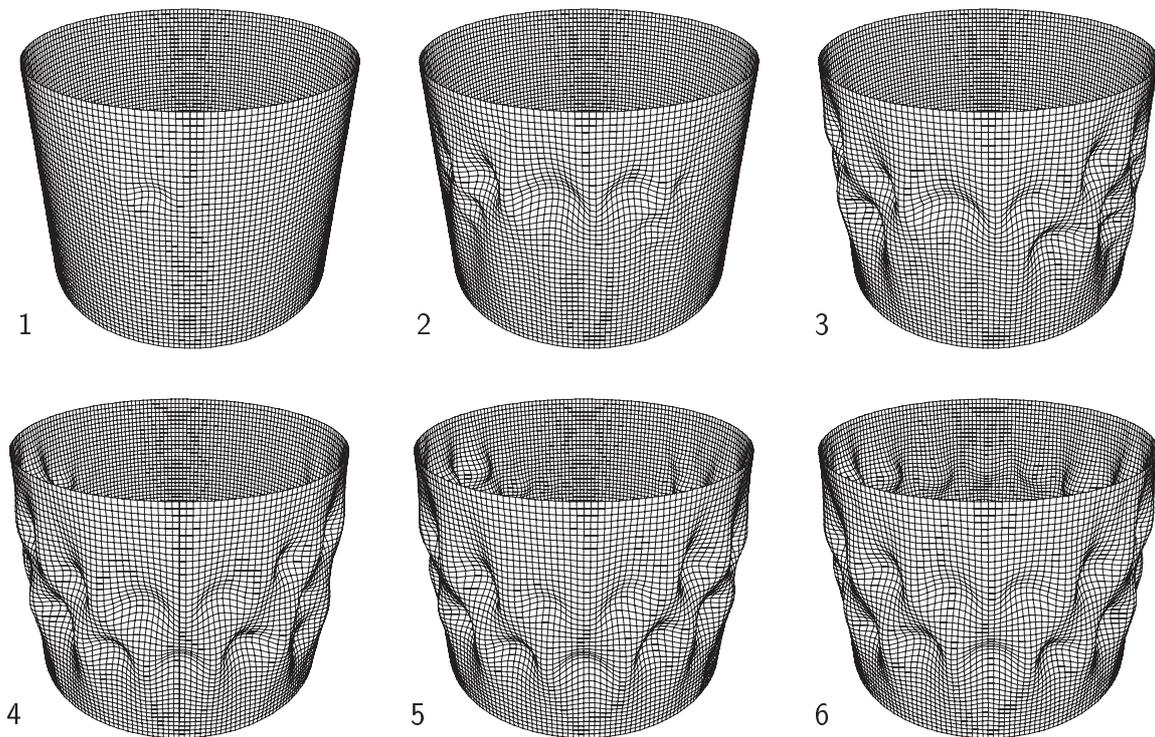


Abbildung 8.13: Verformungsfiguren, transiente verschiebungsgesteuerte Belastung, 10-fach überhöht

sitzt er also eine konstante Steifigkeit und ist in der Lage, eine Last zu tragen. Aus diesem Grund stellt die Nachbeullast eine aussagekräftige Größe zur Bemessung dar.

Die Verformungsfiguren in Abbildung 8.13 mit den zugehörigen Punkten auf der Last-Verformungskurve in den Abbildungen 8.11 und 8.12, zeigen eine Beulentwicklung, die wiederum ihren Ausgangspunkt an der Delle der imperfekten Struktur hat. Von hier breitet sich das Beulmuster über den gesamten Zylinder aus, bis sich schließlich ein regelmäßiges, von Versuchen bekanntes raute-förmiges Beulmuster ausgebildet hat. Dieses Verhalten steht im Einklang mit der Theorie der gestörten Verzweigung, wonach sich die Lösung des imperfekten Systems in einigem Abstand vom singulären Punkt asymptotisch der Lösung des perfekten Systems annähert (vergleiche Abbildung 7.2). Die Anzahl der Beulen in Umfangsrichtung beträgt 16. Das zweireihige Beulmuster befindet sich etwa in Zylindermittle. Würde elastoplastisches Materialverhalten berücksichtigt, so wäre das Rautenmuster mit wachsender Verschiebung noch stärker ausgeprägt.

### 8.6.2 Gewichtskraftgesteuerter Prozess

Für das Modell der massebehafteten Axialbelastung, d.h. eines aufliegenden Gewichtes unter Berücksichtigung dessen Massenträgheit ist die Last-Verformungskurve in Abbildung 8.14 dargestellt. Auch hier wird wieder die gleiche Beullast erreicht wie zuvor, da auch jetzt die Belastung sehr langsam erfolgt. Die Last von  $165\text{ kN}$  wird in einem simulierten Zeit-

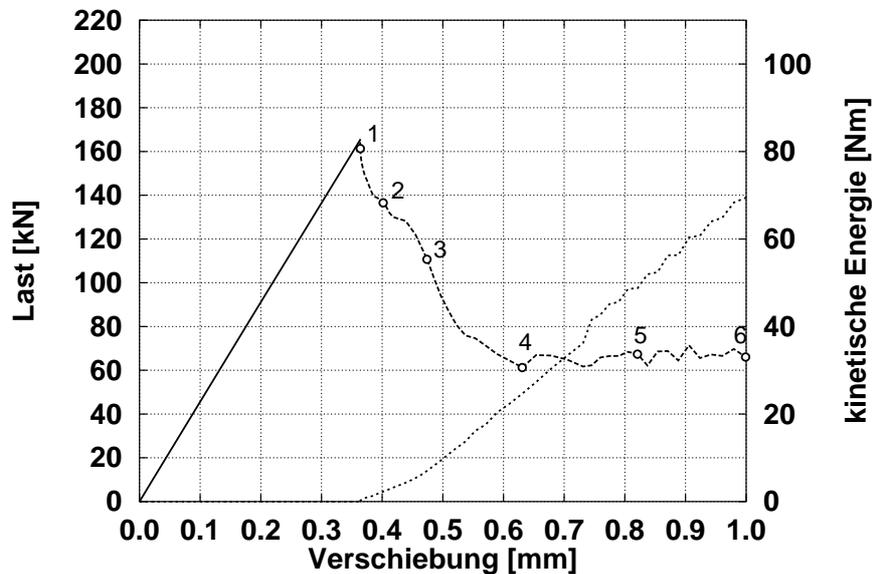


Abbildung 8.14: Last-Verformungskurve und kinetische Energie, Gewichtsbelastung

raum von 13  $h$  aufgebracht. Der Beulvorgang selbst ist dann mit größeren Verschiebungen verbunden, die sich bewegende Masse “schiebt” den Zylinder immer weiter zusammen. Dieser Vorgang dauert dann lediglich 1/20 Sekunde. Dabei vergrößert sich die Geschwindigkeit der Masse am oberen Zylinderrand kontinuierlich. Dies läßt die kinetische Energie, die in Abbildung 8.14 gestrichelt gekennzeichnet ist, weiter ansteigen. Sie bleibt nicht wie beim verschiebungsgesteuerten Prozess im Nachbeulbereich nahezu konstant. Aber auch hier stellt sich eine fast konstante Nachbeullast ein, sie liegt bei ca. 60  $kN$ , etwas höher als beim verschiebungsgesteuerten Prozess.

Die zugehörigen Verschiebungszustände sind in Abbildung 8.15 wiedergegeben, die entsprechenden Zustände in der Last-Verformungskurve in Abbildung 8.14 sind wiederum durch Punkte gekennzeichnet. Die Verschiebungen der Zustände 1-3 sind mit dem Faktor 10 skaliert, die der Zustände 4-6 mit dem Faktor 3. Wegen der größeren Verschiebungen sind die Beulmuster deutlicher ausgeprägt als beim verschiebungsgesteuerten Vorgang. Auch hier beginnt der Beulvorgang wieder an der Stelle mit der Delle im Imperfektionsmuster, siehe Abbildung 8.15<sub>1</sub>. Das im Verlauf zum Teil sehr hochwellige Beulmuster – aufgrund der geringen Ausprägung in Abbildung 8.15 nicht gut erkennbar – entwickelt sich im weiteren Vorgang zu einem regelmäßigen, rautenförmigen Muster ähnlich zu dem beim vorherigen verschiebungsgesteuerten Prozess.

### 8.6.3 Vergleich beider Prozesse

In beiden Fällen, dem verschiebungsgesteuerten Vorgang und dem gewichtsgesteuerten Vorgang wird bei langsamer Lastaufbringung der quasistatische Vorbeulpfad berechnet, die ermittelten Lösungen sind hierfür im statischen Sinne stabil. In beiden Fällen beginnt die Entwicklung des Beulmusters an einer lokalen Imperfektion und breitet sich dann über den

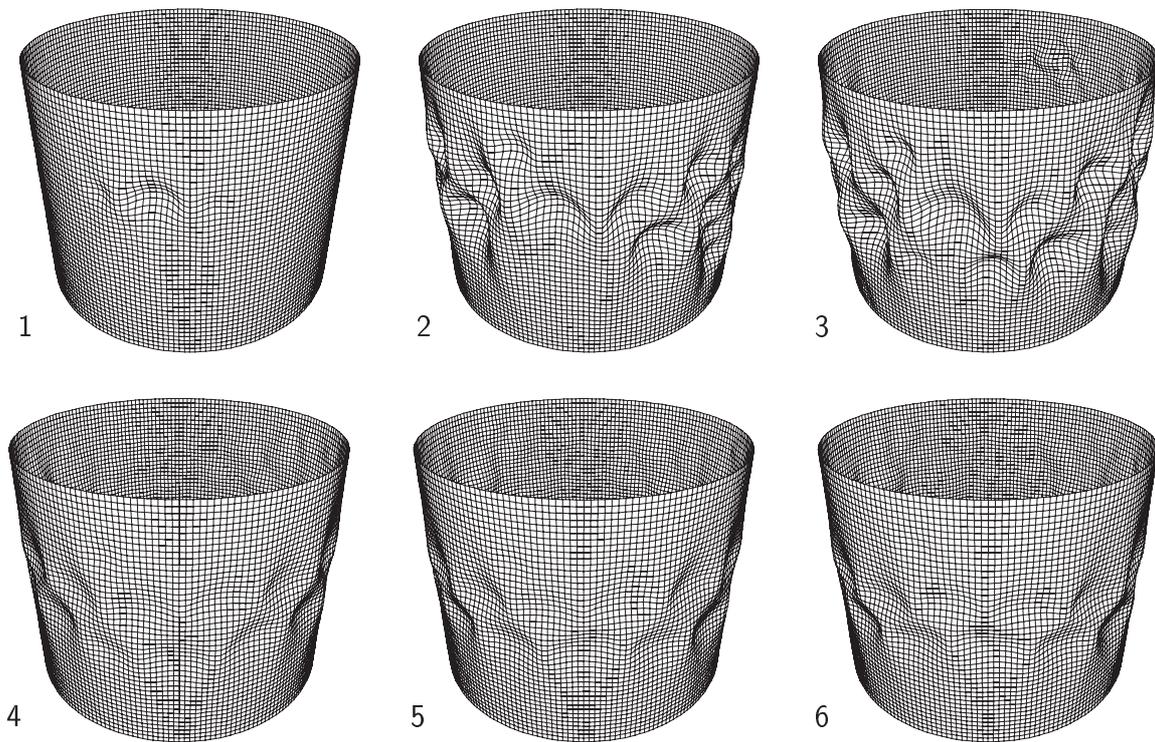
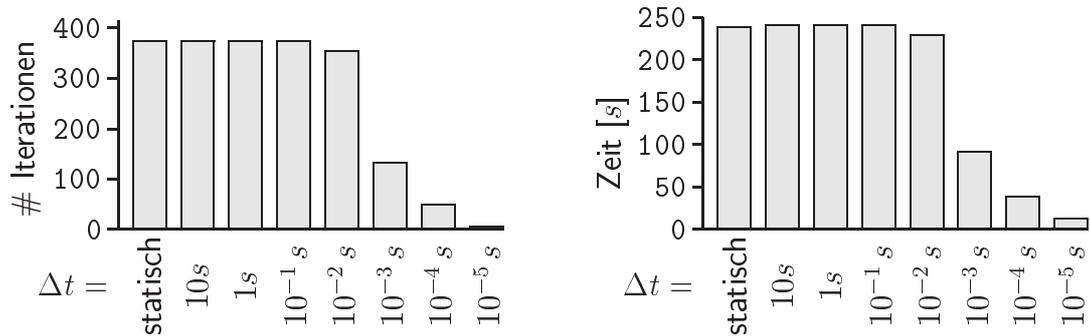


Abbildung 8.15: Verformungsfiguren, Gewichtsbelastung, 10 (1-3) bzw. 3-fach (4-6) überhöht

gesamten Zylinder aus. Am Ende resultiert daraus ein jeweils zweireihiges Rautenbeulmuster. Ebenso bleibt in beiden Fällen die Nachbeullast fast konstant, sie differiert jedoch geringfügig, was bei der Berechnung eines physikalisch unterschiedlichen Problems nicht überrascht. Der gewichtsgesteuerte Beulvorgang ist aber mit größeren Verschiebungen verbunden, die trägheitsbehaftete Masse am oberen Zylinderrand “schiebt” den Zylinder immer weiter zusammen.

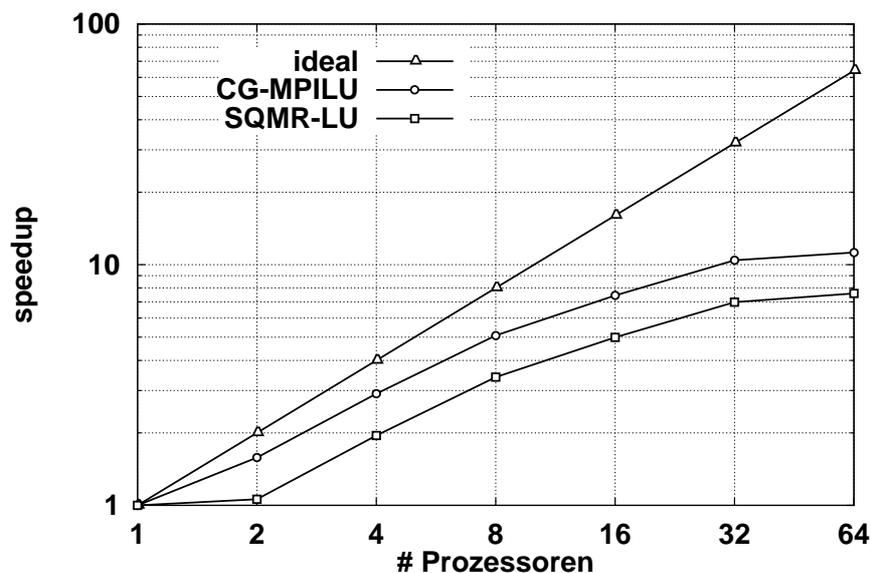
#### 8.6.4 Rechentechnische Aspekte

Der rechenzeitintensivste Teil der transienten Berechnung ist die Bestimmung des eigentlichen Beulvorganges, da dieser im Gegensatz zum Vorbeulbereich mit sehr kleinen Zeitschrittweiten berechnet werden muß. Letzteres hat allerdings bei der Verwendung des Newmark Verfahrens zur Zeitintegration den Vorteil, daß die effektive Steifigkeitsmatrix  $\mathbf{K}^*$  aus Gleichung (2.82) gut konditioniert ist, da der Massenanteil dominiert. Sie ist im Gegensatz zur tangentiellen Steifigkeitsmatrix  $\mathbf{K}_T$  für dieses Problem sogar stets positiv definit. Daher eignen sich für die transienten Probleme auch iterative Lösungsverfahren zur linearen Gleichungslösung gut. Für unterschiedliche Werte der Zeitschrittweite  $\Delta t$  sind in Abbildung 8.16 sowohl die Iterationsanzahl als auch die Rechenzeit zur Lösung eines linearen Gleichungssystems mit der effektiven Steifigkeitsmatrix als Koeffizientenmatrix dargestellt, vgl. [122]. Es wurde das cg-Verfahren mit MPILU-Vorkonditionierung

Abbildung 8.16: Zur Konditionierung der effektiven Steifigkeitsmatrix  $\mathbf{K}^*$ 

gewählt. Die Anzahl der Iterationen und damit einhergehend die Rechenzeit verringert sich deutlich für kleinere Zeitschrittweiten. Für große Zeitschrittweiten ändert sich das Verhalten des iterativen Löser kaum im Vergleich zum statischen Fall, d.h. der Lösung mit der tangentiellen Steifigkeitsmatrix als Koeffizientenmatrix. Im Vorbeulbereich, der mit großen Zeitschritten berechnet werden kann, sind demnach hohe Iterationszahlen zu erwarten, die sich aber während des Beulvorganges, der mit sehr kleinen Zeitschritten berechnet werden muß, deutlich verringern.

Daher eignet sich die transiente Berechnung des Beulvorganges ungleich besser zur parallelen Bearbeitung als die statische Analyse. Unter Anwendung des MPILU-vorkonditionierten cg-Verfahrens wurde auch ein *speedup* berechnet, siehe Abbildung 8.17. Dabei wurden die erhaltenen Rechenzeiten einerseits auf das äquivalente sequentielle Verfahren bezogen (CG-MPILU) und andererseits auf das schnellste sequentielle Verfahren (SQMR-LU). Durch den Einsatz von Parallelrechnern läßt sich die Rechenzeit von 13 Stunden (CG-

Abbildung 8.17: *Speedup* für transiente Berechnung

MPILU) bzw. 10.25 Stunden (SQMR-LU) auf 1.5 Stunden (CG-MPILU, 32 Prozessoren) reduzieren. Dabei flacht die *speedup* Kurve bei Einsatz von mehr als 16 Prozessoren merklich ab; hier macht sich die relativ kleine Problemgröße von 9200 Elementen bemerkbar, bei Verwendung von 32 Prozessoren bearbeitet jeder Prozessor nur noch ca. 290 Elemente.

Wünschenswert für die transiente Berechnung, für sequentielle ebenso wie für parallele Analysen, ist eine automatische Zeitschrittweitensteuerung, die bislang nur für lineare Analysen entwickelt wurde, siehe z.B. [117]. Zwar läßt sich das dort beschriebene Verfahren auch auf nichtlineare Probleme übertragen, wird aber mit dem aktuellen, möglicherweise zu großen Zeitschritt keine Konvergenz erzielt, so versagt dieses Verfahren. Da bei dem untersuchten Zylinder der Beulvorgang sehr plötzlich einsetzt, wird mit einem großen Zeitschritt aus der Berechnung des Vorbeulverhaltens keine Konvergenz mehr erzielt, die automatische Steuerung versagt. Deshalb bewähren sich hier simple Schrittweitensteuerungen, die aufgrund der Anzahl von benötigten Newton-Raphson Iterationen die Zeitschrittweite verkleinern oder auch erhöhen. Sollte keine Konvergenz erzielt werden, muß eine neue Iteration vom letzten auskonvergierten Zustand ausgehend mit kleinerer Schrittweite erfolgen.

## 8.7 Sensitivität von Gleichgewichtslagen im Vorbeulbereich

Mit den vorangegangenen Simulationen wurde ein Beulvorgang simuliert, der von einer kontinuierlich anwachsenden Belastung hervorgerufen wird. Dies erlaubt aber keine Aussage über die Sicherheit einer Gleichgewichtslage im Vorbeulbereich, d.h. ihrer Sensitivität. Letzteres soll in diesem Abschnitt diskutiert werden.

Dabei wird der Zylinderschale bei unterschiedlichem Lastniveau unterschiedlich viel kinetische Energie durch Anfangsgeschwindigkeiten  $v_0$  eingeprägt. Die Größe der kinetischen Energie, die das System erträgt, ohne zu Beulen, wird als quantitatives Maß für die Stabilität betrachtet, der reziproke Wert als Maß für die Sensitivität. Die Vorgehensweise kann als Suche nach der Separatrix im  $2n$ -dimensionalen Raum aufgefaßt werden, der Begriff der Sensitivität ist dann eng mit dem Stabilitätsgrad, der den kleinsten Radiusvektor zur benachbarten Separatrix angibt, verknüpft. Als Vergleichsgröße zu den ermittelten Energien kann die innere Energie des Zylinders im Beulpunkt dienen, sie beträgt  $W_{kin} = 60 Nm$ .

Eine ungeklärte Frage hierbei ist, welche Form der Anfangsgeschwindigkeitsverteilung  $v_0$  zur kleinsten kinetischen Energie führt, die zum Verlassen des Einzugsbereich der untersuchten stabilen Gleichgewichtslage, und damit zur größten Sensitivität führt. Hier können aber auch die vorangegangenen Untersuchungen Hinweise liefern. Bei den Verformungsfiguren der statischen Analysen in Abbildung 8.9 ändert sich die Beulfigur im ersten Nachbeulbereich gegenüber dem Vorbeulbereich kaum, sodaß eine Initiierung des Beulvorganges durch die Vorbeulfigur möglich erscheint. Ebenso könnte sich die mittels der transienten Berechnung ermittelte Nachbeulfigur aus Abbildung 8.13<sub>6</sub> als kritisch erweisen. Darüber hinaus ist bekannt, daß die Eigenmoden kritische Formen darstellen. Diese unterschiedlichen Formen der Anfangsgeschwindigkeitsverteilungen werden im Folgenden auf Gleichgewichtslagen aufgebracht und die Systemantwort untersucht.

