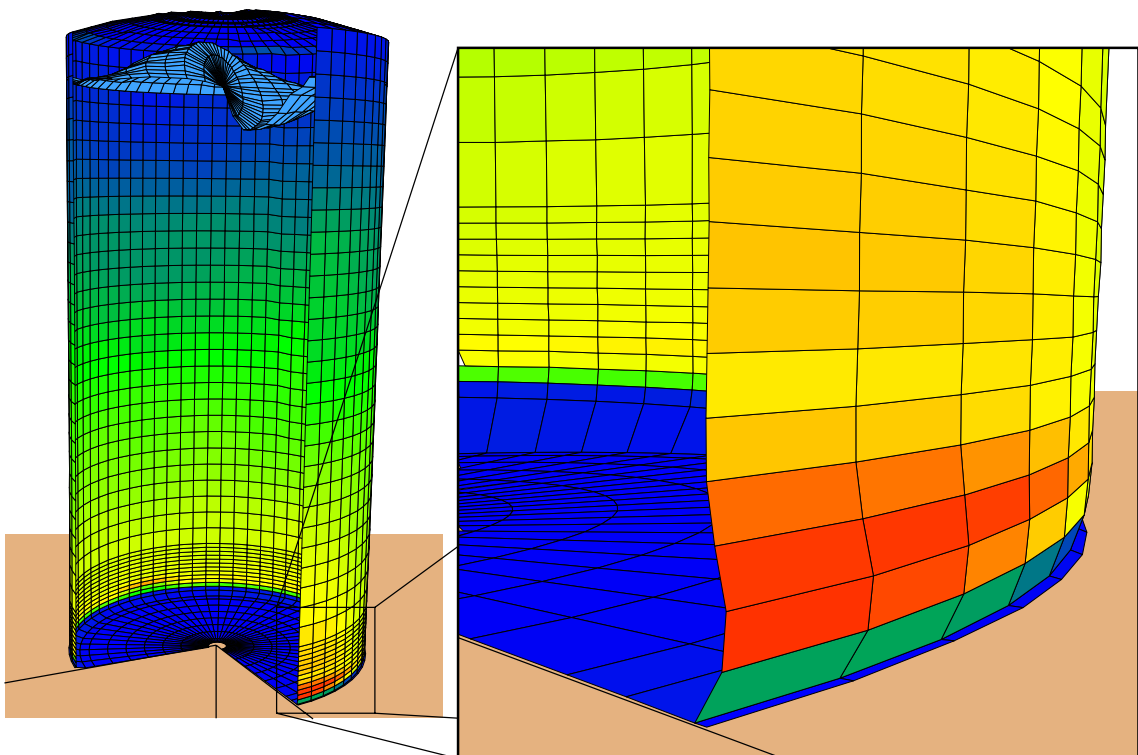


**Parallele Finite-Element-Simulation  
der Bauwerk-Boden-Interaktion mit  
adaptiven Zeitintegrationsverfahren**

Markus Rapolder



# **Parallele Finite-Element-Simulation der Bauwerk-Boden-Interaktion mit adaptiven Zeitintegrationsverfahren**

**Copyright by Markus Rapolder**

Destouchesstraße 3  
80803 München

Alle Rechte, insbesondere das der Übersetzung in fremde Sprachen, vorbehalten.  
Es ist gestattet, dieses elektronische Dokument für eigene Zwecke zu vervielfältigen.

Das vorliegende elektronische Dokument ist in gedruckter Form in der Schriftenreihe  
"Berichte aus dem konstruktiven Ingenieurbau" der Technischen Universität München  
erhältlich (ISSN 0941 - 925X)

## **Parallele Finite-Element-Simulation der Bauwerk-Boden-Interaktion mit adaptiven Zeitintegrationsverfahren**

### **Zusammenfassung**

In der vorliegenden Arbeit werden Methoden zur effizienten und robusten nichtlinearen dynamischen Berechnung von gekoppelten Systemen mit sehr vielen Freiheitsgraden vorgestellt. Die numerische Simulation erfolgt mit einem modifizierten Zeitintegrationsverfahren sowie einer adaptiven Schrittweitensteuerung. Die Algorithmen werden insbesondere für die numerische Behandlung von Interaktionsvorgängen und für die Kontaktsimulation optimiert. Mit Hilfe der Parallelisierung auf Elementebene kann die verwendete semi-analytische Finite-Element-Berechnung beschleunigt werden. Die Leistungsfähigkeit der entwickelten Verfahren wird mit der Simulation von unverankerten, flüssigkeitsgefüllten Behältern unter Erdbebeneinwirkung demonstriert.