Diese Untersuchungen zur Sensitivität der vorgegebenen mechanischen Struktur mit imperfekter Geometrie sind streng zu unterscheiden von der Imperfektionsempfindlichkeit einer Struktur. So ist es hier nicht Ziel, wie z.B. in [29, 28], das Imperfektionsmuster, das zur minimalen ersten Beullast führt, zu bestimmen, sondern die Geometrie bleibt fest vorgegeben. Daß es sich um einen imperfekten Zylinder handelt, ist dabei nebensächlich. Ebenso könnte es sich um eine beliebige – auch perfekte – Geometrie handeln. Gleichwohl sei angemerkt, daß eine perfekte Struktur aufgrund der Vielzahl von vorhandenen Verzweigungen und der Häufung von Singularitäten zu sogenannten *mode-jumping* Phänomenen führen könnte [120] und somit eine imperfekte Struktur einfacher handhabbar ist, beziehungsweise umgekehrt nur unter der Berücksichtigung wirklichkeitsnaher Imperfektionen strukturell stabile und robuste Analysen möglich sind. Während also die Entwicklungen in [28] zum Ziel haben, dasjenige Imperfektionsmuster zu ermitteln, das zur geringsten Beullast führt, ist es hier von Interesse die Sicherheit eines Lastzustandes einer fest vorgegebenen mechanischen Struktur zu beurteilen. Dennoch wäre auch der Einfluß einer “ungünstigsten” Imperfektionsfigur auf die Sensitivität der Struktur sehr interessant.

### 8.7.1 Definition der Anfangsgeschwindigkeitsverteilungen

Wie bereits erwähnt, sollen für die folgenden Untersuchungen im Wesentlichen drei konkrete Anfangsgeschwindigkeitsverteilungen  $\mathbf{v}_0$  verwendet werden, die besonders plausibel erscheinen. Soweit es sich nicht um Eigenschwingungsformen handelt, d.h. es nicht Lösungen von

$$(\mathbf{K}_L - \lambda \mathbf{M})\boldsymbol{\varphi} = \mathbf{0} \quad (8.2)$$

sind, sollen diese Formen  $\mathbf{v}$  in Eigenschwingungsformen  $\boldsymbol{\varphi}_i$  entwickelt werden:

$$\mathbf{v} = b_1 \boldsymbol{\varphi}_1 + b_2 \boldsymbol{\varphi}_2 + b_3 \boldsymbol{\varphi}_3 + \dots = \mathbf{B}\mathbf{Q}. \quad (8.3)$$

Wegen

$$\mathbf{Q}^T \mathbf{M} \mathbf{Q} = \mathbf{I} \quad \text{und} \quad \mathbf{Q}^T \mathbf{K}_L \mathbf{Q} = \Lambda \quad (8.4)$$

lassen sich die gewünschten Koeffizienten  $b_i$  aus

$$\mathbf{B} = \mathbf{v} \mathbf{Q}^T \mathbf{M} \quad (8.5)$$

berechnen. Daraus kann man dann die Anteile der jeweiligen Schwingungseigenformen an der Störungsform  $\mathbf{v}$  ablesen.

In Abbildung 8.18 sind die Koeffizienten  $b_i$  der ersten 100 Eigenformen für das Endbeulmuster aus Abbildung 8.13<sub>6</sub> und der Verschiebungsfigur gemäß Abbildung 8.9 für das Lastniveau von 80  $kN$  und 120  $kN$  gegeben. Dabei wurde mit der  $\infty$ -Norm normiert, d.h.

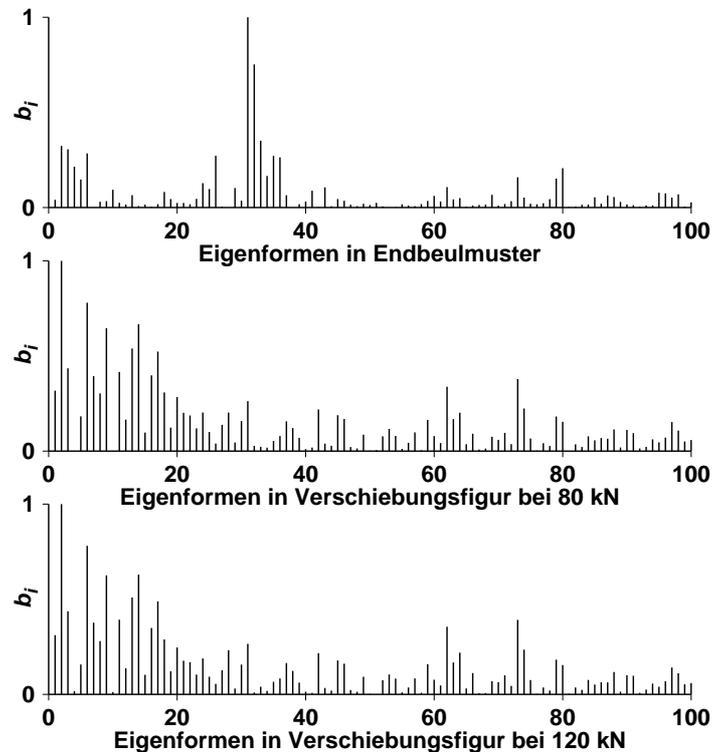


Abbildung 8.18: Eigenkoordinaten der Anfangsgeschwindigkeitsverteilungen

der Maximalwert beträgt 1. Beim Endbeulmuster der transienten Berechnung zeigt sich eine Dominanz einiger höherer Formen, zwischen der 30. und 36. Eigenform. Ebenso sind einige niedrige Eigenformen auch gut erkennbar beteiligt.

Bei der Verschiebungsfigur beim Lastniveau von  $80 \text{ kN}$  dominieren niedrige Eigenformen, es sind jedoch auch höhere Eigenformen enthalten, die hier größere Amplituden besitzen als beim Endbeulmuster. Mit dieser Störung werden demnach auch höhere Frequenzen stärker angeregt. Die Störung mit der Verschiebungsform bei  $120 \text{ kN}$  unterscheidet sich von der bei  $80 \text{ kN}$  praktisch nicht, die Verschiebungsfigur ändert sich qualitativ nicht.

Die erste Eigenschwingungsform der zylindrischen Struktur ist in Abbildung 8.19 dargestellt. Sie stellt ein reines Längsmuster dar.

### 8.7.2 Gleichgewichtslage bei $120 \text{ kN}$

Als erste zu untersuchende stabile Gleichgewichtslage wird die bei  $120 \text{ kN}$  gewählt. Aus den statischen Untersuchungen ist bekannt, daß sich mindestens eine instabile Gleichgewichtslage in der Nachbarschaft befindet, deren Verschiebungsfigur der stabilen Gleichgewichtslage zumindest ähnelt. Es wird zunächst die statische Gleichgewichtslage berechnet und anschließend eine transiente Berechnung unter konstanter Last durchgeführt, wobei eine endlich große Anfangsgeschwindigkeit vorgegeben wird. Um hochfrequente Anteile zu unterdrücken, wird eine numerische Dämpfung über das Newmark Verfahren eingebracht.

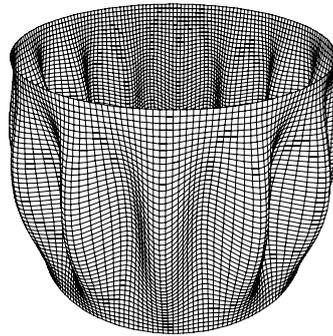
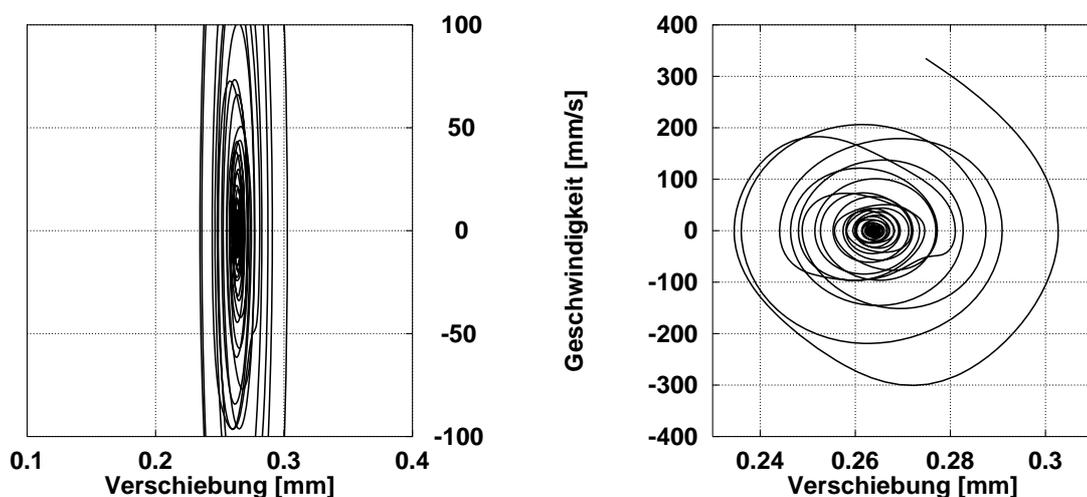


Abbildung 8.19: Erste Schwingungseigenform

Zuerst wird die Anfangsgeschwindigkeit in Form der Verschiebungsfigur vorgegeben, diese Form also verstärkt. In Abbildung 8.20 ist das Phasenportrait des oberen Zylinderrandes der sich einstellenden Bewegung dargestellt. Zur besseren Vergleichbarkeit der Bewegung wird im jeweils linken Diagramm eine konstante Skalierung beibehalten, im rechten Diagramm ist eine für die dargestellte Bewegung optimale Skalierung gewählt. Es ist eine abklingende Schwingung um die Gleichgewichtslage erkennbar. Die Anfangsgeschwindigkeit wurde so gewählt, daß sich eine Energie von  $W_{kin} = 0.5 Nm$  ergibt. Bei der Betrachtung des statischen Residuums ist die Nähe einer weiteren Gleichgewichtslage nicht zu erkennen, siehe Abbildung 8.21. Bei von der stabilen statischen Gleichgewichtslage abweichenden Verformungen wächst das statische Residuum an, weitere Gleichgewichtslagen in diesen Formen sind nicht zu vermuten. Ähnliche Kurvenverläufe ergeben sich auch für erheblich größere Energien. Ein unbegrenztes Ansteigen der Verschiebung der Zylinderoberseite konnte erst bei einer kinetischen Energie von ca.  $W_{kin} = 50 Nm$ , also in Größenordnung der inneren Energie, beobachtet werden. Das System reagiert auf diese Störform insensitiv.

Abbildung 8.20: Phasendiagramm bei Störung mit Verformungsfigur,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$

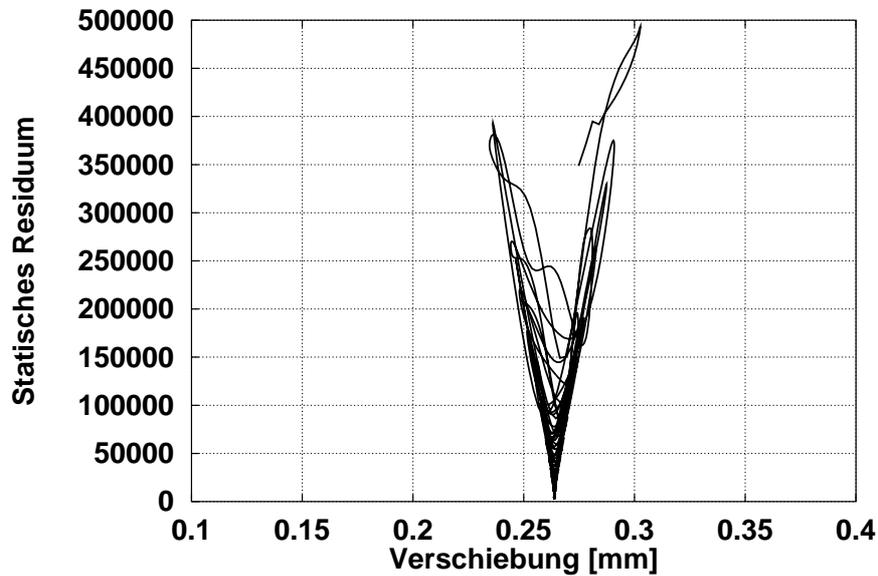


Abbildung 8.21: Statisches Residuum bei Störung mit Verformungsfigur,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$

Ein gleichartiges Systemverhalten stellt sich auch bei der Störung mit der Endbeulfigur aus Abbildung 8.13<sub>6</sub> ein. Das Phasendiagramm ist in Abbildung 8.22 wiedergegeben und beschreibt eine abklingende Schwingung in der vorgegebenen Beulform. Die vorgegebene Energie der Störung beträgt wie zuvor  $W_{kin} = 0.5 Nm$ . Auch hier gleicht der Verlauf der statischen Residuumsnorm dem vorherigen bei der Anfangsgeschwindigkeit in der Verformungsfigur, siehe Abbildung 8.23. Es läßt sich auch hier keine statische Gleichgewichtslage in der Nachbarschaft vermuten. Mit dieser Anfangsgeschwindigkeitsverteilung läßt sich

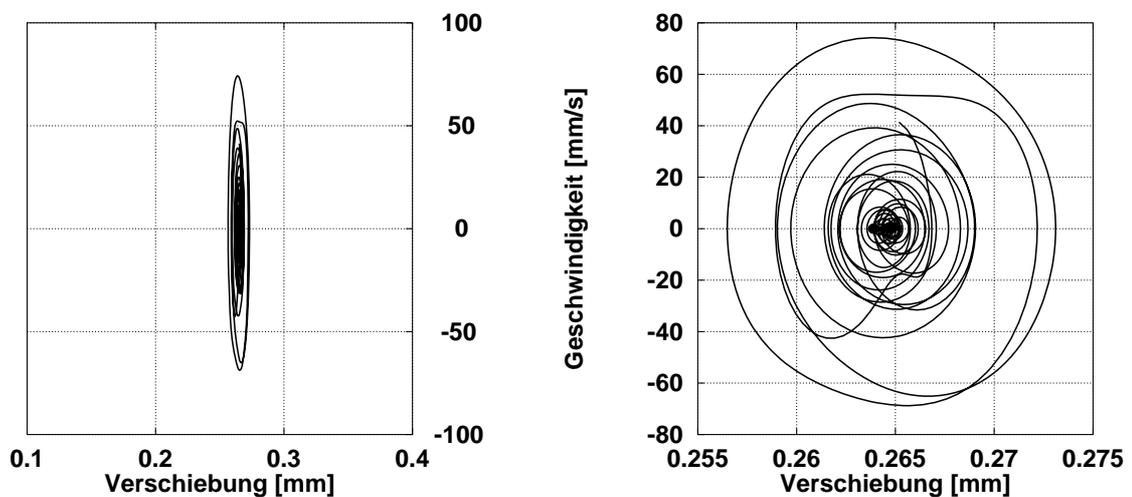


Abbildung 8.22: Phasendiagramm bei Störung mit Endbeulmuster,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$

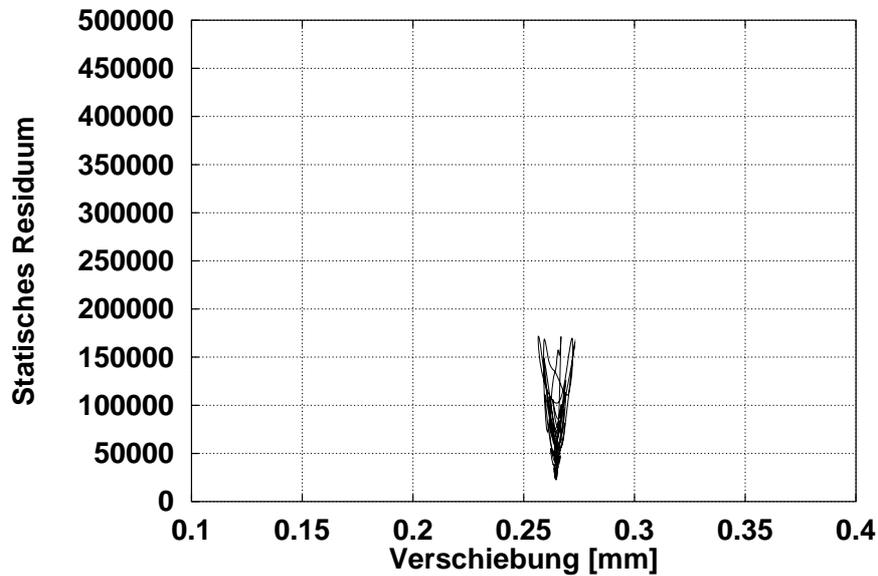


Abbildung 8.23: Statisches Residuum bei Störung mit Endbeulmuster,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$

ein Beulvorgang bereits mit einer kinetischen Energie von  $W_{kin} = 1.2 Nm$  erreichen. Der Zylinder reagiert auf diese Anfangsgeschwindigkeitsverteilung deutlich sensibler als auf die Verschiebungsfigur der statischen Gleichgewichtslage.

Noch sensibler reagiert der untersuchte Zylinder auf eine Anfangsgeschwindigkeitsverteilung in der ersten Eigenform. Wie aus der Phasenkurve in Abbildung 8.24 ersichtlich ist, steigt die Verschiebung des oberen Zylinderrandes nach einigen Schwingungsvorgängen an, der Zylinder beult. Die stabile Ausgangsgleichgewichtslage ist durch einen • ge-

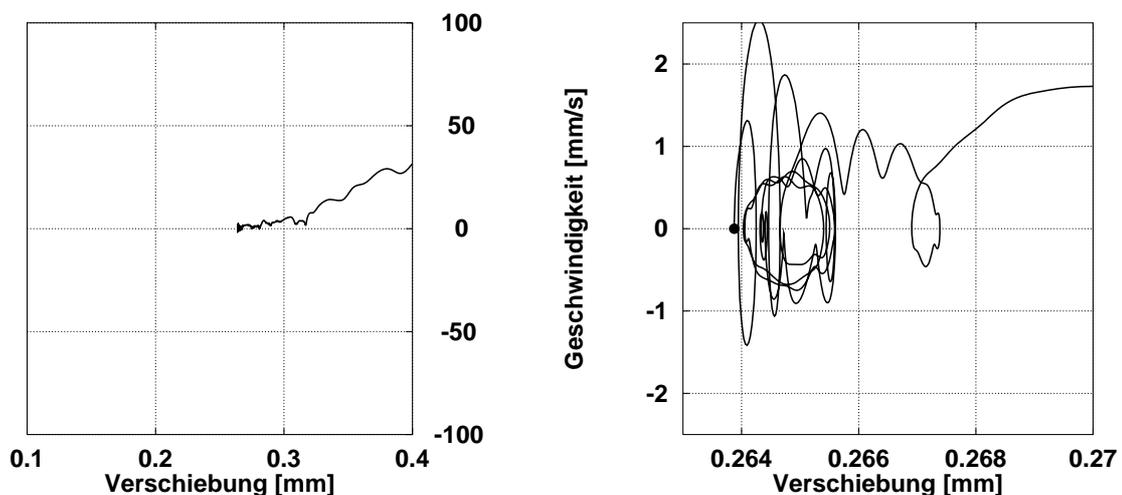


Abbildung 8.24: Phasendiagramm bei Störung mit erster Eigenform,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$

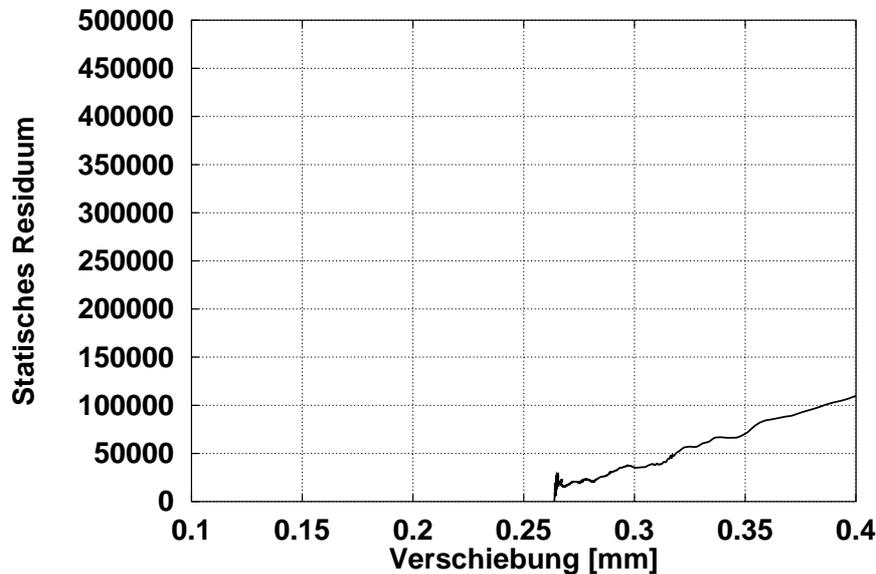


Abbildung 8.25: Statisches Residuum bei Störung mit erster Eigenform,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$

kennzeichnet. Die kinetische Energie dieser Anfangsgeschwindigkeit beträgt wiederum  $W_{kin} = 0.5 Nm$ , die Struktur reagiert auf diese Form sensibler als auf die beiden vorangegangenen. Auch hier kann der statischen Residuumsnorm in Abbildung 8.25 abermals kein Hinweis auf eine statische Gleichgewichtslage entnommen werden. Für diesen Beulvorgang sind in Abbildung 8.26 charakteristische Verformungsfiguren dargestellt. Bei der ersten Verformungsfigur hat sich die als Anfangsgeschwindigkeit aufgebrachte erste Eigenschwingungsform voll entwickelt. Bei der zweiten Verformungsfigur ist bereits eine Beule an der rechten unteren Seite erkennbar, die sich bei der dritten dargestellten Verformungsfigur weiter ausgeprägt hat. Von dort ausgehend breitet sich anschließend das Beulmuster weiter aus, bis sich wie in der 6. Verformungsfigur das Beulmuster über den gesamten Zylinder erstreckt. Anzumerken bleibt, daß dieser Beulvorgang nicht wie bei den transienten Berechnungen der Abschnitte 8.6.1 und 8.6.2 von der ausgeprägten Delle der ursprünglichen Geometrie ausgeht, obwohl auch hier die erste Schwingungseigenform ein Maximum aufweist, d.h. hier eine starke Anregung erfolgt, sondern von der großflächigeren Imperfektion an der rechten vorderen Seite (vergleiche Abbildung 8.3).

Eine Störung in der zweiten Eigenform wiederum mit der kinetischen Energie von  $W_{kin} = 0.5 Nm$  verursacht kein Beulen der Zylinderstruktur, siehe Abbildung 8.27. Mit dieser Anfangsgeschwindigkeit werden nur sehr kleine Verschiebungen des oberen Zylinderrandes erreicht. Bei einer Simulation über einen längeren als den dargestellten Zeitraum, kommt das System wieder bei der mit • gekennzeichneten Ausgangsgleichgewichtslage zur Ruhe.

Zusammenfassend erweist sich für diese Gleichgewichtslage also die erste Eigenschwingungsform als die kritischste der untersuchten Störungen. Mit der Zuführung der sehr geringen Energie von  $W_{kin} = 0.5 Nm$  verglichen mit der inneren Energie des Zylinders von

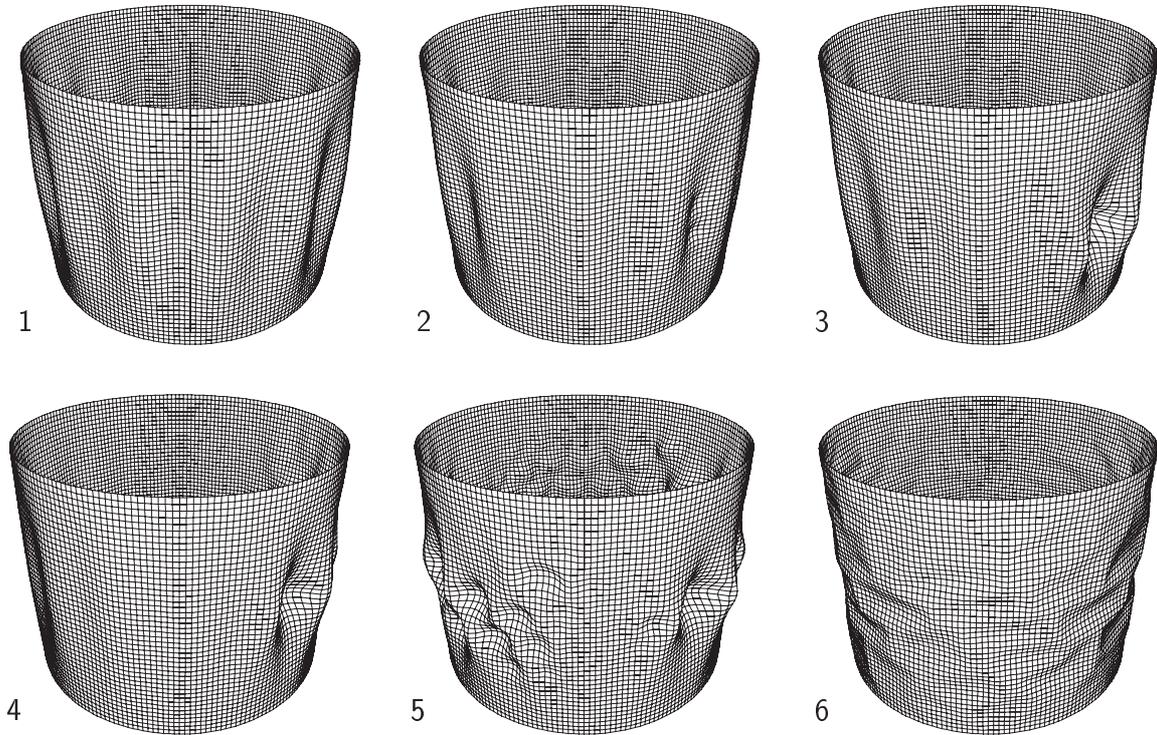


Abbildung 8.26: Verformungsfiguren, Störung mit erster Schwingungseigenform,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$ , skaliert mit dem Faktor 20 (1-3), 10 (4-5) und 1 (6)

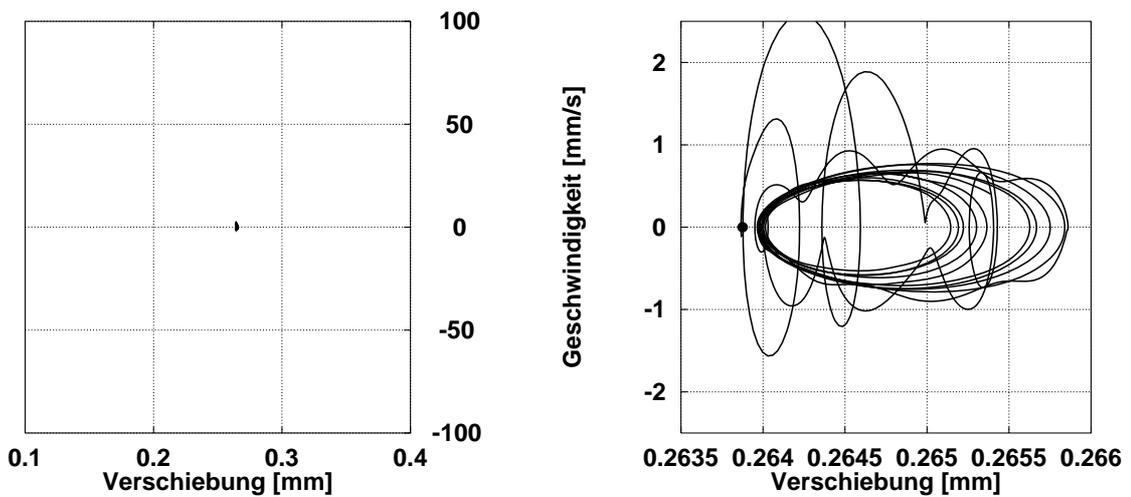


Abbildung 8.27: Phasendiagramm bei Störung mit zweiter Eigenform,  $W_{kin} = 0.5 Nm$ , Last  $120 kN$

ca.  $60 Nm$ , kann ein Beulen der Struktur ausgelöst werden. Diese Gleichgewichtslage ist aufgrund der geringen erforderlichen Energie demnach von hoher Sensitivität.

### 8.7.3 Gleichgewichtslage bei $80 kN$

Bei der statisch stabilen Gleichgewichtslage bei  $80 kN$  ist nichts über eine benachbarte Gleichgewichtslage bekannt. Dieses Lastniveau liegt jedoch oberhalb der durch die transienten Simulationen ermittelten Nachbeullast von ca.  $53 kN$ , sodaß ein Beulen der Struktur erwartet werden kann. Nach den Ergebnissen aus der Untersuchung der Gleichgewichtslage bei  $120 kN$  sollte auch hier die erste Schwingungseigenform eine kritische Störung darstellen. Tatsächlich kann durch die Störung mit der ersten Schwingungseigenform mit einer Energie von  $W_{kin} = 2.7 Nm$  ein Beulen des Zylinders ausgelöst werden, siehe Abbildung 8.28. In Abbildung 8.29 sind die zu diesem Beulvorgang zugehörigen Verformungsfiguren dargestellt. Zunächst bildet sich wieder die angeregte erste Schwingungseigenform aus 8.29<sub>1</sub>. Anders jedoch, als bei der höheren Last von  $120 kN$ , geht dieses die gesamte Länge des Zylinders umfassende Beulmuster in ein Muster über, das in der Zylindermittle konzentriert ist 8.29<sub>2-3</sub>. Mit dem Einfallen weiterer Beulen ober- und unterhalb dieses mittigen Beulmusters 8.29<sub>4</sub> breitet sich das Muster über den gesamten Zylinder aus 8.29<sub>5</sub>. Abbildung 8.29<sub>6</sub> zeigt dann das ausgeprägte Beulmuster unskaliert.

Bei einer Störung mit dem Endbeulmuster aus Abbildung 8.13<sub>6</sub> stellt sich ein Versagen des Zylinders bei einer Störung mit einer Energie von  $W_{kin} = 5 Nm$  ein, siehe Abbildung 8.30. Wie bei der Störung mit der ersten Schwingungseigenform nimmt auch hier die Sensitivität beim niedrigeren Lastniveau ab, abermals ist die Sensitivität gegenüber der Eigenform höher. In Abbildung 8.31<sub>1</sub> ist das angeregte Beulmuster gut erkennbar, dieses

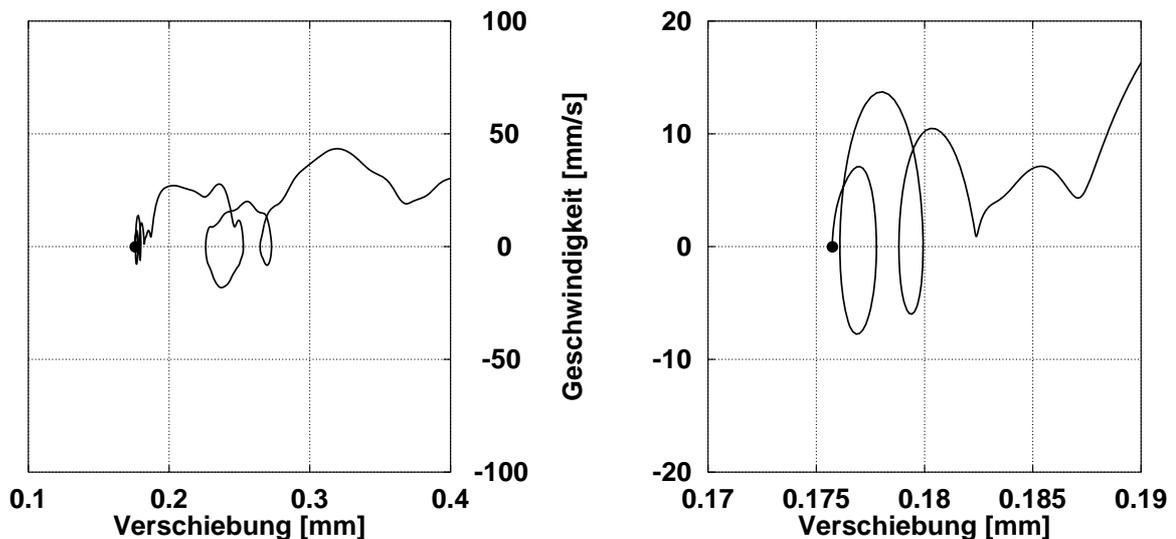


Abbildung 8.28: Phasendiagramm bei Störung mit erster Schwingungseigenform,  $W_{kin} = 2.7 Nm$ , Last  $80 kN$

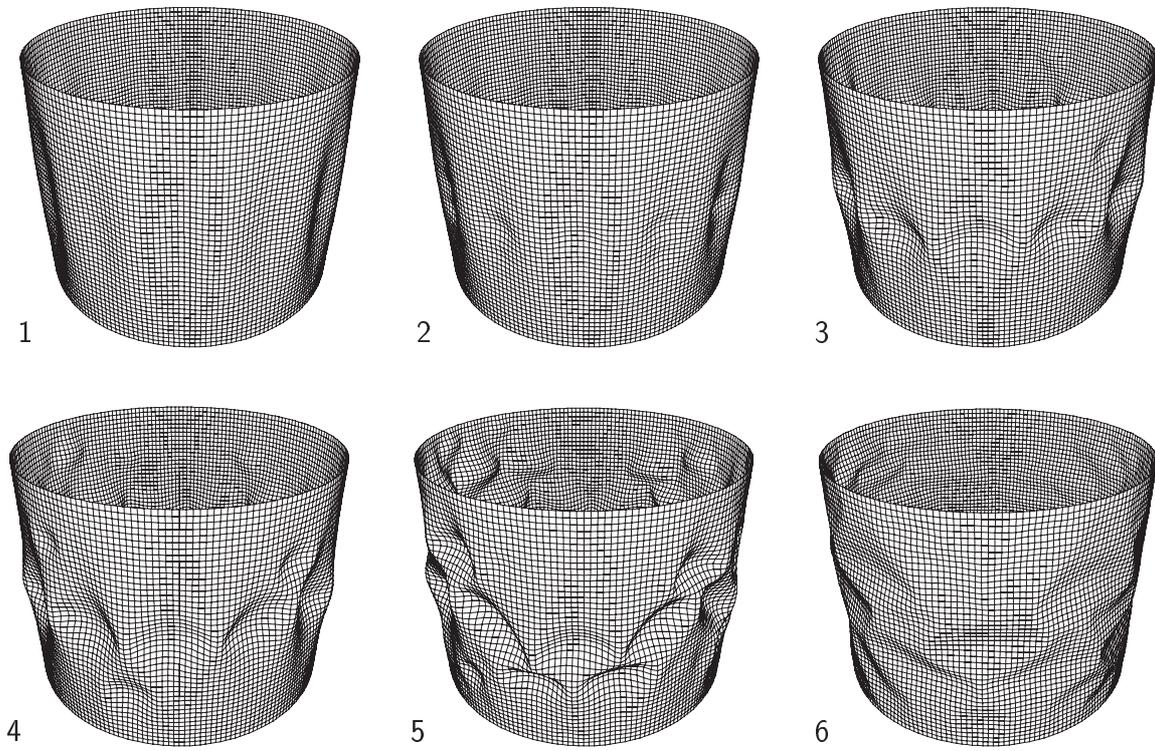


Abbildung 8.29: Verformungsfiguren, Störung mit erster Schwingungseigenform,  $W_{kin} = 2.7 Nm$ , Last  $80 kN$ , skaliert mit dem Faktor 5 (1-5) und 1 (6)

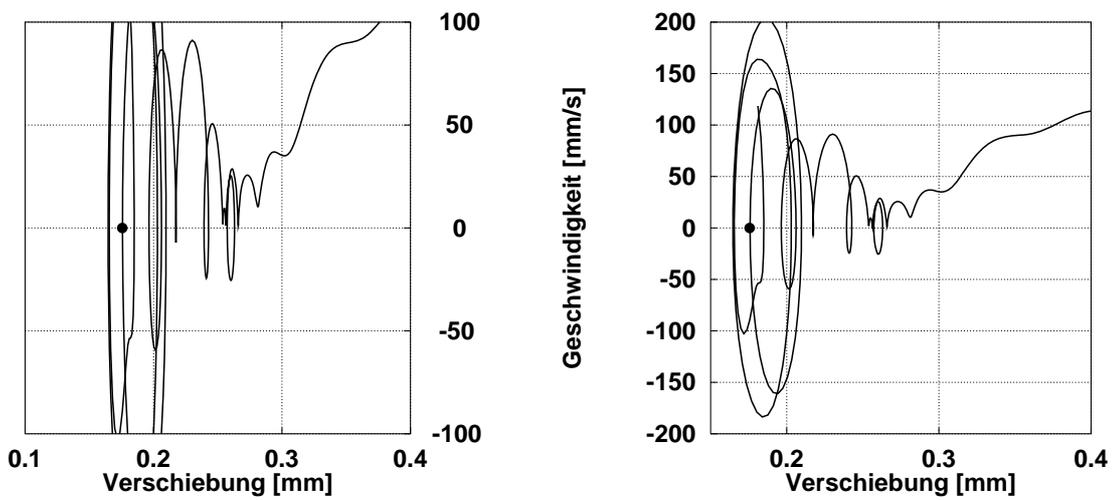


Abbildung 8.30: Phasendiagramm bei Störung mit Endbeulmuster,  $W_{kin} = 5 Nm$ , Last  $80 kN$

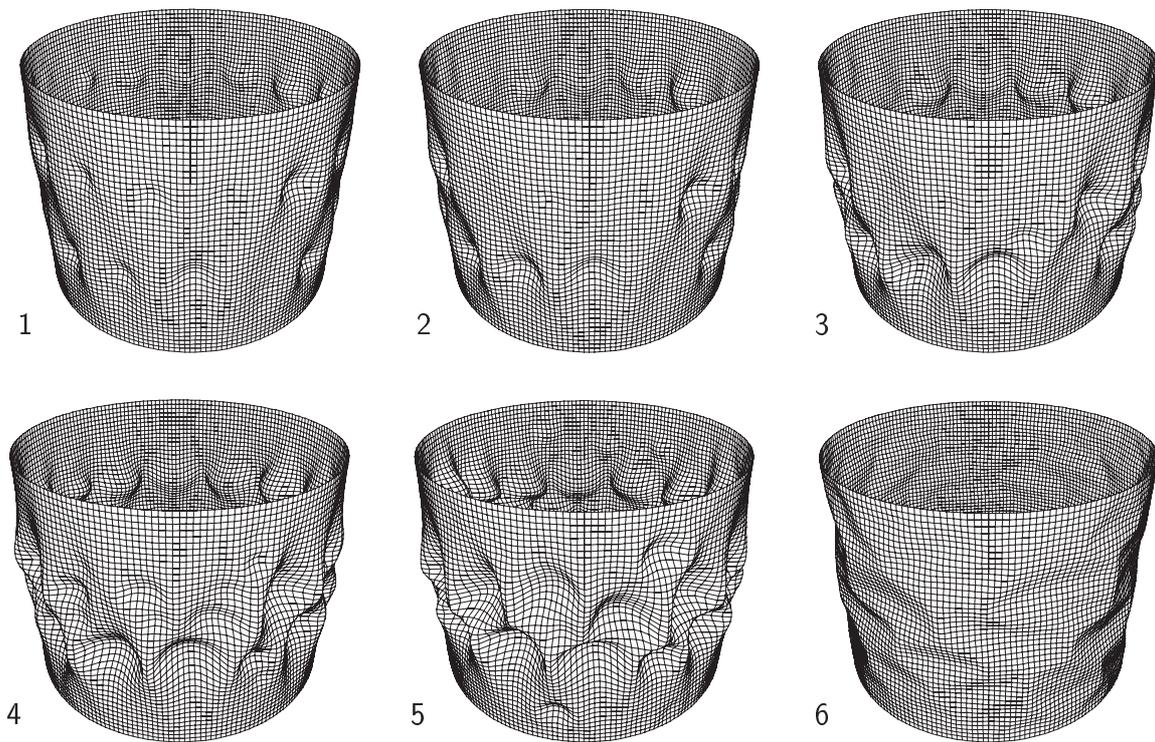


Abbildung 8.31: Verformungsfiguren, Störung mit Endbeulform,  $W_{kin} = 5 Nm$ , Last  $80 kN$ , skaliert mit dem Faktor 10 (1-5) und 1 (6)

Muster ändert sich während des Beulvorganges nicht wesentlich, im Gegensatz zu dem Beulvorgang aufgrund der Störung mit der ersten Schwingungseigenform.

Die Vermutung, daß die Störung mit der Verschiebungsfigur, die aus der transienten Berechnung bei der Last von  $80 kN$  resultiert, zu kleineren kinetischen Energien führt, bestätigt sich nicht. Eine Störung mit dieser Figur mit der kinetischen Energie von  $W_{kin} = 5 Nm$  führt nicht zum Beulen des Zylinders.

Auf eine Störung mit der aus der statischen Berechnung resultierenden Verschiebungsfigur reagiert die Struktur wiederum insensitiv, siehe Abbildung 8.32, wobei der Betrag der zugeführten kinetischen Energie  $12.5 Nm$  ist.