## **Parallel Finite-Element-Analysis of the Structure-Soil-Interaction with Adaptive Time-Integration Procedures**

### **Summary**

This thesis deals with the derivation of efficient and robust algorithms for the dynamic analysis of coupled structures with a large number of degrees of freedom. The numerical simulations are based on a modified time integration scheme combined with an adaptive time step control. The algorithms are optimized for the numerical treatment of interaction effects and contact simulations. Parallel computing of the semi-analytical Finite Element model accelerates the numerical process. Calculations of unanchored liquid filled storage tanks under earthquake excitation exhibit the efficiency of the derived algorithms.

### Vorwort

Die vorliegende Arbeit entstand in den Jahren 1995 bis 2000 während meiner Tätigkeit als wissenschaftlicher Mitarbeiter und Assistent am Lehrstuhl für Statik der Technischen Universität München.

Mein Dank gilt Herrn Prof. Dr.-Ing. W. Wunderlich für die Übernahme des ersten Berichts und die Gewährung des zur Erstellung der Arbeit notwendigen Freiraums. Bei Herrn Prof. Dr. rer. nat. E. Rank möchte ich mich für die wertvollen Hinweise und die Übernahme des Koreferats bedanken, ebenso bei Herrn Prof. Dr.-Ing. H. Grundmann für die Leitung der Prüfungskommission.

Für die angenehme Zusammenarbeit und die großzügige Unterstützung bedanke ich mich sehr herzlich bei Herrn Prof. Dr.-Ing. K.-U. Bletzinger, Herrn Prof. Dr.-Ing. G. Knittel, Herrn Prof. Dr.-Ing. G. Kiener sowie bei Herrn Prof. Dr.-Ing. K. Schikora.

Das motivierende Arbeitsklima am Lehrstuhl und das freundschaftliche Verhältnis unter den Assistenten sowie die hilfreichen Anregungen von Herrn Prof. Dr.-Ing. H. Cramer haben sehr zum Gelingen dieser Arbeit beigetragen.

Der größte Dank gebührt schließlich meiner Frau Petra, die mich immer wieder ermutigt hat und viele Stunden auf mich verzichten mußte. Nicht zuletzt möchte ich mich bei meinen Eltern bedanken, die mir meine Ausbildung ermöglicht haben.

München, im August 2000

Markus Rapolder

Vollständiger Abdruck der von der Fakultät für Bauingenieur- und Vermessungswesen der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr.-Ing. H. Grundmann

Prüfer der Dissertation:

1. Univ.-Prof. Dr.-Ing. W. Wunderlich, emeritiert
2. Univ.-Prof. Dr. rer. nat. E. Rank

Die Dissertation wurde am 13. April 2000 bei der Technischen Universität München eingereicht und durch die Fakultät für Bauingenieur- und Vermessungswesen am 21. Juli 2000 angenommen.

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	2
1.2	Ziele und Inhalte der Arbeit . . . . .	3
<b>2</b>	<b>Nichtlineare Finite-Element-Berechnung</b>	<b>5</b>
2.1	Grundlagen der Kontinuumsmechanik . . . . .	5
2.1.1	Kinematik . . . . .	5
2.1.2	Gleichgewicht . . . . .	7
2.1.3	Materialgesetz . . . . .	8
2.2	Finite-Element-Methode . . . . .	8
2.2.1	Prinzip der virtuellen Verschiebungen . . . . .	9
2.2.2	Diskretisierung . . . . .	9
2.2.3	Sonderfall: Rotationssymmetrische Finite Elemente . . . . .	10
2.3	Nichtlineare Gleichungslösung . . . . .	11
2.3.1	Newton-Raphson-Verfahren . . . . .	11
2.3.2	Quasi-Newton-Verfahren . . . . .	12
2.3.3	Konvergenzkriterien . . . . .	13
<b>3</b>	<b>Berechnung im Zeitbereich</b>	<b>15</b>
3.1	Grundgleichungen . . . . .	15
3.2	Zeitintegrationsverfahren – Überblick . . . . .	16
3.2.1	Einschrittverfahren – Mehrschrittverfahren . . . . .	17