#### 8.7.4 Gleichgewichtslage bei $50 kN$

Für die Gleichgewichtslage bei  $50 kN$  sollen nur noch die Ergebnisse für die erste Eigenform dargestellt werden, die Ergebnisse für die anderen Störformen sind prinzipiell gleich. Eine Störung mit der Energie  $W_{kin} = 0.5 Nm$ , die bei der Gleichgewichtslage bei  $120 kN$  zum Versagen führte, hat bei dieser Gleichgewichtslage erwartungsgemäß nur kleine Schwingungen um die Gleichgewichtslage zur Folge, siehe Abbildung 8.33.

Wird die Störung in der ersten Eigenform mit der hundertfachen Energie  $W_{kin} = 50 Nm$  eingebracht, so erhält man, wie aus Abbildung 8.34 ersichtlich, relativ große Auslenkungen

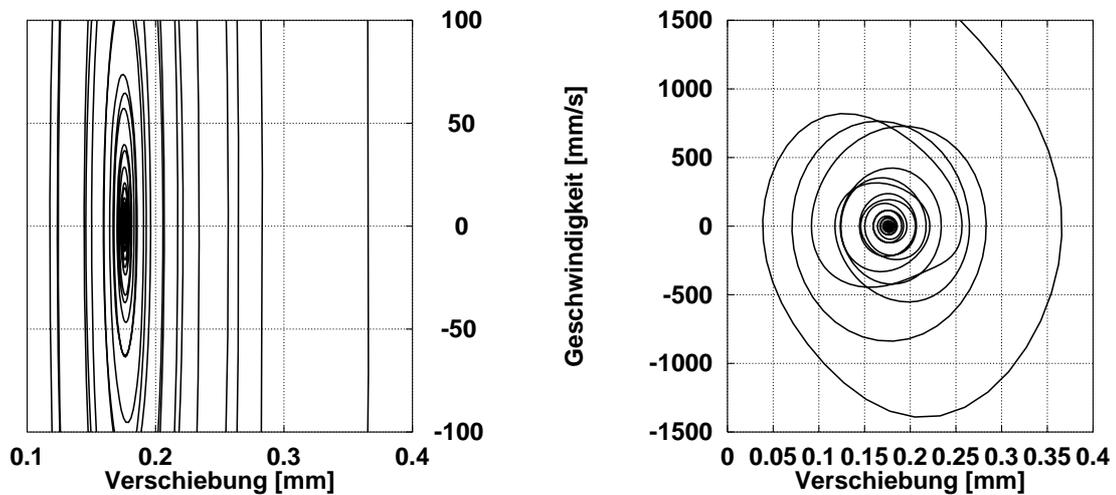


Abbildung 8.32: Phasendiagramm bei Störung mit Verformungsfigur,  $W_{kin} = 12.5 Nm$ , Last  $80 kN$

des oberen Zylinderrandes, jedoch kein Beulen. Zu Beachten ist dabei, daß bei der transienten Analyse in Abschnitt 8.6.1 das Beulen bereits bei einer Auslenkung des oberen Zylinderrandes von  $0.364 mm$  einsetzt. Dieser Wert wird hier weit überschritten, ohne daß die Struktur versagt. Auch kann aufgrund des statischen Residuums in Abbildung 8.35 keine Gleichgewichtslage bei großen Verschiebungen vermutet werden. Ein Versagen der Struktur kann erst mit einer unrealistisch hohen Energie von über  $W_{kin} = 2000 Nm$  erreicht werden. Für die Störung in der Endbeulform beträgt die Energie ca.  $W_{kin} = 400 Nm$ .

Diese Gleichgewichtslage verhält sich wie erwartet gegenüber den aufgebrauchten Störungen insensitiv. Unabhängig von der Anfangsgeschwindigkeitsverteilung ist es nur durch

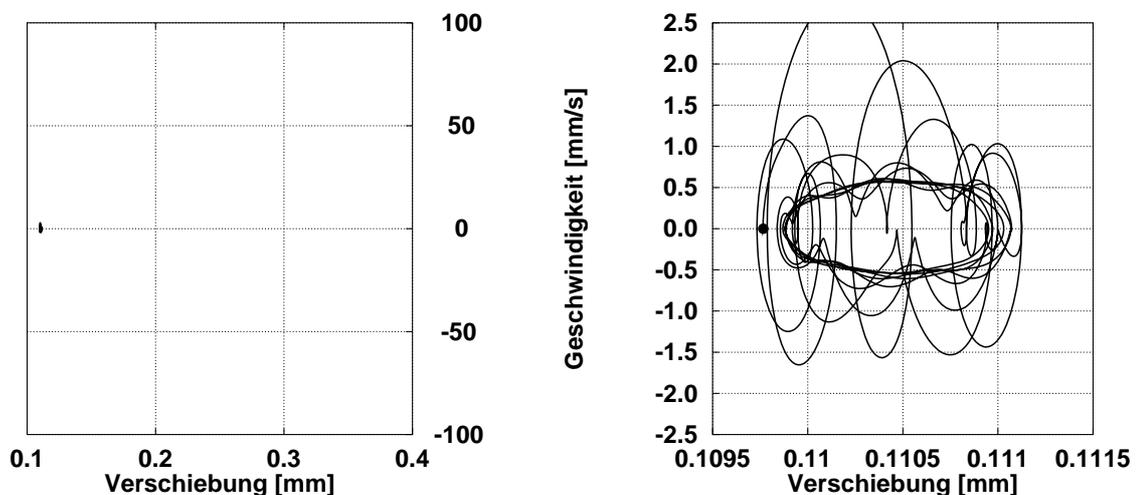


Abbildung 8.33: Phasendiagramm bei Störung mit erster Eigenform,  $W_{kin} = 0.5 Nm$ , Last  $50 kN$

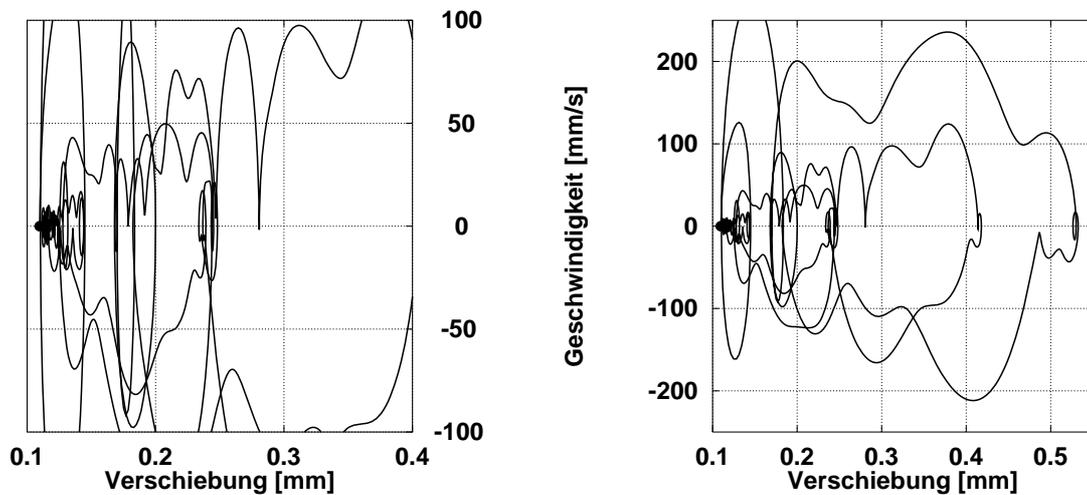


Abbildung 8.34: Phasendiagramm bei Störung mit erster Eigenform,  $W_{kin} = 50 Nm$ , Last  $50 kN$

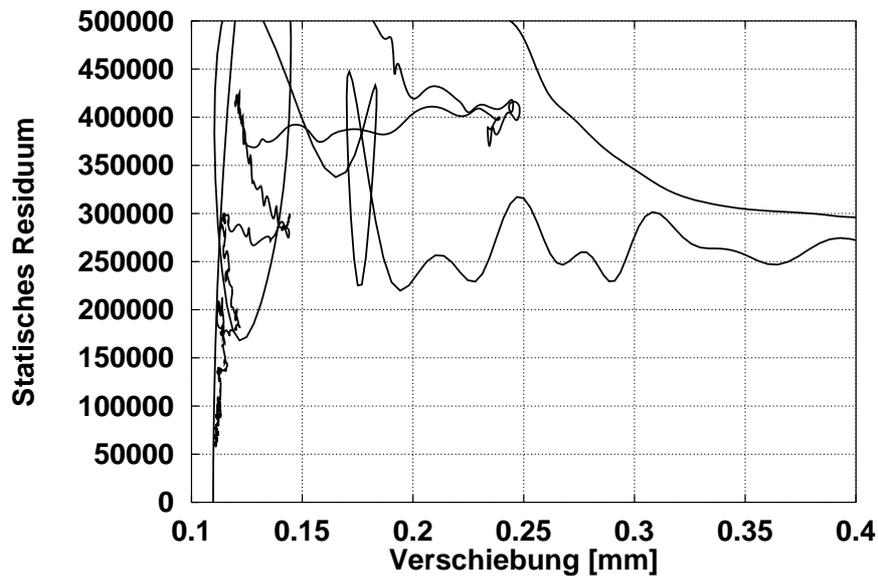


Abbildung 8.35: Statisches Residuum bei Störung mit erster Eigenform,  $W_{kin} = 50 Nm$ , Last  $50 kN$

sehr hohe kinetische Energien möglich, einen Beulvorgang zu initiieren. D.h. die Nachbeullast scheint die niedrigst mögliche Traglast zu sein und bis dahin liegt keine Stabilitätsgefährdung vor.

## 8.7.5 Sensitivität

In Abbildung 8.36 ist der Zusammenhang zwischen der Sensitivität und dem Lastniveau der stabilen Gleichgewichtslagen im Vorbeulbereich für die Störungen in der ersten Schwingungseigenform und der Endbeulform aus der transienten Berechnung dargestellt. Die berechneten Punkte sind durch Symbole gekennzeichnet.

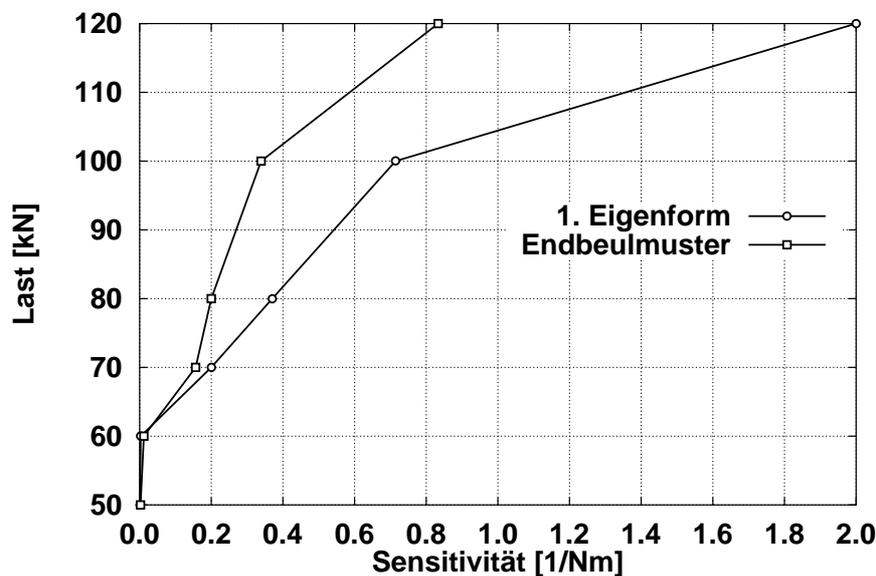


Abbildung 8.36: Sensitivität

Für Lasten in der Nähe der ersten Beullast ist eine große Sensitivität zu beobachten. Die Sensitivität für die erste Eigenform ist dabei deutlich höher als für die Endbeulform. Bei kleineren Lasten verringert sich die Sensitivität für beide Störungsformen. Bemerkenswert ist die Tatsache, daß generell mit der Anfangsgeschwindigkeitsverteilung in der ersten Schwingungseigenform die höchste Sensitivität ermittelt wird. Bei dem Lastniveau von  $60 \text{ kN}$  und kleiner ist die Sensitivität gegenüber der Beulform höher als für die erste Schwingungseigenform. Die dazugehörigen kinetischen Energien sind dabei aber sehr groß. Bei kleinen Lasten ist demnach auch die Sensitivität für beide Anfangsgeschwindigkeitsverteilungen gering. Dennoch muß beachtet werden, daß bei mäßig großen Lasten zwischen  $60 - 80 \text{ kN}$  die zum Beulen des Zylinders notwendigen Energien noch in der Größenordnung der inneren Energie der Struktur sind. Erst bei Lasten unterhalb der Nachbeullast, die mittels einer transienten Berechnung ermittelt wurde, nehmen diese Energien sehr hohe Werte im Bereich von mehreren hundert bzw. tausend  $Nm$  an, sodaß diese Gleichgewichtslagen als insensitiv bezeichnet werden können.

## 8.8 Untersuchungen an symmetrischen Zylinderausschnitten

Aus Effizienzgründen wird bei symmetrischen Strukturen oft angestrebt, an Stelle des vollständigen Systems nur ein Teilsystem unter Einarbeitung von entsprechenden Symmetriestandbedingungen zu berechnen. Durch diese Modellierung werden nur symmetrische Lösungen zugelassen, antisymmetrische oder unsymmetrische Lösungen ausgeschlossen. Auf diese Weise können aber nur perfekte Systeme behandelt werden, da im Allgemeinen Systeme mit geometrischen Imperfektionen nicht mehr symmetrisch sind. Werden dennoch bei geometrisch imperfekten Systemen lediglich Teilsysteme modelliert und mit Symmetriestandbedingungen versehen, so wird damit nicht nur die Lösung, sondern auch die Geometrie symmetrisiert. Die dann ermittelte Lösung kann bei imperfektionsempfindlichen mechanischen Strukturen von der Lösung des vollständigen, unsymmetrischen Systems stark abweichen. Der Einfluß dieser Modellierungsfrage soll in den folgenden Abschnitten am Beispiel des in der vorliegenden Arbeit betrachteten imperfekten Zylinders untersucht werden. Es werden zwei Zylinderhälften jeweils symmetrisch ergänzt und mit Symmetriestandbedingungen modelliert. Hiermit entstehen zwei neue Modelle von imperfekten Zylindern, deren Sollgeometrie zwar mit dem ursprünglichen Modell übereinstimmt, die jedoch beide unterschiedliche, symmetrische Imperfektionsmuster besitzen. Mit diesen beiden Modellen wird die transiente Berechnung des Beulvorganges analog zu Abschnitt 8.6.1 durchgeführt und die Sensitivität von jeweils zwei Gleichgewichtslagen exemplarisch bestimmt. Ebenso wird mit einem Zylinderviertel verfahren. Diese Ergebnisse werden anschließend mit den Resultaten des vollständigen Modells verglichen.

### 8.8.1 Untersuchungen an Zylinderhälften

Bei der ersten untersuchten Zylinderhälfte ist der vordere linke Teil der Struktur entfernt und durch Symmetriestandbedingungen ersetzt. In Abbildung 8.38<sub>1</sub> ist das Modell dargestellt. In diesem Modell ist die auffällige Delle im Imperfektionsmuster des ursprünglichen Zylinders (vgl. Abbildung 8.3) entfernt. Für den transienten Beulvorgang ist das Last-Verschiebungsdiagramm in Abbildung 8.37 angegeben. Die berechnete Beullast beträgt 185 kN und liegt damit ca. 10% über der Beullast des vollständigen Systems. Dies ist eine Folge der Imperfektionsempfindlichkeit des untersuchten axial belasteten Zylinders. Während beim vollständigen System der Beulvorgang von der Delle im Imperfektionsmuster ausgeht, beginnt der Beulvorgang bei der ersten Zylinderhälfte an einer anderen Stelle, siehe Abbildung 8.38<sub>1</sub>. Das Beulmuster ist zunächst deutlich hochwelliger als beim vollständigen System. Im Unterschied zu der Endbeulfigur in Abbildung 8.13<sub>6</sub> befindet sich das zweireihige Endbeulmuster der ersten Zylinderhälfte weiter am unteren Zylinderrand. Die Anzahl der Beulen in Umfangsrichtung ist aber gleich. Ebenso stimmt die Nachbeullast mit ca. 50 kN gut überein.

Bei der zweiten Zylinderhälfte ist die rechte hintere Hälfte entfernt und durch Symmetriestandbedingungen ersetzt, siehe Abbildung 8.40<sub>1</sub>. Die Last-Verschiebungskurve ist in Abbildung 8.37 dargestellt. Die Beullast beträgt 169 kN und weicht nur um ca. 2% von der Beullast des vollständigen Systems ab. Auch der Beulvorgang der symmetrisch ergänzten

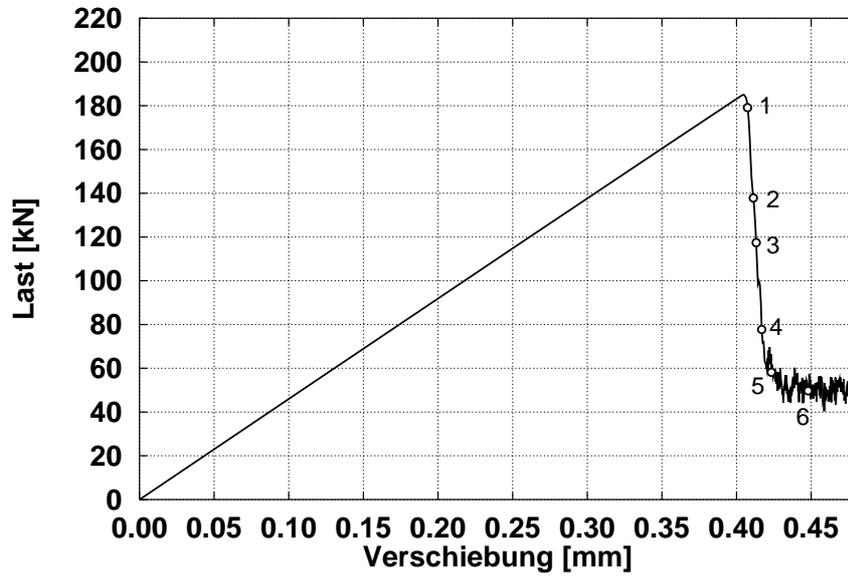


Abbildung 8.37: Last-Verformungskurve Zylinderhälfte 1, transiente Belastung

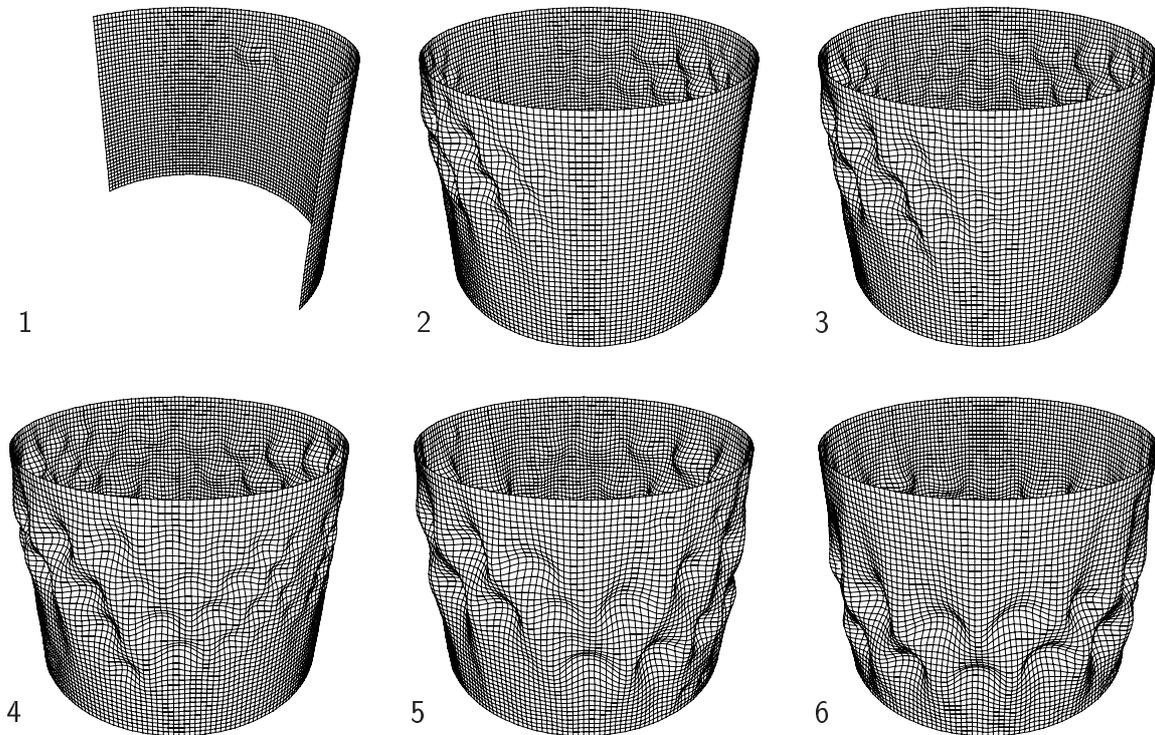


Abbildung 8.38: Verformungsfiguren einer Zylinderhälfte 1, transiente Belastung, 10-fach überhöht

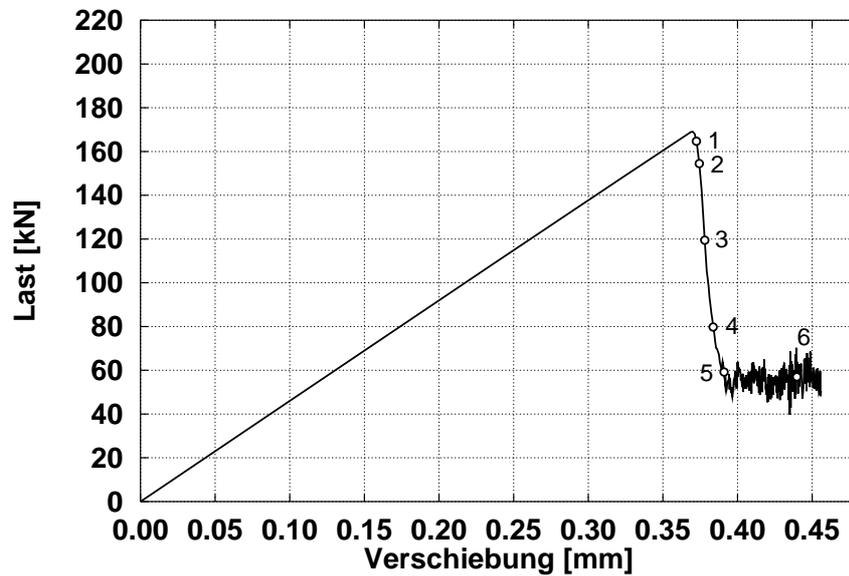


Abbildung 8.39: Last-Verformungskurve Zylinderhälfte 2, transiente Belastung

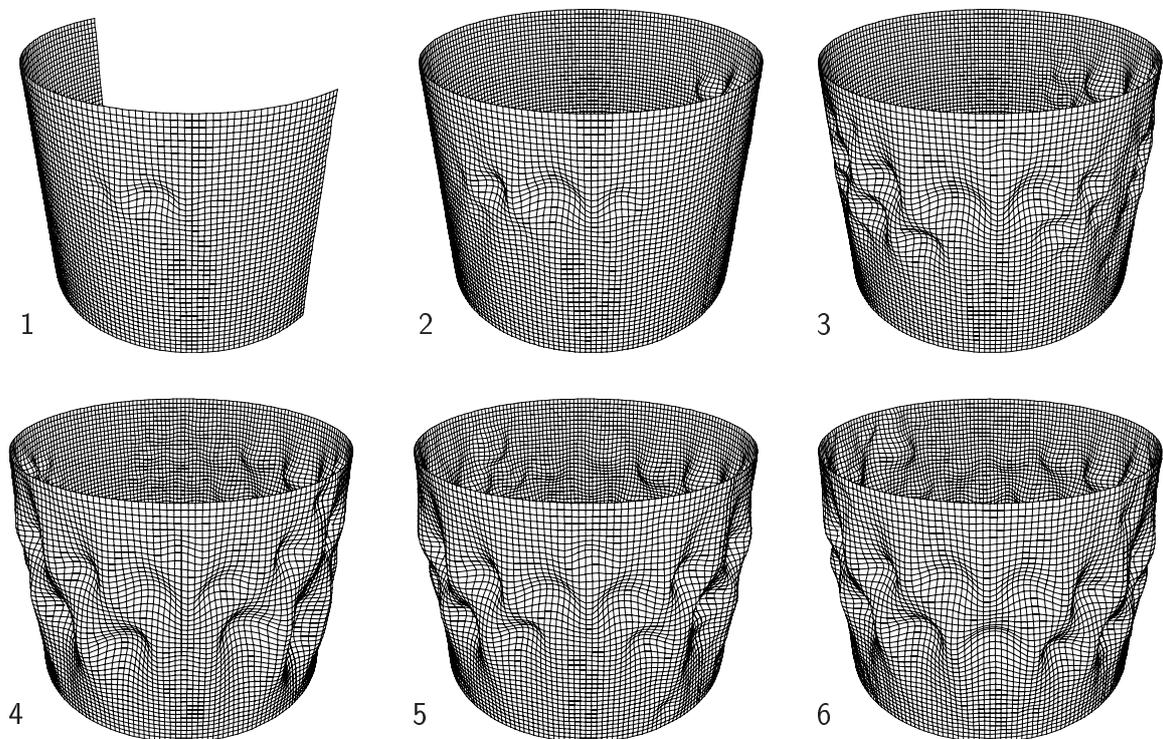


Abbildung 8.40: Verformungsfiguren einer Zylinderhälfte 2, transiente Belastung, 10-fach überhöht

zweiten Zylinderhälfte entspricht weit mehr dem der vollständigen Struktur als bei der ersten Zylinderhälfte, da er ebenfalls durch die Delle im Imperfektionsmuster geprägt ist. Die ermittelte Nachbeulfigur ist ähnlich zum vollständigen Zylinder zweireihig und in Zylindermitte plaziert. Es befinden sich ebenfalls 16 Beulen in Umfangsrichtung. Aufgrund der erzwungenen Symmetrie ist die Nachbeulfigur der zweiten Zylinderhälfte jedoch gegenüber der Figur der vollständigen Struktur gedreht. Auch hier liegt die Nachbeullast wieder bei ca.  $50 \text{ kN}$ .

### 8.8.2 Untersuchungen an einem Zylinderviertel

Eine weitere Verringerung des numerischen Aufwandes gegenüber der Berechnung von Zylinderhälften kann durch die Behandlung von Zylindervierteln erzielt werden. Hierzu wird ein nur ein mögliches Viertel exemplarisch behandelt, siehe Abbildung 8.42<sub>1</sub>. Die Last-Verschiebungskurve zeigt Abbildung 8.41. Die Beullast beträgt  $206 \text{ kN}$  und liegt somit deutlich höher als bei den Zylinderhälften und beim vollständigen Zylinder. Der Unterschied zum vollständigen Zylinder beträgt 25%. Auch bei diesem System ergibt sich wieder eine nahezu konstante Nachbeullast, die wiederum ca.  $50 \text{ kN}$  beträgt.

Bei den Beulfiguren des Viertelsystems sind deutliche Unterschiede zu den vorher berechneten Zylindermodellen erkennbar. Bei dem untersuchten Zylinderviertel startet der Beulvorgang am oberen Rand mit kleinen Beulen, die sich um den Zylinderumfang schließen, Abbildung 8.42<sub>1-3</sub>. Dieses hochwellige Beulmuster breitet sich im Beulvorgang zum unteren Zylinderrand hin aus, Abbildung 8.42<sub>3-6</sub>. Dabei findet ein Übergang zu einem zweireihigen Beulmuster mit jeweils 16 Beulen in Umfangsrichtung statt. Dieses Beulmuster ist, wie schon das der Zylinderhälfte 1, am unteren Zylinderrand plaziert. Für die nachfolgende Sensitivitätsberechnung wird das Muster in Abbildung 8.42<sub>6</sub> gewählt.

Für die transiente Berechnung des Zylinderviertels wurde die Geschwindigkeit des oberen Zylinderrandes mit  $v = 0.011 \text{ mm/s}$  vorgegeben. Bei der Berechnung aller anderen Modelle beträgt diese Geschwindigkeit  $v = 0.01 \text{ mm/s}$ . Bei der niedrigeren Geschwindigkeitsvorgabe ergibt sich für das betrachtete Zylinderviertel eine negative Last im Nachbeulbereich. D.h. der Zylinder versucht, den oberen Zylinderrand nach unten zu ziehen, bzw. schneller zu kollabieren, als es die Verschiebungsvorgabe zulässt. Im physikalischen Experiment würde sich also der Zylinder von der Lastplatte lösen. Aufgrund der Eigensteifigkeit der Prüfapparatur gelänge ihm das jedoch nicht, die Lastplatte bzw. der Hydraulikzylinder würde nachfedern. Durch die leichte Erhöhung der vorgegebenen Geschwindigkeit des oberen Zylinderrandes kann die durchgängige Belastung ohne Ablösen des Zylinders modelliert werden. Auch bei dieser leichten Erhöhung bleibt der Belastungsvorgang im Vorbeulbereich quasistatisch.

### 8.8.3 Vergleich der Sensitivität

Für die symmetrischen Zylinderhälften 1 und 2 und den Viertelzylinder aus den vorangegangenen Abschnitten wird die Sensitivität an den Gleichgewichtslagen bei  $80 \text{ kN}$  und

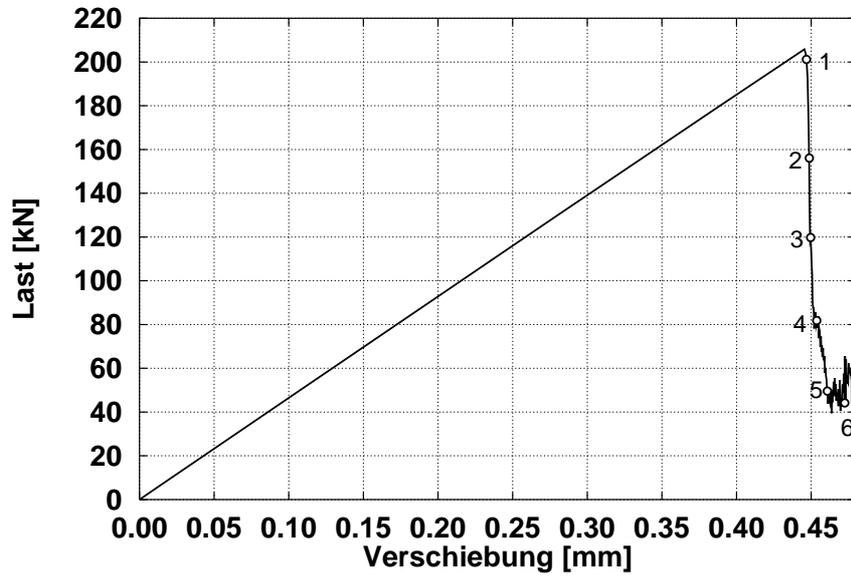


Abbildung 8.41: Last-Verformungskurve Zylinderviertel, transiente Belastung

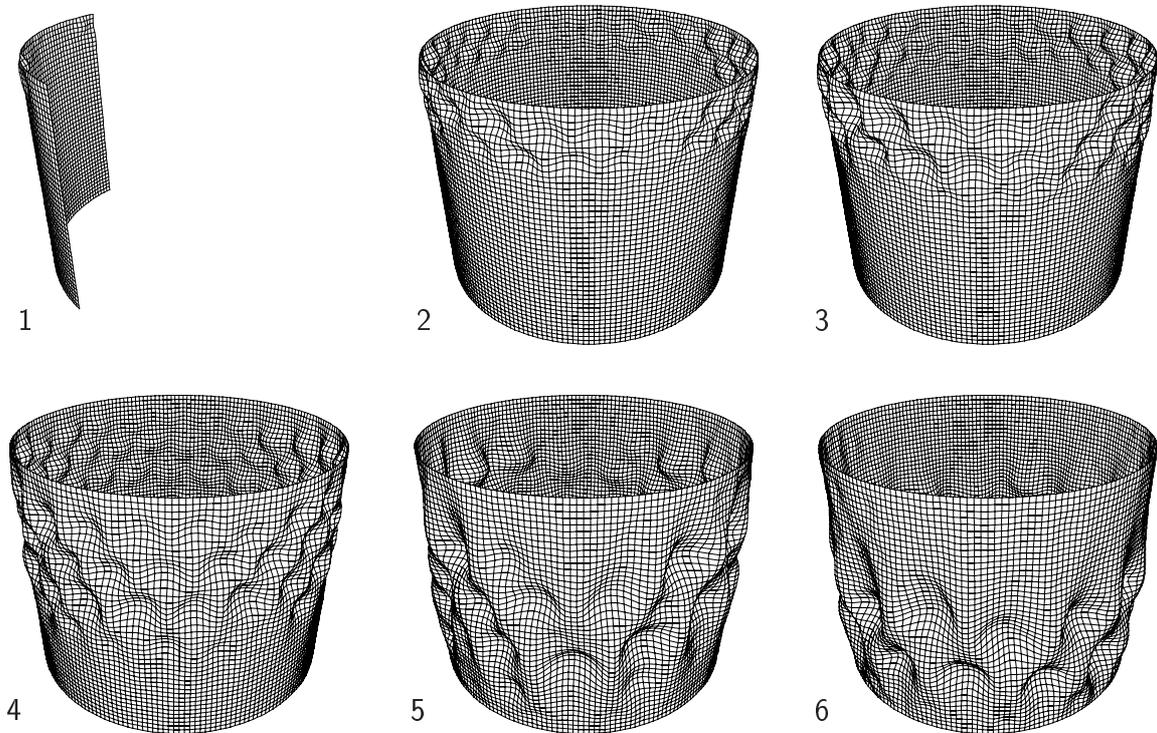


Abbildung 8.42: Verformungsfiguren eines Zylinderviertels, transiente Belastung, 10-fach überhöht

	Last	Sensitivität der Zylindermodelle [ $1/Nm$ ]			
		Vollständiges System	Zylinderhälfte 1	Zylinderhälfte 2	Zylinderviertel
1. Schwingungseigenform	120 $kN$	2.00	1.32	2.00	2.00
	80 $kN$	0.37	0.28	0.31	0.19
Endbeulmuster	120 $kN$	0.83	0.65	0.84	0.49
	80 $kN$	0.20	0.19	0.17	0.17

Tabelle 8.1: Sensitivität der Zylinderhälften 1 und 2 und des Zylinderviertels

120  $kN$  bestimmt. Als Anfangsgeschwindigkeitsverteilungen werden die erste Schwingungseigenform und das Endbeulmuster aus der transienten Berechnung verwendet. Die so erzielten Ergebnisse sind in Tabelle 8.1 angegeben, zum Vergleich sind die Werte für das vollständige System ebenfalls enthalten.

Für alle symmetrischen Teilsysteme verhält sich die Gleichgewichtslage bei 120  $kN$  und bei 80  $kN$  sensitiver gegenüber der ersten Schwingungseigenform. Bei beiden untersuchten Anfangsgeschwindigkeitsverteilungen verringert sich die Sensitivität für Gleichgewichtslagen mit kleinerer Last.

Die Sensitivitäten bezüglich der Anfangsgeschwindigkeitsverteilung in der ersten Schwingungseigenform sind für alle Modelle bei beiden Laststufen ähnlich. Bei der Zylinderhälfte 1 ergibt sich beim Lastniveau von 120  $kN$  eine geringere Sensitivität. Dies ist nicht unerwartet, da der Beulpunkt dieser Zylinderhälfte gegenüber der anderen Zylinderhälfte und dem vollständigen System deutlich erhöht ist und die untersuchte Gleichgewichtslage bei 120  $kN$  dadurch vom Beulpunkt weiter entfernt ist, als bei den anderen Modellen. Da auch das untersuchte Zylinderviertel eine deutlich erhöhte Beullast besitzt, wäre ein geringerer Wert als 2.00 für die Sensitivität beim Lastniveau von 120  $kN$  zu erwarten gewesen. Bei der Betrachtung der ersten Eigenform der unterschiedlichen Modelle zeigt sich aber ein Unterschied in der Eigenform des Viertelzylinders, bei dem sich 12 Längsbeulen in Umfangsrichtung befinden, während es bei den anderen Systemen 13 Beulen sind. Die bei dem vollständigen System und den Zylinderhälften angeregte Schwingungsform ist bei dem Zylinderviertel aus Symmetriegründen nicht möglich. Auch die abweichende Sensitivität gegenüber der ersten Schwingungseigenform bei dem Lastniveau von 80  $kN$  ist darauf zurückzuführen.

Eine ähnliche Situation ergibt sich für die Anfangsgeschwindigkeitsverteilung in der Nachbeulfigur. Die Sensitivitäten für den vollständigen Zylinder und die Zylinderhälfte 2 stimmen sehr gut überein. Eine Ausnahme stellt aus den oben geschilderten Gründen wieder die Zylinderhälfte 1 und das Zylinderviertel beim Lastniveau von 120  $kN$  dar. Beim niedrigeren Lastniveau bei 80  $kN$ , d.h. in größerer Entfernung vom Beulpunkt, stimmt die Sensitivität für alle Strukturen gut überein.

#### 8.8.4 Diskussion der Modellierung mit Zylinderausschnitten

Zwei wesentliche Unterschiede gehen mit der Modellierung des untersuchten Zylinders als Zylinderausschnitte einher. Zunächst werden die möglichen Lösungen auf symmetrische Lösungen beschränkt. Durch die Symmetrierandbedingungen wird das System symmetrisch vervollständigt, was insbesondere für das Imperfektionsmuster gilt. Deshalb ist zu erwarten, daß die Resultate für diese imperfektionsempfindliche Struktur deutliche Unterschiede zu den Ergebnissen des vollständigen Systems aufweisen. In einiger Entfernung von den singulären Punkten sollten sich die unterschiedlichen Lösungen aber wieder asymptotisch annähern, wie z.B. aus Abbildung 7.2 ersichtlich. Voraussetzung hierzu ist jedoch, daß es sich in beiden Fällen um den selben Lösungsast handelt, was aufgrund der Einschränkung der Lösungsmöglichkeiten nicht notwendigerweise gegeben sein muß.

Wie bereits in Abschnitt 8.4 geschildert, ist die Beullast stark von den Imperfektionen abhängig. Während dort nur die Amplitude des Imperfektionsmusters variiert wurde, ändert sich bei der Modellierung mit Zylinderausschnitten das Imperfektionsmuster selbst. Der Beginn des Beulvorgangs des vollständigen Zylinders ist wesentlich durch eine "Delle" im Imperfektionsmuster geprägt. Diese "Delle" ist in der Zylinderhälfte 2 enthalten. Für den vollständigen Zylinder und die Zylinderhälfte 2 wird dann auch eine fast übereinstimmende Beullast berechnet. Bei den beiden anderen untersuchten Zylinderausschnitten, der Zylinderhälfte 1 und dem Zylinderviertel, werden hingegen davon deutlich abweichende Beullasten ermittelt.

Die Nachbeullasten, d.h. die Lösung in größerer Entfernung des singulären Punktes, zeigen eine große Übereinstimmung zwischen den Zylinderausschnitten und dem vollständigen System. Auch die Nachbeulmuster haben wesentliche gemeinsame Charakteristika. So sind die Beulfiguren jeweils zweireihig mit je 16 Beulen in Umfangsrichtung. Bei der Zylinderhälfte 1 und dem Zylinderviertel befindet sich dieses Muster jedoch weiter am unteren Zylinderrand. Da es sich bei den beiden Zylinderhälften um das gleiche Modell handelt, das sich nur durch das Imperfektionsmuster unterscheidet, ist das gleiche Nachbeulmuster möglich. Aufgrund der unterschiedlichen Nachbeulfiguren handelt es sich aber um unterschiedliche Lösungsäste. Offensichtlich ist dies eine Folge des unterschiedlichen Beulvorganges aufgrund des geänderten Imperfektionsmusters.

Die für die Zylinderausschnitte an zwei Laststufen exemplarisch bestimmten Werte für die Sensitivität stimmen sehr gut mit denen des vollständigen Zylinders überein. Die auftretenden Unterschiede sind bei der Zylinderhälfte 1 durch die feste Wahl der Laststufe bei  $120\text{ kN}$  bedingt, die sich bei diesem System in größerer Entfernung von der Beullast befindet als bei den anderen Systemen. Für die Unterschiede ist hier also ausschließlich das geänderte Imperfektionsmuster ausschlaggebend, die Verschiebungen können durch die Zylinderhälften noch ausreichend abgebildet werden. Die ermittelten Werte für das Zylinderviertel stimmen nur teilweise mit dem vollständigen Zylinder überein. Bei diesem Modell machen sich die Einschränkungen der Verschiebungsmöglichkeiten durch die Symmetrierandbedingungen bemerkbar. So wurde z.B. die als Anfangsgeschwindigkeitsverteilung genutzte erste Schwingungseigenform gegenüber den anderen Systemen geändert. Die Sensitivität gegenüber dieser Anfangsgeschwindigkeitsverteilung ist daher nicht direkt

mit den anderen Teilsystemen vergleichbar.

Insgesamt läßt sich feststellen, daß mit der Modellierung des Zylinders durch mit Symmetriestandbedingungen versehenen Zylinderausschnitte grundsätzlich das Systemverhalten gut abgebildet werden kann. Die damit erzielten Ergebnisse bezüglich Nachbeullasten, Beulmuster und Sensitivität zeigen eine gute Übereinstimmung mit dem vollständigen System. Es ist aber zu beachten, daß neben den Lösungen auch das Imperfektionsmuster symmetrisiert wird und daher imperfektionsempfindliche Größen, wie z.B. die Beullast, nur unzureichend wiedergegeben werden. Bei einer starken Einschränkung der Verschiebungen, wie z.B. beim Zylinderviertel, zeigen sich aber auch qualitative Unterschiede, die nicht allein auf das geänderte Imperfektionsmuster zurückführbar sind. Dies ist im Einzelfall nachzuprüfen.