3.2.2	Indirekte Verfahren . . . . .	17
3.2.3	Explizite Verfahren . . . . .	18
3.2.4	Implizite Verfahren . . . . .	18
3.2.5	Energieerhaltende Verfahren . . . . .	19
3.2.6	Raum–Zeit–Diskretisierung . . . . .	19
3.3	Modifizierter impliziter Zeitintegrationsalgorithmus . . . . .	20
3.4	Eigenschaften in der linearen Dynamik . . . . .	23
3.4.1	Spektrale Stabilität . . . . .	25
3.4.2	Genauigkeit . . . . .	27
3.4.3	Numerische Dissipation . . . . .	27
3.4.4	Overshooting . . . . .	36
3.5	Eigenschaften in der nichtlinearen Strukturodynamik . . . . .	36
3.5.1	Stoß- und Kontaktsimulation . . . . .	37
3.5.2	Energetische Stabilität . . . . .	40
3.5.3	Genauigkeit . . . . .	46
<b>4</b>	<b>Adaptive Zeitschrittsteuerung</b>	<b>47</b>
4.1	Integrationsfehler . . . . .	48
4.2	Relativer Fehler - Bezugsgröße . . . . .	50
4.3	Schrittweitenanpassung . . . . .	52
4.3.1	Beispiel: Gedämpft schwingender Stab . . . . .	53
4.4	Zusatzmaßnahmen für Kontaktsimulation . . . . .	56
4.5	Begleitende Maßnahmen . . . . .	60
4.6	Numerische Simulation: Schalenbeulen . . . . .	61
<b>5</b>	<b>Parallele Finite-Element-Berechnung</b>	<b>65</b>
5.1	Allgemeines . . . . .	65
5.2	Grundlagen der Parallelen Berechnung . . . . .	66
5.2.1	Rechnersysteme – Architekturen . . . . .	66

5.2.2	Programmiermodelle und Hilfsmittel . . . . .	68
5.2.3	Effektivität der Parallelisierung . . . . .	72
5.3	Parallelisierung der dynamischen Berechnung . . . . .	76
5.3.1	Gebietszerlegung . . . . .	77
5.3.2	Parallelisierung auf Elementebene . . . . .	78
5.3.3	Dynamischer Lastausgleich . . . . .	83
5.3.4	Parallele Simulation – Ergebnisse . . . . .	88
<b>6</b>	<b>Anwendungsbeispiel: Flüssigkeitsgefüllte Behälter</b>	<b>93</b>
6.1	Systembeschreibung Behälter . . . . .	93
6.1.1	Schadensfälle . . . . .	95
6.1.2	Bemessung - Überblick . . . . .	96
6.2	Finite-Element-Modell . . . . .	97
6.2.1	Kopplung unterschiedlicher Elemente . . . . .	99
6.3	Dynamische Einwirkung: Erdbeben . . . . .	101
6.4	Parameterstudie . . . . .	105
6.4.1	Einfluß der Behältergeometrie . . . . .	106
6.4.2	Verhalten bei unterschiedlicher Bodensteifigkeit . . . . .	114
6.4.3	Einfluß der Einwirkung . . . . .	115
6.4.4	Strukturversagen . . . . .	118
6.4.5	Schlußfolgerungen aus den Berechnungen . . . . .	120
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>123</b>
	<b>Notationsübersicht</b>	<b>125</b>
	<b>Literaturverzeichnis</b>	<b>129</b>



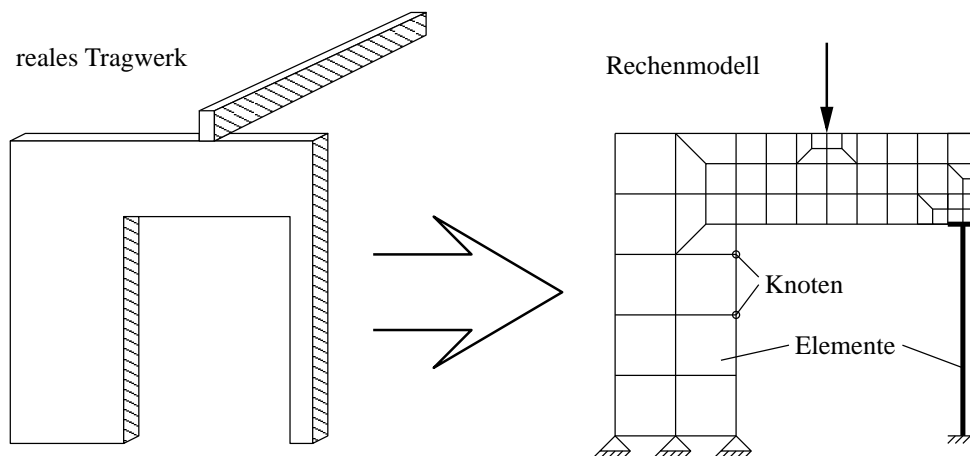


# Kapitel 1

## Einleitung

Bereits *Galileo Galilei*\* befaßte sich mit der statischen Berechnung von Bauteilen. Besonders bekannt sind seine Überlegungen zum Kragarm in [Gal38]. Eine praktische Bedeutung hatten allerdings erst die Arbeiten von *Claude Navier*†, die den Grundstein zur statischen Berechnung im heutigen Sinne legten.

Mit der Erfindung des Digitalrechners durch *Konrad Zuse*‡ war die Entwicklung neuer und verbesserter Verfahren zur Berechnung von Tragstrukturen im gesamten Ingenieurwesen möglich. Vor allem die *Finite-Element-Methode (FEM)* (Abbildung 1.1) sowie Methoden zur Lö-



**Abb. 1.1:** Modellbildung - Grundgedanke der FEM

sung größerer Gleichungssysteme bilden seit etwa 40 Jahren die Voraussetzungen zur detaillierten numerischen Analyse komplexer Systeme [Bat86, RS97, WR95].

Die Grundlagen und Rechenmethoden zur dynamischen Strukturanalyse wurden ebenfalls weiterentwickelt. Mit der Methode der Finiten Elemente und sogenannten Zeitschrittverfahren kann heute die nichtlineare dynamische Untersuchung komplizierter und gekoppelter Bauteile durchgeführt werden.

\*italienischer Mathematiker und Philosoph, 1564-1642

†französischer Ingenieur, 1785-1836

‡deutscher Bauingenieur, 1910-1995

## 1.1 Motivation

Die Berechnung von komplexen Strukturen mit Hilfe der Finite-Element-Methode erfordert einen beachtlichen numerischen Aufwand. Seit den Anfängen der computerorientierten Mechanik wird deshalb versucht, die Anzahl der Unbekannten und damit den Rechenaufwand durch geeignete Maßnahmen zu verringern:

- Die Reduktion der drei räumlichen Dimensionen des allgemeinen Kontinuums auf zwei Dimensionen bei Scheiben, Platten und Schalen oder die Approximation eines stabförmigen Körpers mit Hilfe von eindimensionalen Stabmodellen stellen allgemein bekannte und effiziente Möglichkeiten zur Minimierung des numerischen Aufwands dar.
- Analog kann eine der Geometrie der zu untersuchenden Struktur angepasste Finite-Element-Formulierung verwendet werden. Dies bietet sich vor allem bei der Rotations-symmetrie an [Wun67]. Gerade bei linear elastischen Problemen sind solche Konzepte überaus effizient. Sie können jedoch auch auf physikalisch und geometrisch nichtlineare Problemstellungen angewendet werden [WCO85, WOSL87].
- Eine weitere Möglichkeit stellt der gezielte Einsatz linearer und nichtlinearer Elemente dar, wenn a priori bekannt ist, in welchen Bereichen nur linear elastische Effekte auftreten. Die Bereiche mit linearem Strukturverhalten können dann mittels geeigneter Maßnahmen vereinfacht bearbeitet werden. Hierzu eignet sich die (dynamische) Substrukturtechnik [Bre96, Goe91, Sch96b, Tem95, Wol88] oder die Randelement-Methode [Kre96, YKV99].
- Die dynamische Untersuchung linear elastischer Systeme mittels der Eigenformmethode (Modalanalyse) führt ebenso zu einer enormen Reduktion des Rechenaufwands [CP93, Gru97, Lin84]. Das dynamische Verhalten des Systems wird hierbei nur noch durch seine charakteristischen Eigenformen und -frequenzen erfaßt. Eine Untersuchung von stark nichtlinearem Verhalten ist jedoch nicht möglich.