## 8.9 Zusammenfassung und Wertung der Resultate

Obgleich prinzipiell möglich [28, 38], führen statische Stabilitätsuntersuchungen an dem betrachteten imperfekten Zylinder nicht zu einer erfolgreichen Berechnung des Nachbeulverhaltens. Der Grund hierfür liegt einerseits bei der numerischen Pfadverfolgung, die wegen des komplizierten Nachbeulverhaltens des Zylinders mit vielen Lastwechseln und lokalem Beulen nicht zuverlässig automatisch gesteuert werden kann. Andererseits machen Konvergenzprobleme des Newton-Raphson Verfahrens im Bereich der vielen singulären Stellen die Bestimmung des Lösungspfades bis in den Nachbeulbereich praktisch undurchführbar.

Transiente Analysen, die das dynamische Verhalten der Struktur wiedergeben, erlauben hingegen, den Beulvorgang bis in den Nachbeulbereich zu bestimmen. Die ermittelte Nachbeullast steht mit Experimenten und Bemessungsvorschriften sehr gut in Einklang. Auch das berechnete Nachbeulmuster bildet wesentliche Charakteristika des Versuchs ab. Allerdings sei auf Untersuchungen in [120] hingewiesen, wonach unterschiedliche Anfangsbedingungen beim Beulbeginn zu unterschiedlichen Nachbeulmustern führen können, was sich aber bei den betrachteten Zylindern nicht bewahrheitete. Die beschriebenen Simulationen können sehr effizient auch mit iterativen Lösern parallel durchgeführt werden, da die Koeffizientenmatrix, die bei der Verwendung des Newmark Verfahrens entsteht, bei kleinen Zeitschritten gut konditioniert ist. Nachteilig bei dieser Vorgehensweise ist, daß durch den Lösungsprozess keine Information über die tangentielle Steifigkeitsmatrix geliefert wird. Ebenso sind keine effizient nutzbaren Kriterien zur Bestimmung der Stabilität der Bewegung im Rahmen der Methode der Finiten Elemente verfügbar.

Zur Untersuchung der Sensitivität von Gleichgewichtslagen im Vorbeulbereich wurde dem System an verschiedenen Laststufen kinetische Energie in unterschiedlicher Form eingeprägt. Dadurch konnte der Zylinder bei unterschiedlichem Lastniveau, das über der berechneten Nachbeullast lag, zum Beulen angeregt werden. Unterhalb des Lastniveaus der Nachbeullast gelang dies nur mit sehr großen, für die Praxis irrelevanten Energien, woraus geschlossen werden kann, daß die berechnete Nachbeullast für diese Struktur tatsächlich eine charakteristische Größe darstellt, unterhalb der nur eine geringe Beulgefahr besteht.

Für Lasten ab ca.  $70 \text{ kN}$  ist der Zylinder am sensitivsten gegenüber einer Anfangsgeschwindigkeitsverteilung in der ersten Schwingungseigenform. Darunter ist eine höhere Sensitivität für die Endbeulform aus der transienten Berechnung festzustellen. Mit dieser Anregung kann direkt in die Nachbeulform übergegangen werden. Allerdings sind die bei diesem niedrigen Lastniveau erforderlichen kinetischen Energien für beide Anfangsgeschwindigkeitsverteilungen sehr hoch, verglichen mit der inneren Energie der Struktur. Zur Bestimmung der Sensitivität ist daher eine Beschränkung auf die erste Schwingungseigenform in weiten Bereichen zulässig. Erst bei sehr niedrigen Lasten erweist sich diese Anfangsgeschwindigkeitsverteilung nicht mehr als die kritischste, und andere, auch praktisch relevante Anfangsgeschwindigkeitsverteilungen müssen hier gleichfalls getestet werden. Zu Bedenken ist allerdings, daß viele beliebige Geschwindigkeitsverteilungen das System breitbandig anregen, und daher auch die hier betrachteten Störformen enthalten sind; dies gilt insbesondere für die erste Eigenform.

Die bei diesem Vorgehen bestimmte Energie, die zum Versagen des Zylinders notwendig ist, steigt von  $0.5 \text{ Nm}$  bei einem Lastniveau von  $120 \text{ kN}$  auf den zehnfachen Wert bei  $80 \text{ kN}$  an, was einen deutlichen Rückgang der Sensitivität der Gleichgewichtslage dokumentiert. Dennoch überrascht dieser niedrige Wert der Energie, gemessen an der inneren Energie des Zylinders. Auch Gleichgewichtslagen, die weit von der kritischen Beullast entfernt sind, sind wie aus der Praxis des Beulens bekannt, sensitiv.

Die durch die aufgebrachten Störungen ausgelösten Beulvorgänge enden zwar ausnahmslos in Rautenbeulmustern, wie sie aus der experimentellen Praxis bekannt sind, die Entwicklung des Beulmusters verläuft jedoch unterschiedlich. Zum Teil entwickelt sich das Beulmuster von einer lokalen Beule ausgehend, wie bei der transienten Belastung, andererseits bildet sich bei anderen Störungen unmittelbar großflächig ein vollständiges Beulmuster aus.

Für den untersuchten Zylinder mit imperfekter Geometrie zeigt die Betrachtung des statischen Residuums, mit dem es bei einfachen mechanischen Strukturen gelingt, benachbarte Gleichgewichtslagen aufzufinden, kein Minimum bei einem Verformungszustand an. Es ist also nicht gelungen, hiermit einen der vielen möglichen statischen Nachbeulpfade anzusteuern. Selbst bei der Kenntnis einer weiteren Gleichgewichtslage und der Störung mit einer dieser ähnlichen Form kann die Gleichgewichtslage nicht erkannt werden. Für realistische, große Strukturen scheint das Vorgehen mit Residuumberechnung und Dämpfung der Schwingung demnach nicht geeignet, um benachbarte Gleichgewichtslagen zu bestimmen. Die transiente Analyse mit sinnvoller allgemeiner Imperfektionsform scheint aber eine für die Ingenieurpraxis sinnvolle Alternative zu bieten.

## 9 Zusammenfassung und Ausblick

Das Ziel der vorliegenden Arbeit lag in der Untersuchung der Stabilität und Sensitivität von Gleichgewichtslagen mechanischer Strukturen mittels der Methode der Finiten Elemente. Im Hinblick auf reale, komplexe Strukturen, die zu mechanischen Modellen mit vielen Freiheitsgraden führen, wurde dabei besonderer Wert auf eine effiziente numerische Behandlung gelegt. Den wichtigsten Aspekt stellte dabei die Auflösung der bei der Berechnung des nichtlinearen Strukturverhaltens mit Hilfe der FE-Methode entstehenden linearen Gleichungssysteme dar. Hierzu wurde zunächst die Einsatzmöglichkeit direkter und iterativer Lösertypen geprüft. Das verwendete Finite Element Programm wurde unter Beschränkung auf iterative Krylov-Unterraum Verfahren zur linearen Gleichungslösung parallelisiert. Am konkreten mechanischen System eines axial belasteten Stahlzylinders wurden Untersuchungen bezüglich Stabilität und Sensitivität unter Nutzung von Parallelrechnern durchgeführt und ausführlich diskutiert. Im Folgenden werden die wesentlichen Ergebnisse kurz zusammengefaßt.

Die Vergleiche der unterschiedlichen Lösertypen ergaben, daß sich iterative Krylov-Unterraum Verfahren grundsätzlich auch zur Lösung von Versagens- und Stabilitätsproblemen dünner Schalenstrukturen eignen. Ihnen ist vor allem ein niedriger Speicherbedarf zueigen. Dies machte iterative Löser insbesondere dann attraktiv, wenn sehr große Probleme zu lösen waren, oder Rechner mit nur sehr kleinem Hauptspeicher zur Verfügung standen. In diesen Fällen zeichneten sich iterative Löser meist auch durch kürzere Rechenzeiten aus. Direkte Lösungsverfahren waren hingegen immer dann effizient einzusetzen, wenn mit einer Koeffizientenmatrix mehrere rechte Seiten gelöst werden mußten, wie es z.B. beim Bogenlängenverfahren der Fall ist. Auch beim Einsatz von Vektorrechnern erwiesen sich die gut vektorisierbaren direkten Algorithmen als vorteilhaft. Bei sehr schlecht konditionierten Gleichungssystemen, wie sie z.B. bei der nichtlinearen statischen Analyse von axial belasteten Zylindern entstehen, konvergierten die iterativen Löser nicht mehr so gut. Direkte Löser waren dann meist vorzuziehen.

Eine wichtige Fragestellung beim Einsatz iterativer Löser liegt in der Wahl einer geeigneten Vorkonditionierung zur Konvergenzbeschleunigung. Dies wird umso wichtiger, je schlechter die zu lösenden Probleme konditioniert sind. In der vorliegenden Arbeit wurden rein algebraische Vorkonditionierungsstrategien eingesetzt, da diese für alle Problemklassen ohne Einschränkung nutzbar sind. Hierarchische und Mehrgitterstrategien fanden trotz der bekannten Vorteile keine Berücksichtigung. Es zeigte sich, daß für 3D Kontinuumsprobleme eine einfache Jacobi-Vorkonditionierung bereits sehr gute Ergebnisse lieferte, diese hingegen bei dünnen Schalenproblemen nicht mehr ausreichend war. Varianten der unvollständigen Cholesky Zerlegung lieferten in den meisten Fällen zufriedenstellende Ergebnisse. Insbesondere die Vernachlässigung von Koppeleinträgen unterschiedlicher Knotenfreiheitsgrade in der Steifigkeitsmatrix und anschließender unvollständiger Faktorisierung führte in vielen Fällen zu effizienten iterativen Lösungen. Dies gilt für 3D Kontinuumsprobleme ebenso wie für Schalenprobleme. Ein weiterer Aspekt war die Permutation der Steifigkeitsmatrix vor der Berechnung einer unvollständigen Zerlegung zur Vorkonditionierung. Bei Steifigkeitsmatrizen mit großer Bandbreite, wie sie z.B. durch automatische

Netzgeneratoren oder bei adaptiver Verfeinerung entstehen, ist eine Band- bzw. Profilotimierung von großem Vorteil. Bei allen nichtlinearen Problemen, die im Verlauf der vorliegenden Arbeit berechnet wurden, stellte sich aber die Kombination von direkten und iterativen Verfahren als die mit Abstand effizienteste Lösungsstrategie heraus. Dazu wurde in jeder Newton Iteration die Steifigkeitsmatrix zunächst faktorisiert, also eine direkte Lösung berechnet, und anschließend diese faktorisierte Matrix zur Vorkonditionierung der nachfolgend zu lösenden linearen Gleichungssysteme eingesetzt. Eine Halbierung der Gesamtrechenzeit durch den Einsatz dieses Verfahrens war dabei nicht ungewöhnlich. Allerdings wird hierfür auch mehr Speicherplatz benötigt, als zur rein direkten Lösung.

Eine weitere Verkürzung der Rechenzeit wird durch den Einsatz von Parallelrechnern ermöglicht. Im Rahmen der vorliegenden Arbeit wurde das Finite Elemente Programm FEAP für statische und dynamische nichtlineare Analysen ohne Beschränkung auf spezielle Elementtypen für ausschließliche Verwendung iterativer Lösungsstrategien parallelisiert. Dabei findet vor einer parallelen Bearbeitung als sequentieller Schritt zunächst eine Gebietszerlegung statt. Hierbei wird jedem Prozessor ein Teilgebiet des Gesamtproblems zur Bearbeitung zugewiesen und außerdem erhält er Informationen über benachbarte Teilgebiete. Zur Gebietszerlegung finden allgemein erhältliche Algorithmen Verwendung, die gute Aufteilungen mit gleich großen Teilgebieten und kleinen Rändern zwischen den Teilgebieten liefern. Eine einmal zerlegte Struktur kann anschließend für verschiedene parallele Berechnungen genutzt werden. Ein dynamischer Lastausgleich findet während der parallelen Berechnung nicht statt, da dies wegen des erheblichen Aufwandes zur parallelen Berechnung der Neuverteilung und zur Migration der Elemente nur bei adaptiven Anwendungen sinnvoll ist, die hier nicht behandelt wurden. Die mit der parallelen Programmversion erzielten Verkürzungen der Rechenzeit, gemessen mit dem sogenannten *speedup*, sind ausgezeichnet. Vergleicht man die Rechenzeiten mit dem äquivalenten sequentiellen Lösungsverfahren, so stellt sich ein nahezu optimaler inkrementeller *speedup* nahe 2 bei der Verdoppelung der Anzahl der verwendeten Prozessoren ein. Ein Vergleich mit dem schnellsten verfügbaren sequentiellen Lösungsverfahren fällt allerdings für sehr schlecht konditionierte Probleme nicht so gut aus, da hier iterative Lösungsverfahren generell gegenüber direkten Lösern weniger vorteilhaft sind.

Die Beurteilung der Stabilität einer Gleichgewichtslage eines konservativen Systems ist mit der Methode der Finiten Elemente in einfacher Weise möglich. Auch das Nachbeul- und Verzweigungsverhalten kann meist, verbunden mit einigem numerischem Aufwand, bestimmt werden. Bei der praktischen Berechnung realer Systeme, wie z.B. bei dem in der vorliegenden Arbeit untersuchten axial belasteten, geometrisch imperfekten Kreiszyylinder, erweist sich die Vorgehensweise der numerischen Pfadverfolgung allerdings nicht immer als erfolgreich, wenn der numerische Aufwand in sinnvollem Rahmen gehalten werden soll. Zum Einen treten bei komplexem Nachbeulverhalten Konvergenzprobleme des lokal konvergenten Newton Verfahrens im Bereich nahezu singulärer Tangentensteifigkeitsmatrizen auf, zum Anderen wird oft eine manuelle Steuerung des Kurvenverfolgungsalgorithmus notwendig. Das Auftreten von Verzweigungspunkten zieht die Notwendigkeit gesonderter numerischer Behandlung und anschließender Verfolgung auch der zusätzlichen Lösungspfade nach sich. Insgesamt läßt sich feststellen, daß diese rein statische Vorgehensweise für die Untersuchung realitätsnaher Systeme mit vielen Freiheitsgraden kaum

geeignet ist.

Demgegenüber steht die Möglichkeit, das Beulverhalten nicht durch statische, sondern mittels transienter Simulationen, d.h. unter Berücksichtigung der Trägheitskräfte, zu beschreiben. Dadurch wird es möglich, mit moderatem numerischem Aufwand das reale Beulverhalten der Struktur zu berechnen. Für den konkret untersuchten axial belasteten Kreiszyylinder konnte mittels der transienten Berechnung des Beulvorganges das Verhalten der Struktur bis in den Nachbeulbereich ermittelt werden, was durch rein statische Analysen nicht gelang. Die sich einstellende, nahezu konstante Nachbeullast stimmte sehr gut mit dem experimentell ermittelten Wert überein. Der Vorbeulbereich war bei der transienten Berechnung durch die langsame Lastaufbringung identisch mit der statischen Berechnung. Der Beulvorgang selbst wurde stark von den Imperfektionen der Struktur geprägt, wie Untersuchungen an symmetrisch ergänzten Teilstrukturen ergaben. Das sich einstellende Nachbeulmuster hingegen blieb auch bei den modifizierten Strukturen in seinen wesentlichen Charakteristika unverändert.

Mindestens ebenso wichtig wie das Nachbeulverhalten einer Struktur ist die Sensitivität der Gleichgewichtslagen im Vorbeulbereich. Insbesondere bei Bauteilen, die den ausgebeulten Zustand nie erreichen sollen, wie z.B. bei dem betrachteten Siloschuß, ist dies eine zentrale Größe beim Entwurf. Die Sensitivität wurde definiert als der reziproke Wert der kinetischen Energie, die dem System mindestens eingeprägt werden muß, um ausgehend von einer stabilen Gleichgewichtslage eine zweite stabile Gleichgewichtslage oder unbegrenzt anwachsende Verschiebungen zu erreichen. Für den betrachteten Zylinder mit imperfekter Geometrie zeigte sich nahe der Beullast erwartungsgemäß eine hohe Sensitivität. Das Einprägen kinetischer Energie in Form der ersten Schwingungseigenform durch die Vorgabe einer Anfangsgeschwindigkeitsverteilung in der statischen Gleichgewichtslage führte bereits für geringe Energien zu einem Beulen der Struktur. Bei einer anderen Gleichgewichtslage mit niedrigerem Lastniveau war hingegen ein Vielfaches der Energie zum Auslösen des Beulvorganges notwendig, die Sensitivität der letzteren Gleichgewichtslage war also geringer. Dennoch war die Energie, die zur Auslösung des Beulvorganges notwendig ist, klein verglichen mit der inneren Energie der Struktur, sodaß auch die Gleichgewichtslage mit niedriger Last nicht als sicher betrachtet werden kann. Interessant war auch, daß für Gleichgewichtslagen nahe der Nachbeullast nicht die erste Eigenform, sondern die bei der transienten Berechnung des Beulens sich einstellende Verformungsfigur zu höherer Sensitivität führte.

Ein Mangel des beschriebenen Vorgehens besteht darin, daß die Sensitivität nicht allgemein sicher ermittelt werden kann, da letztendlich die kritische Anfangsgeschwindigkeitsverteilung nicht mit Sicherheit vorhersagbar ist. In weiten Bereichen war aber die erste Schwingungseigenform als kritische Störungsform zu beobachten. Beim Beispiel der Siloschale legten die Untersuchungen einer weiteren Gleichgewichtslage mit einem Lastniveau unterhalb der Nachbeullast, die sich gegenüber allen betrachteten Störungen als insensitiv erwies, den Schluß nahe, daß nur die Gleichgewichtslagen unterhalb dieses Lastniveaus wirklich sicher sind. D.h., die transiente Berechnung des Beulvorganges gibt das Systemverhalten und die Nachbeullast realistisch wieder.

Zur Absicherung dieser Ergebnisse sollten noch weitere, ähnliche Untersuchungen an Zy-

lindern bzw. anderen beulgefährdeten Strukturen erfolgen. Besonders wünschenswert wären verlässliche Aussagen über die kritische Anfangsgeschwindigkeitsverteilung, hier ist weitere Forschung vonnöten. Auch die Untersuchung der Interaktion zwischen “ungünstigen” Imperfektionsformen und der Sensitivität von Strukturen erscheint von hoher praktischer Bedeutung.

Bei der Lösertechnologie wird mit weiter wachsender Problemgröße kaum ein Weg an den Mehrgitterlösern vorbei führen. Allerdings scheint das Problem der schlechten Konditionierung noch nicht voll gelöst, auch wenn der Einsatz solcher Lösertypen für unterschiedliche Elementtypen, insbesondere bei hierarchischen adaptiven Netzen große Rechenzeiterparnisse verspricht. Die Restriktion und Prolongation von geschichtsabhängigen Variablen und elementinternen Freiheitsgraden zwischen den Netzen muß aber auch noch zufriedenstellend geklärt werden. Ebenso ist der Einsatz dieser Löser für nicht hierarchische Netze noch zu diskutieren. Hier könnten entsprechende Netzvergrößerungsstrategien zu Grobgitternetzen führen, die dann zur Mehrgitterlösung verwendet werden können.

Ein lohnendes Ziel für zukünftige Arbeiten liegt auch in der Kopplung adaptiver Verfahren mit parallelen Lösungsmethoden. Adaptive Netzverfeinerungsstrategien ermöglichen es, mit einer minimalen Anzahl von Unbekannten eine Lösung mit vorgegebener Genauigkeit zu erzielen. Durch die Verbindung mit parallelen Lösungsmethoden kann diese Aufgabe dann in deutlich verkürzter Zeit erledigt werden. Das größte Problem hierbei liegt in der dynamischen Verteilung der Rechenlast, erste Ergebnisse hierzu liegen im Schrifttum aber bereits vor. Größte Probleme werden dann unter Einsatz preiswerter Parallelrechner auf Basis von PC-Clustern effizient lösbar.

## Literatur

- [1] 4, DIN 18800 TEIL, *Stahlbauten, Stabilitätsfälle, Schalenbeulen*, Normenausschuß Bauwesen, Deutsches Institut für Normung e.V., Beuth Verlag GmbH, Berlin, November 1990.
- [2] AMDAHL, G., *The validity of the single processor approach to achieving large scale computing capabilities*, in Proceedings of the AFIPS Computing Conference, 30, 1967, pp. 483–485.
- [3] BARNARD, S.T. UND SIMON, H.D., *Fast multilevel implementation of recursive spectral bisection for partitioning unstructured problems*, Concurrency: Practice and Experience, 6 (1994), pp. 101–117.
- [4] BARRET, R., BERRY, M., CHAN, T.F., DEMMEL, J., DONATO, J., DONGARRA, J., EIJKHOUT, V., POZO, R., ROMINE, C. UND VAN DER VORST, H., *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1994.
- [5] BASTIAN, P., *Parallele adaptive Mehrgitterverfahren*, Teubner Verlag, Stuttgart, 1996.
- [6] BASTIAN, P., HACKBUSCH, W. UND WITTUM, G., *Additive and multiplicative multi-grid — a comparison*, Computing, 60 (1998), pp. 345–364.
- [7] BATHE, K.J., *Finite Elemente Methoden*, Springer-Verlag, Berlin, 1986.
- [8] BATHE, K. UND DVORKIN, E., *On the automatic solution of nonlinear finite element equations*, Computer & Structures, 17 (1983), pp. 871–879.
- [9] BECKER, E. UND BÜRGER, W., *Kontinuumsmechanik*, Teubner, Stuttgart, 1975.
- [10] BERGAN, P.G., HORRIGMOE, G., KRÅKELAND, B. UND SØREIDE, T.H., *Solution techniques for non-linear finite element problems*, Int. J. Num. Meth. Eng., 12 (1978), pp. 1677–1696.
- [11] BETTEN, J., *Kontinuumsmechanik*, Springer-Verlag, 1993.
- [12] BOERSMA, A., *Parallele Finite-Elemente-Methoden in der Festkörpermechanik und ihre Lösung mit algebraischen Mehrgitterverfahren*, Dissertation, Fachbereich Mechanik, Universität Darmstadt, 1995.
- [13] BRENDDEL, B., *Geometrisch nichtlineare Elastostabilität*, Dissertation, Institut für Baustatik, Universität Stuttgart, 1979.
- [14] BRONSTEIN, I.N. UND SEMENDAJEW, K.A., *Taschenbuch der Mathematik*, Verlag Nauka und Teubner Verlagsgesellschaft, 1989.
- [15] BRUASET, A.M., *A Survey of Preconditioned Iterative Methods*, Longman Scientific & Technical, 1995.

- 
- [16] BUDIANSKY, B., *Theory of buckling and post-buckling behavior of elastic structures*, Advances in Applied Mechanics, 14 (1974), pp. 2–65.
- [17] BURMEISTER, A., *Dynamische Stabilität nach der Methode der Finiten Elemente mit Anwendungen auf Kugelschalen*, Dissertation, Institut für Baustatik, Universität Stuttgart, 1987.
- [18] BURMEISTER, A. UND RAMM, E., *Dynamic stability analysis of shell structures*, in Computational Mechanics of Nonlinear Response of Shells, Krätzig, W.B. und Oñate, E., eds., Springer-Verlag, 1990, pp. 152–163.
- [19] CARSTENSEN, C. UND WAGNER, W., *Detecting symmetry-breaking bifurcation points of symmetric structures*, Int. J. Num. Meth. Eng., 36 (1993), pp. 3019–3039.
- [20] CARSTENSEN, C. UND WRIGGERS, P., *On perturbation behaviour in non-linear dynamics*, Communications in Numerical Methods in Engineering, 9 (1993), pp. 165–175.
- [21] CLOUGH, R.W. UND PENZIEN, J., *Dynamics of Structures*, Mc-Graw-Hill, 1975.
- [22] COTTLE, R.W., *Manifestations of the Schur complement*, Linear Algebra and its Applications, 8 (1974), pp. 189–211.
- [23] CRISFIELD, M.A., *A fast incremental/iterative solution procedure that handles “snap-through”*, Computer & Structures, 13 (1981), pp. 55–62.
- [24] —, *Finite Elements and Solution Procedures for Structural Analysis, Volume I: Linear Analysis*, Pineridge Press, 1986.
- [25] —, *Non-linear Finite Element Analysis of Solids and Structures, Volume I*, John Wiley & Sons, 1991.
- [26] DE BOER, R., *Vektor- und Tensorrechnung für Ingenieure*, Springer-Verlag, 1982.
- [27] DEMBO, R.S. UND STEIHAUG, T., *Truncate-Newton algorithms for large-scale unconstrained optimization*, Mathematical Programming, 26 (1983), pp. 190–212.
- [28] DEML, M., *Ein Finite-Element-Konzept zur Traglastanalyse imperfektionssensitiver Schalenstrukturen*, Dissertation, Lehrstuhl für Statik, Technische Universität München, 1997.
- [29] DEML, M. UND WUNDERLICH, W., *Direct evaluation of the ‘worst’ imperfection shape in shell buckling*, Computer Methods in Applied Mechanics and Engineering, 149 (1997), pp. 201–222.
- [30] DICKINSON, J.K. UND FORSYTH, P.A., *Preconditioned conjugate gradient methods for three-dimensional linear elasticity*, Int. J. Num. Meth. Eng., 37 (1994), pp. 2211–2234.

- [31] DINKLER, D., *Stabilität dünner Flächentragwerke bei zeitabhängigen Einwirkungen*. Habilitationsschrift, Technische Universität Braunschweig, 1988.
- [32] DOLL, S., *Zur Behandlung großer elastoviskoplastischer Deformationen bei isochor-volumetrisch entkoppeltem Stoffverhalten*, Dissertation, Institut für Mechanik, Universität Karlsruhe, 1998.
- [33] DUFF, I.S. UND REID, J.K., *The multifrontal solution of indefinite sparse symmetric linear equations.*, ACM Trans. Math. Softw., 9 (1983), pp. 302–325.
- [34] ECCS, *Buckling of Steel Shells – European Recommendations*. Fourth Edition, No. 56, European Convention for Structural Steelwork, 1988.
- [35] EISENSTAT, S.C., *Efficient implementation of a class of preconditioned conjugate gradient methods*, SIAM J. Sci. Stat. Comput., 2 (1981), pp. 1–4.
- [36] EISENSTAT, S.C., GURSKY, M.C., SCHULTZ, M.H. UND SHERMAN, A.H., *Yale sparse matrix package I: The symmetric codes*, Int. J. Num. Meth. Eng., 18 (1982), pp. 1145–1151.
- [37] ELLER, C. UND NAWROTZKI, P., *Dynamische Stabilitätsanalyse eines Dreigelenkstabwerks unter Verwendung numerischer Nachweiskonzepte*, ZAMM Zeitung für angewandte Mathematik und Mechanik, 78 (1998), pp. 483–493.
- [38] ERIKSSON, A. UND KOUHIA, R., *On step size adjustments in structural continuation problems*, Computer & Structures, 55 (1995), pp. 495–506.
- [39] ERIKSSON, A. UND PACOSTE, C., *Sensitivity aspects in the critical behaviour of elastic structures: A numerical approach*, in Computational Mechanics, Idelsohn, S., Oñate, E. und Dvorkin, E., eds., Barcelona, Spanien, 1998, CIMNE.
- [40] FARHAT, C., *A simple and efficient automatic FEM domain decomposer*, Computer & Structures, 28 (1988), pp. 579–602.
- [41] FIEDLER, M., *Algebraic connectivity of graphs*, Czech. Math., 98 (1973), pp. 298–305.
- [42] FISH, J., BELSKY, V. UND GOMAA, S., *Unstructured multigrid method for shells*, Int. J. Num. Meth. Eng., 39 (1996), pp. 1181–1197.
- [43] FREUND, R.W., *A transpose-free quasi-minimum residual algorithm for non-hermitian linear systems*, SIAM J. Sci. Comput., 14 (1993), pp. 470–482.
- [44] FREUND, R.W. UND NACHTIGAL, N.M., *QMR: a quasi-minimal residual method for non-hermitian linear systems*, Numerische Mathematik, 60 (1991), pp. 315–339.
- [45] ———, *A new Krylov-subspace method for symmetric indefinite linear systems*, in Proceedings of the 14th IMACS World Congress on Computational and Applied Mathematics, Ames, W. F., ed., IMACS, 1994, pp. 1253–1256.

- 
- [46] GEBHARDT, H., *Finite Element Konzepte für schubelastische Schalen*, Dissertation, Institut für Baustatik, Universität Karlsruhe, 1990.
- [47] GEIST, AL, *PVM : parallel virtual machine ; a users' guide and tutorial for networked parallel computing*, The MIT Press, 1994.
- [48] GEORGE, A., *Nested dissection of a regular finite element mesh*, SIAM J. Numer. Anal., 10 (1973), pp. 345–363.
- [49] —, *Direct solution of sparse and positive definite systems: some basic ideas and open problems*, in *Sparse Matrices and their Uses*, Duff, I. S., ed., Academic Press, 1981, pp. 283–306.
- [50] GEORGE, A. UND LIU, J. W. H., *Computer solution of large sparse positive definite systems*, Prentice Hall, Englewood Cliffs, N.Y., 1981.
- [51] —, *The evolution of the minimum degree ordering algorithm*, SIAM Review, 31 (1989), pp. 1–19.
- [52] GERVAIS, J., OUKIT, A. UND PIERRE, R., *Finite element analysis of the buckling and mode jumping of a rectangular plate*, Dynamics and Stability of Systems, 12 (1997).
- [53] GOLUB, G. H. UND VAN LOAN, C. H., *Matrix Computations*, John Hopkins University Press, Baltimore, 1993.
- [54] GRUTTMANN, F., *Strukturanalysen bei endlichen Deformationen*. Skriptum zur Vorlesung, 1996.
- [55] GUPTA, A., *Fast and effective algorithms for graph partitioning and sparse matrix ordering*, IBM J. Res. Develop., 41 (1997), pp. 171–183.
- [56] GUPTA, A., KARYPIS, G. UND KUMAR, V., *Highly scalable parallel algorithms for sparse matrix factorization*, Techn. Bericht., Department of Computer Science, University of Minnesota, 1995.
- [57] HACKBUSCH, W., *Iterative Solution of Large Sparse Systems of Equations*, Springer-Verlag, 1994.
- [58] HAGEDORN, P., *Non-Linear Oscillations*, Clarendon Press, Oxford, 1988.
- [59] HAHN, W., *Stability of Motion*, Springer-Verlag, 1967.
- [60] HAUPTMANN, R., *Strukturangepaßte geometrisch nichtlineare Finite Elemente für Flächentragwerke*, Dissertation, Institut für Mechanik, Universität Karlsruhe, 1997.
- [61] HAUPTMANN, R., SCHWEIZERHOF, K. UND DOLL, S., *A note on the application of the enhanced assumed strain method to the 'solid shell' concept to avoid volumetric locking in the range of incompressibility*, zur Veröffentlichung eingereicht, (1998).

- 
- [62] HENDRICKSON, B. UND LELAND, R., *An improved spectral graph partitioning algorithm for mapping parallel computations*, SIAM J. Sci. Comput., 16 (1995), pp. 452–469.
- [63] HESTENES, M. UND STIEFEL, E., *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Stand., 49 (1952), pp. 409–436.
- [64] HILBURGER, M.W., WAAS, A.M. UND STARNES, J.H., *Modeling the dynamic response and establishing post-buckling/post snap-thru equilibrium of discrete structures via a transient analysis*, Transactions of the ASME, 64 (1997), pp. 590–595.
- [65] HILLMANN, J., *Grenzlasten und Tragverhalten axial gestauchter Kreiszyinderschalen im Vor- und Nachbeulbereich*, Dissertation, Institut für Statik, Technische Universität Braunschweig, 1985.
- [66] HUGHES, T.J.R., *The Finite Element Method*, Prentice Hall, Englewood Cliffs, New Jersey, 1987.
- [67] HUGHES, T.J.R., LEVIT, I. UND WINGET, J., *An element-by-element solution algorithm for problems of structural and solid mechanics*, Computer Methods in Applied Mechanics and Engineering, 36 (1983), pp. 241–254.
- [68] HUGHES, T.J.R. UND PISTER, K.S., *Consistent linearization in mechanics of solids and structures*, Computer & Structures, 8 (1978), pp. 391–397.
- [69] JENNINGS, A. UND MALIK, G.M., *Partial elimination*, J. Inst. Math. Applic., 20 (1977), pp. 307–316.
- [70] JÜRCKE, R., *Zur Stabilität und Imperfektionsempfindlichkeit elastischer Schalen-tragwerke – Finite Element Formulierung der Anfangs-Nachbeul-Theorie*, Dissertation, Institut für konstruktiven Ingenieurbau, Ruhr-Universität Bochum, 1985.
- [71] KARYPIS, G. UND KUMAR, V., *A fast and high quality multilevel scheme for irregular graphs*, Techn. Bericht., TR 95-35, Department of Computer Science, University of Minnesota, 1995.
- [72] —, *METIS: unstructured graph partitioning and sparse matrix ordering*, Techn. Bericht., Department of Computer Science, University of Minnesota, 1995.
- [73] KELLER, H.B., *Numerical solution of bifurcation and nonlinear eigenvalue problems*, Application of Bifurcation Theory, 38 (1977), pp. 359–384.
- [74] KELLEY, C.T., *Iterative Methods for Linear and Nonlinear Equations*, Vol. 16, Frontiers in Applied Mathematics, SIAM, Philadelphia, 1995.
- [75] KERNIGHAN, B.W. UND LIN, S., *An efficient heuristic procedure for partitioning graphs*, The Bell System Technical Journal, (1970), pp. 291–307.

- [76] KHOROMSKIJ, B.N. UND WITTUM, G., *Robust iterative method for elliptic problems with highly varying coefficients in thin substructures*, Numerische Mathematik, 73 (1996), pp. 449–472.
- [77] KLAAS, O., *Adaptive Finite-Element-Methoden für MIMD-Parallelrechner zur Behandlung von Strukturproblemen mit Anwendung auf Stabilitätsprobleme*, Forschungs- und Seminarberichte aus dem Bereich Mechanik der Universität Hannover Bericht-Nr. F 96/7, Institut für Baumechanik und numerische Mechanik, 1996.
- [78] KLAAS, O., NIEKAMP, R. UND STEIN, E., *Parallel eigenvalue computations on hierarchical refined finite element meshes*, Computing in Civil and Building Engineering, (1995), pp. 553–559.
- [79] KLEIBER, M., *Parameter Sensitivity in Nonlinear Mechanics*, John Wiley & Sons, 1997.
- [80] KNEBEL, K., *Stabilität von Stahlzylindern mit unilateralen Randbedingungen bei statischen und dynamischen Beanspruchungen*, Dissertation, Institut für Mechanik, Universität Karlsruhe, 1997.
- [81] KNEBEL, K., PEIL, U., SCHULZ, H., SCHWEIZERHOF, K. UND UMMENHOFER, T., *Stabilität von stählernen Silozylinderschalen bei Belastung mit ruhendem und bewegtem Schüttgut*, SFB 219 Silo Abschlußbericht, Universität Karlsruhe, 1997.
- [82] KOITER, W.T., *Over de Stabilliteit van het elastisch Evenwicht*, Dissertation, Polytechniek Instituut Delft, H.J. Paris Amsterdam, 1945. Englische Übersetzung: On the stability of elastic equilibrium, NASA TT F-10, 833, 1967.
- [83] KOUHIA, R. UND MIKKOLA, M., *Tracing the equilibrium path beyond simple critical points*, Int. J. Num. Meth. Eng., 28 (1989), pp. 2923–2941.
- [84] ———, *Some aspects on efficient path-following*, Computer & Structures, 72 (1999), pp. 509–524.
- [85] KRÄTZIG, W., *Eine einheitliche statische und dynamische Stabilitätstheorie für Pfadverfolgungsalgorithmen in der numerischen Festkörpermechanik*, ZAMM Zeitung für angewandte Mathematik und Mechanik, 69 (1989), pp. 203–213.
- [86] KREUZER, E., *Numerische Untersuchung nichtlinearer dynamischer Systeme*, Springer-Verlag, 1987.
- [87] KRÖPLIN, B., DINKLER, D. UND HILLMANN, J., *An energy perturbation applied to nonlinear structural analysis*, Computer Methods in Applied Mechanics and Engineering, (1985), pp. 885–897.
- [88] KULISCH, U., *Memorandum über Computer, Arithmetik und Numerik*, Institut für Angewandte Mathematik, Universität Karlsruhe, 1997.

- [89] LÄMMER, L., *Parallelisierung von Anwendungen der Finite-Elemente-Methode im Bauingenieurwesen*, Bericht 1/97, Institut für Numerische Methoden und Informatik im Bauwesen, Technische Hochschule Darmstadt, 1997.
- [90] LANCZOS, C., *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Stand., 45 (1950), pp. 255–282.
- [91] ———, *Solution of systems of linear equations by minimized iterations*, J. Res. Nat. Bur. Stand., 49 (1952), pp. 33–53.
- [92] LEIPHOLZ, H., *Stabilität elastischer Systeme*, G. Braun Karlsruhe, 1980.
- [93] LENHARDT, I., *Krylov-Unterraum-Verfahren für Gleichungssysteme aus der Strukturmechanik auf sequentiellen und parallelen Rechnern*, Vol. 29, Tectum-Verlag, 1999. Edition Wissenschaft, Reihe Mathematik.
- [94] LENHARDT, I. UND ROTTNER, TH., *Krylov subspace methods for structural finite element analysis*, Parallel Computing, 25 (1999), pp. 861–875.
- [95] LJAPUNOV, M. A., *Problème Général de la Stabilité du Mouvement*, Vol. 17, Annals of Mathematics Studies, Princeton University Press, 1947. Original in russisch, 1892.
- [96] LUENBERGER, D. G., *Introduction to Linear and Nonlinear Programming*, Addison Wesley Publishing Company, 1973, Kap. (8) Conjugate direction methods, pp. 168–189.
- [97] MALKIN, J. G., *Theorie der Stabilität einer Bewegung*, Oldenbourg, 1959.
- [98] MATKOWSKY, B. J. UND REISS, E. L., *Singular perturbations of bifurcations*, SIAM J. Appl. Math., 33 (1977), pp. 230–255.
- [99] MINORSKY, N., *Introduction to non-linear mechanics*, Edwards Brothers Inc., Ann Arbor, Michigan, 1947.
- [100] NAWROTZKI, P., KRÄTZIG, W. UND MONTAG, U., *A unified computational stability concept for conservative and non-conservative shell responses*, Computer & Structures, 64 (1997), pp. 221–238.
- [101] NAYFEH, A., *Perturbation Methods*, John Wiley & Sons, 1973.
- [102] NEWMARK, N. M., *A method of computation for structural dynamics*, J. Eng. Mech. Div. ASCE, (1959), pp. 67–94.
- [103] NOGUCHI, H. UND HISADA, T., *Development of a sensitivity analysis method for nonlinear buckling load*, JSME Int. J. Ser. A, 38 (1995).
- [104] PAIGE, C. C. UND SAUNDERS, M. A., *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 12 (1975), pp. 617–629.

- [105] PAPADRAKAKIS, M., *Accelerating vector iteration methods*, J. Appl. Mech., 53 (1986), pp. 291–297.
- [106] —, *A truncated Newton-Lanczos method for overcoming limit and bifurcation points*, Int. J. Num. Meth. Eng., 29 (1990), pp. 1065–1077.
- [107] PAPADRAKAKIS, M. UND GANTES, C.J., *Preconditioned conjugate- and secant-Newton methods for non-linear problems*, Int. J. Num. Meth. Eng., 28 (1989), pp. 1299–1316.
- [108] PAPADRAKAKIS, M. UND SMEROU, S., *A new implementation of the Lanczos method in linear problems*, Int. J. Num. Meth. Eng., 29 (1990), pp. 141–159.
- [109] PARLETT, B.N., *A new look at the Lanczos algorithm for solving symmetric systems of linear equations*, Linear Algebra and its Applications, 29 (1980), pp. 323–346.
- [110] PETERSEN, C., *Statik und Stabilität der Baukonstruktionen*, Vieweg, 1982.
- [111] PLANIC, I. UND SAJE, M., *A quadratically convergent algorithm for the computation of stability points: the application of the determinant of the tangent stiffness matrix*, Internal Report 97/3, Faculty of Civil and Geodetic Engineering, University of Ljubljana, 1997.
- [112] POTHEN, A., SIMON, H.D. UND LIU, K., *Partitioning sparse matrices with eigenvectors of graphs*, SIAM J. Matrix Anal. Appl., 11 (1990), pp. 430–452.
- [113] RAMM, E., *Geometrisch nichtlineare Elastostatik und Finite Elemente*. Habilitationsschrift, Universität Stuttgart, 1976.
- [114] —, *Strategies for tracing the nonlinear response near limit points*, in Europe-US-Workshop on Nonlinear Finite Element Analysis in Structural Mechanics, Bathe, K.J., Stein, E. und Wunderlich, W., eds., Springer-Verlag, 1980, pp. 63–89.
- [115] RASHID, Y.R., *Three-dimensional analysis of elastic solids—I analysis procedure*, Int. J. Solids Structures, 5 (1969), pp. 1311–1331.
- [116] REESE, S. UND WRIGGERS, P., *A Finite Element Method for stability problems in finite elasticity*, Int. J. Num. Meth. Eng., 38 (1995), pp. 1171–1200.
- [117] RICCIUS, J., *Adaptive Methoden zur statischen und dynamischen Analyse von Flächentragwerken mit linearen Finiten Elementen*, Dissertation, Institut für Mechanik, Universität Karlsruhe, 1997.
- [118] RIKS, E., *An incremental approach to the solution of snapping and buckling problems*, Int. J. Solids Structures, 15 (1979), pp. 529–551.
- [119] RIKS, E., BROGAN, F.A. UND RANKIN, C.C., *Numerical aspects of shell stability analysis*, in Computational Mechanics of Nonlinear Response of Shells, Krätzig, W.B. und Oñate, E., eds., Springer Verlag, 1990, pp. 125–151.

- [120] RIKS, E., RANKIN, C.C. UND BROGAN, F.A., *On the solution mode jumping phenomena in thin-walled shell structures*, Computer Methods in Applied Mechanics and Engineering, 136 (1996), pp. 59–92.
- [121] ROTTNER, TH., LENHARDT, I., ALEFELD, G. UND SCHWEIZERHOF, K., *Nonlinear structural finite element analysis using the preconditioned Lanczos method on serial and parallel computers*, BIT, 37 (1997), pp. 759–769.
- [122] ROTTNER, TH. UND SCHWEIZERHOF, K., *Transient FE-analyses of silo buckling behaviour using parallel computers*, in Developments in Computational Mechanics with High Performance Computing, Topping, B.H.V., ed., Civil-Comp Press, 1999, pp. 79–87.
- [123] ROTTNER, TH., SCHWEIZERHOF, K., LENHARDT, I. UND ALEFELD, G., *Fully parallel nonlinear finite element computations using the pre-conditioned Lanczos method for solving linear equations*, in Advances in Computational Mechanics with High Performance Computing, Topping, B.H.V., ed., Civil-Comp Press, 1998, pp. 95–101.
- [124] SALERNO, G. UND CASCIARO, R., *Mode jumping and attractive paths in multi-mode elastic buckling*, Int. J. Num. Meth. Eng., 40 (1997), pp. 833–861.
- [125] SAMUELSON, L. UND EGGWERTZ, S., *Shell Stability Handbook*, Elsevier Applied Science, 1992.
- [126] SCHMIDT, J.G. UND STARKE, G., *Coarse space orthogonalization for indefinite linear systems of equations arising in geometrically nonlinear elasticity*, Preprint Nr. 97/2, Institut für Wissenschaftliches Rechnen und Mathematische Modellbildung, Universität Karlsruhe, 1997.
- [127] SCHWARZ, H.R., *Methode der finiten Elemente*, Teubner Studienbücher Mathematik, Stuttgart, 3. Aufl., 1991.
- [128] SCHWEIZERHOF, K., *Nichtlineare Berechnung von Tragwerken unter verformungsabhängiger Belastung mit finiten Elementen*, Dissertation, Institut für Baustatik, Universität Stuttgart, 1982.
- [129] ———, *Quasi-Newton Verfahren und Kurvenverfolgungsalgorithmen für die Lösung nichtlinearer Gleichungssysteme in der Strukturmechanik*, Schriftenreihe des Instituts für Baustatik Heft 9, Universität Karlsruhe, 1989.
- [130] SCHWEIZERHOF, K., HAUPTMANN, R., KNEBEL, K., RAABE, M. UND ROTTNER, TH., *Statische und dynamische FE-Stabilitätsuntersuchungen an Siloschalen mit ungleichförmiger Schüttgutfüllung*, in Finite Elemente in der Baupraxis, Wriggers, P., Meißner, U., Stein, E. und Wunderlich, W., eds., Ernst & Sohn, 1998, pp. 287–306.
- [131] SCHWEIZERHOF, K. UND MITARBEITER, *FEAP-MeKa*, 1988-2000. Basierend auf: Taylor, R.L.: A Finite Element Analysis Program, University of Berkeley, 1991.

- [132] SCHWEIZERHOF, K. UND WRIGGERS, P., *Consistent linearization for path following methods in nonlinear FE analysis*, Computer Methods in Applied Mechanics and Engineering, 59 (1986), pp. 261–279.
- [133] SIMO, J.C. UND RIFAI, M.S., *A class of mixed assumed strain methods and the method of incompatible modes*, Int. J. Num. Meth. Eng., 29 (1990), pp. 1595–1638.
- [134] SNIR, M., OTTO, S.W., HUSS-LEDERMAN, S., WALKER, D.W. UND DONGARRA, J., *MPI – The Complete Reference*, The MIT Press, 1996.
- [135] SOPHIANOPOULOS, D.S., *Point attractors and dynamic buckling of autonomous systems under step loading*, Int. J. Solids Structures, 36 (1999), pp. 5357–5386.
- [136] SPEICHER, G., *Beulstabilität langer stählener Kreiszyinderschalen unter Biegebeanspruchung*, Dissertation, Institut für Stahlbau, Universität Karlsruhe, 1998.
- [137] SPOHR, I., *Störenergie-Konzept für den elasto-plastischen Beulsicherheitsnachweis beliebig belasteter Schalen*, Dissertation, Institut für Statik, Technische Universität Braunschweig, 1998.
- [138] STIEFENHOFER, A. UND RICHTER, H., *Evidence for 2-dimensional Finite-Element-Analysis of endodontic post and cores*, European Prosthodontic Association/Deutsche Gesellschaft für Zahnärztliche Prothetik und Werkstoffkunde (EPA/DGZPW), 163 (1996), p. 105.
- [139] THOMPSON, J.M.T. UND HUNT, G.W., *A general theory of elastic stability*, John Wiley & Sons, 1973.
- [140] TRANEL, G., *Stabilitätsnachweis beliebiger Schalen mit dem Konzept der Störenergie*, Dissertation, Institut für Statik, Technische Universität Braunschweig, 1994.
- [141] TRUESDELL, C. UND NOLL, W., *The non-linear field theories of mechanics*, in Handbuch der Physik III/3, Flügge, S., ed., Springer-Verlag, 1965.
- [142] UMMENHOFER, T., *Stabilitätsverhalten imperfekter zylindrischer Stahlsiloschalen – experimentelle und numerische Untersuchungen*, Dissertation, Versuchsanstalt für Stahl, Holz und Steine, Universität Karlsruhe, 1996.
- [143] VIELSACK, P., *Zur Anwendung der Störungsrechnung auf lineare Eigenwertprobleme der Stabstatik*, Ingenieur-Archiv, 46 (1977), pp. 271–280.
- [144] ———, *Stabilitätskriterien in der Strukturmechanik*. Skriptum zur Vorlesung, 1999.
- [145] VIELSACK, P., SPRENGER, W. UND HESSE, T., *Sensitivität stabiler Gleichgewichtszustände elastischer Strukturen*, Stahlbau, 64 (1995), pp. 321–325.
- [146] WAGENHUBER, W., *Imperfektionssensitivität und rechnerischer Nachweis der Beulsicherheit dünner Schalen*, Dissertation, Institut für Statik, Technische Universität Braunschweig, 1989.

- 
- [147] WAGNER, C., KINZELBACH, W. UND WITTUM, G., *Schur-complement multigrid*, Numerische Mathematik, 75 (1997), pp. 523–545.
- [148] WAGNER, W., *A note on FEM buckling analysis*, Communications in Numerical Methods in Engineering, 11 (1995), pp. 149–158.
- [149] WAGNER, W. UND WRIGGERS, P., *A simple method for the calculation of post-critical branches*, Engineering Computations, 5 (1988), pp. 103–109.
- [150] WASZCZYSZYN, Z., CICHON, C. UND RADWAŃSKA, M., *Stability of Structures by Finite Element Methods*, Vol. 40, Studies in Applied Mechanics, Elsevier, 1994.
- [151] WATHEN, A.J., *An analysis of some element-by-element techniques*, Computer Methods in Applied Mechanics and Engineering, 74 (1989), pp. 271–287.
- [152] WRIGGERS, P., *Konsistente Linearisierungen in der Kontinuumsmechanik und ihre Anwendung auf die Finite-Element-Methode*, Forschungs- und Seminarberichte aus dem Bereich Mechanik der Universität Hannover Bericht-Nr. 88/4, Institut für Baumechanik und numerische Mechanik, 1988.
- [153] WRIGGERS, P. UND C., CARSTENSEN, *An efficient algorithm for the computation of stability points of dynamical systems under step load*, Engineering Computations, 9 (1992), pp. 669–679.
- [154] WRIGGERS, P. UND SIMO, J.C., *A general procedure for the direct computation of turning and bifurcation points*, Int. J. Num. Meth. Eng., 30 (1990), pp. 155–176.
- [155] WRIGGERS, P., WAGNER, W. UND MIEHE, CH., *A quadratically convergent procedure for the calculation of stability points in finite element analysis*, Computer Methods in Applied Mechanics and Engineering, 70 (1988), pp. 329–347.
- [156] ZIENKIEWICZ, O.C. UND TAYLOR, R.L., *The Finite Element Method*, McGraw-Hill, 1989.

## A Mathematische Grundlagen

In diesem Abschnitt sollen kurz einige mathematischen Beziehungen angegeben werden, die in der vorliegenden Arbeit verwendet wurden. Für eine ausführliche Darstellung wird auf die Literatur verwiesen, z.B. [26].

### A.1 Rechenregeln zur Divergenz

$$\operatorname{div}(\mathbf{T}\mathbf{v}) = \mathbf{v} \operatorname{div}(\mathbf{T}^T) + \mathbf{T}^T \operatorname{grad} \mathbf{v} \quad (\text{A.1})$$

$$\operatorname{div}(\mathbf{v} \times \mathbf{T}) = \mathbf{v} \times \operatorname{div} \mathbf{T} + \operatorname{grad} \mathbf{v} \times \mathbf{T} \quad (\text{A.2})$$

### A.2 Gaußscher Integralsatz

Mit dem Gaußschen Integralsatz oder auch Divergenztheorem können Volumen- in Flächenintegrale transformiert werden:

$$\int_v \operatorname{div} \mathbf{u} \, dv = \int_a \mathbf{u} \cdot \mathbf{n} \, da \quad (\text{A.3})$$

beziehungsweise

$$\int_v \operatorname{div} \mathbf{T} \, dv = \int_a \mathbf{T} \cdot \mathbf{n} \, da, \quad (\text{A.4})$$

mit dem nach außen gerichteten Flächennormalenvektor  $\mathbf{n}$ .

## B Finite Element Formulierung am Beispiel eines ebenen Scheibenelementes

Die in Abschnitt 2.2 eingeführte Finite Element Formulierung ist in Matrizendarstellung von dem gewählten Modell abhängig. So unterscheiden sich Scheiben-, Volumen-, und Schalenelemente in der Anzahl der Koordinaten und Freiheitsgrade pro Knoten, was sich wiederum auf die Darstellung der Matrizen in den speziellen Fällen auswirkt. Für den einfachsten Fall der ebenen, geometrisch nichtlinearen Scheibe soll deshalb hier exemplarisch die Formulierung angegeben werden.

Zunächst sind in der Ebene zwei Koordinaten zur Angabe der Lage der Knoten in der Ausgangskonfiguration  $\mathbf{X}_I$  notwendig:

$$\mathbf{X}_I = \begin{bmatrix} X_{1I} \\ X_{2I} \end{bmatrix}, \quad (\text{B.1})$$

die Interpolation mittels der Ansatzfunktionen  $N_I(\xi, \eta)$  erfolgt über

$$\mathbf{X}^h = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \sum_{I=1}^{I_e} N_I(\xi, \eta) \mathbf{X}_I. \quad (\text{B.2})$$

Entsprechend dem isoparametrischen Konzept kann die Lage der Knoten in der Momentankonfiguration durch

$$\mathbf{x}^h = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \sum_{I=1}^{I_e} N_I(\xi, \eta) (\mathbf{X}_I + \mathbf{d}_I) \quad (\text{B.3})$$

mit dem Knotenverschiebungsvektor  $\mathbf{d}$  mit

$$\mathbf{u}^h = \sum_{I=1}^{I_e} N_I(\xi, \eta) \mathbf{d}_I \quad (\text{B.4})$$

beschrieben werden. Durch die matrizielle Darstellung der Spannungs- und Verzerrungstensoren

$$\mathbf{S} = \begin{bmatrix} S_{11} \\ S_{22} \\ S_{12} \end{bmatrix} \quad \text{und} \quad \mathbf{E} = \begin{bmatrix} E_{11} \\ E_{22} \\ 2E_{12} \end{bmatrix} \quad (\text{B.5})$$

erhält man unter Verwendung von (2.8) und der Einarbeitung der Diskretisierung die Darstellung

$$\mathbf{E}^h = \begin{bmatrix} \frac{1}{2}(\mathbf{x}_1^T \mathbf{x}_1 - 1) \\ \frac{1}{2}(\mathbf{x}_2^T \mathbf{x}_2 - 1) \\ \mathbf{x}_{,1}^T \mathbf{x}_{,2} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(2u_{1,1} + u_{1,1}^2 + u_{2,1}^2) \\ \frac{1}{2}(2u_{2,2} + u_{2,2}^2 + u_{1,2}^2) \\ u_{1,2} + u_{1,1}u_{1,2} + u_{2,1} + u_{2,1}u_{2,2} \end{bmatrix} \quad (\text{B.6})$$

und weiter unter Berücksichtigung von

$$\delta \mathbf{x}^h = \sum_{I=1}^{I_e} N_I \delta \mathbf{d}_I \quad \text{und} \quad \delta \mathbf{x}_{,\alpha}^h = \sum_{I=1}^{I_e} N_{I,\alpha} \delta \mathbf{d}_I \quad \text{mit} \quad \alpha = 1, 2 \quad (\text{B.7})$$

kann  $\delta \mathbf{E}^h$  als (siehe Gleichung (2.45))

$$\delta \mathbf{E}^h = \begin{bmatrix} \delta E_{11} \\ \delta E_{22} \\ 2\delta E_{12} \end{bmatrix} = \sum_{I=1}^{I_e} \begin{bmatrix} N_{I,1} \mathbf{x}_{,1}^T \\ N_{I,2} \mathbf{x}_{,2}^T \\ N_{I,1} \mathbf{x}_{,2}^T + N_{I,2} \mathbf{x}_{,1}^T \end{bmatrix} \delta \mathbf{d}_I = \sum_{I=1}^{I_e} \mathbf{B}_I \delta \mathbf{d}_I \quad (\text{B.8})$$

angegeben werden. Die Ableitungen  $N_{I,\alpha}$  können dabei aus den Ableitungen der Ansatzfunktionen in den lokalen Einheitskoordinaten  $\boldsymbol{\xi} = [\xi, \eta]^T$  mit der Jacobimatrix  $\mathbf{J}$  gewonnen werden:

$$N_{I,\alpha} = \begin{bmatrix} N_{I,1} \\ N_{I,2} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} N_{I,\xi} \\ N_{I,\eta} \end{bmatrix}. \quad (\text{B.9})$$

Der ebenfalls in Gleichung (2.45) benötigte Term

$$D\delta\mathbf{E} = \begin{bmatrix} \delta\Delta E_{11} \\ \delta\Delta E_{22} \\ 2\delta\Delta E_{12} \end{bmatrix} = \sum_{I=1}^{I_e} \sum_{K=1}^{I_e} \delta\mathbf{d}_I^T \begin{bmatrix} N_{I,1}N_{K,1} \\ N_{I,2}N_{K,2} \\ \frac{1}{2}(N_{I,1}N_{K,2} + N_{I,2}N_{K,1}) \end{bmatrix} \Delta\mathbf{d}_K \quad (\text{B.10})$$

ergibt durch anschließende Bildung des inneren Tensorproduktes mit dem Spannungstensor  $\mathbf{S}$

$$D\delta\mathbf{E} \cdot \mathbf{S} = \text{trace}(\mathbf{S}\delta\mathbf{E}^T) = \sum_{I=1}^{I_e} \sum_{K=1}^{I_e} \delta\mathbf{d}_I^T \hat{S}_{IK} \mathbf{I} \Delta\mathbf{d}_K = \sum_{I=1}^{I_e} \sum_{K=1}^{I_e} \delta\mathbf{d}_I^T \mathbf{G}_{IK} \Delta\mathbf{d}_K \quad (\text{B.11})$$

mit

$$\hat{S}_{IK} = S_{11}N_{I,1}N_{K,1} + S_{22}N_{I,2}N_{K,2} + S_{12}(N_{I,1}N_{K,2} + N_{I,2}N_{K,1}). \quad (\text{B.12})$$

Unter Verwendung eines St. Venant Kirchhoff Materialgesetzes

$$\mathbf{S} = \mathbf{C}\mathbf{E} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \mathbf{E} \quad (\text{B.13})$$

lautet dann die linearisierte Form des Prinzips der virtuellen Verschiebungen für ein Scheibenelement  $e$  der konstanten Dicke  $t$

$$DG^e \cdot \Delta\mathbf{u} = \int_{A^e} (\delta\mathbf{E}^T \mathbf{C} \Delta\mathbf{E} + \Delta\delta\mathbf{E}^T \mathbf{S}) t dA \quad (\text{B.14})$$

$$= \sum_{I=1}^{I_e} \sum_{K=1}^{I_e} \delta\mathbf{d}_I^T \underbrace{\int_{(A^e)} (\mathbf{B}_I^T \mathbf{C} \mathbf{B}_K + \mathbf{G}_{IK}) t dA}_{\mathbf{K}_{T IK}^e} \Delta\mathbf{d}_K \quad (\text{B.15})$$

## C Speicherstruktur

Wenn die Koeffizientenmatrix nur dünn besetzt ist, empfiehlt es sich, nur die von Null verschiedenen Einträge abzuspeichern. Jedoch muß dann eine Zuordnung der gespeicherten

Einträge zur entsprechenden Position in der Matrix durch weitere Indexfelder erfolgen. Für symmetrische Matrizen muß darüber hinaus nur die obere bzw. untere Dreieckmatrix gespeichert werden.

Für die  $n \times n$  Matrix

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 13 & 0 & 15 \\ 0 & 2 & 0 & 24 & 25 \\ 13 & 0 & 3 & 0 & 0 \\ 0 & 24 & 0 & 4 & 0 \\ 15 & 25 & 0 & 0 & 5 \end{pmatrix} \quad (\text{C.1})$$

soll die sogenannte *compressed column storage* Speicherstruktur (siehe z.B. [4] oder [36]) erläutert werden. In einem ersten `real`-Feld `a` werden alle Einträge im oberen Dreieck der

	1	2	3	4	5	6	7	8	9
a	1	13	15	2	24	25	3	4	5
ja	1	3	5	2	4	5	3	4	5
ia	1	4	7	8	9	10			

Tabelle C.1: Kompakte Speicherstruktur

Matrix  $\mathbf{A}$  zeilenweise gespeichert. Dieses Feld hat die Länge  $\text{nonzero}(\mathbf{A})$ , also die Anzahl aller Nichtnullelemente von  $\mathbf{A}$ . In einem weiteren `integer`-Feld `ja` wird die Zuordnung zur jeweiligen Spalte realisiert. Beispielsweise gehört der dritte Matrixeintrag (Wert 15) in die fünfte Spalte, der siebte Eintrag (Wert 3) in die dritte Spalte. Damit ist aber noch nicht die Zugehörigkeit eines Matrixeintrages zu einer Zeile festgelegt. Dieses geschieht durch ein weiteres `integer`-Feld `ia` der Länge  $n + 1$ . Die Einträge dieses Feldes enthalten jeweils die Nummer des ersten Eintrages einer Zeile. So beginnt die erste Zeile mit dem ersten Eintrag, die zweite Zeile mit dem vierten Matrixeintrag, die dritte Zeile mit dem siebten Eintrag usw. Der letzte Eintrag `ia(n+1)` ist für symmetrische Matrizen stets `ia(n)+1`; durch diesen zusätzlichen Eintrag sind Schleifen über Matrixzeilen einfach realisierbar, da die Elemente der  $j$ -ten Zeile von `ia(j)` bis `ia(j+1)-1` gespeichert sind. Dieser Zugriff wird bei einer symmetrischen Matrix durch den zusätzlichen Eintrag im `ia`-Feld auch für die letzte Zeile ermöglicht.

# Lebenslauf

<b>Name</b>	Thomas Rottner
<b>Geburtsdatum</b>	2. Juli 1967
<b>Geburtsort</b>	Karlsruhe
<b>Schulbesuch</b>	1973 – 1977 Grundschule Wöschbach (Kreis Karlsruhe)  1977 – 1978 Realschule Linkenheim (Kreis Karlsruhe)  1978 – 1986 Gymnasium Neureut (Kreis Karlsruhe) Abschluß mit allgemeiner Hochschulreife
<b>Wehrdienst</b>	1986 – 1987
<b>Studium</b>	1987 – 1994 Maschinenbau an der Universität Karlsruhe (TH) Abschluß als Diplom-Ingenieur
<b>Beruf</b>	seit 1994 wissenschaftlicher Mitarbeiter am Institut für Mechanik der Universität Karlsruhe (TH)