Bei nichtlinearem Strukturverhalten sind deshalb Methoden zur umfassenden Reduktion der Berechnung nur bedingt verwendbar. Häufig müssen große Gebiete einer nichtlinearen Untersuchung zugänglich sein, insbesondere wenn das zu ermittelnde Verhalten vor der Berechnung nicht abgeschätzt werden kann oder wenn ohnehin zu erwarten ist, daß diese Gebiete ein nichtlineares Verhalten aufweisen werden.

Für die Simulation von Systemen mit sehr vielen Freiheitsgraden und sehr kleinen Zeitschritten wird eine enorme Rechenzeit von mehreren Tagen oder sogar Wochen benötigt. Darüberhinaus sind aufgrund begrenzter Ressourcen nur Systeme mit einer entsprechend limitierten Anzahl von Freiheitsgraden verarbeitbar [CF88, GCM99]. Eine Leistungssteigerung ist somit bezüglich der Verringerung der Rechenzeit als auch hinsichtlich der Erhöhung des verarbeitbaren Problemumfangs wünschenswert.

Mit dem gleichzeitigen Einsatz mehrerer Prozessoren oder verknüpfter Rechenanlagen (Parallelrechner) kann dieses Ziel erreicht werden [JT98, Kla96, WN93]. Hierzu werden meist

Algorithmen zur effektiven Zerlegung des Gesamtproblems in einzelne Teilbereiche (Gebietszerlegung) untersucht [Boe95, FP90, HA96, Läm97a]. Die Einbindung paralleler Gleichungslöser sowie die *Parallelisierung auf Elementebene* wird derzeit ebenfalls intensiv erforscht [IST97, Kre96, RKR98, WR99].

Die nichtlineare dynamische Untersuchung basiert meist auf Zeitschrittverfahren (Zeitintegrationsverfahren). Schwierigkeiten bereiten bei diesen Untersuchungen neben der enormen Rechenzeit auch numerische Instabilitäten, die eine zuverlässige Simulation stark beeinträchtigen können [BH83, Hug87, ST92]. Es müssen deshalb Verfahren verwendet werden, die eine numerisch stabile und zuverlässige Simulation gewährleisten und gleichzeitig einen minimalen Rechenaufwand benötigen.

Die Simulation gekoppelter Strukturen muß darüberhinaus die Kontaktfuge zwischen unterschiedlichen Körpern erfassen, sofern keine dauerhafte Verbindung gewährleistet werden kann. Die Kontaktsimulation führt zu einem stark nichtlinearen Strukturverhalten und bereitet zusätzliche Probleme. Vor allem treten störende Oszillationen auf, die ebenfalls besondere Maßnahmen bei der Zeitintegration erfordern [AP97, VV90, Wri81].

## 1.2 Ziele und Inhalte der Arbeit

Für die nichtlineare dynamische Untersuchung von Bauwerken und komplexen Systemen benötigt man robuste und effiziente Algorithmen, damit eine numerische Simulation schnell und zuverlässig erfolgen kann. Ziel der vorliegenden Arbeit ist es, diese Methoden darzustellen und sinnvolle oder notwendige Modifikationen aufzuzeigen.

Zunächst werden in **Kapitel 2** die Grundlagen der nichtlinearen dynamischen Finite-Element-Methode erläutert. Es werden wichtige Definitionen und Vorgehensweisen eingeführt.

Algorithmen zur nichtlinearen dynamischen Berechnung werden in **Kapitel 3** behandelt. Einem Überblick über die allgemein verwendeten Verfahren folgt die Herleitung eines impliziten Zeitintegrationsverfahrens. Es wird gezeigt, daß sich diese Methode für viele Probleme in der nichtlinearen Strukturmechanik eignet. Der Algorithmus zeichnet sich durch einen geringen numerischen Aufwand aus und ist ausreichend numerisch robust. Durch eine Modifikation kann er nicht nur für besondere Iterationsverfahren verwendet werden, sondern ist damit auch sehr gut in unterschiedliche Finite-Element-Programme zu implementieren.

Neben einem Vergleich mit anderen Verfahren werden anschließend geeignete Parametrisierungen diskutiert. Es wird außerdem gezeigt, welche Einschränkungen bei der Berechnung von Wellenausbreitungen sowie bei der Kontaktsimulation beachtet werden müssen.

Die Erhöhung der Genauigkeit sowie die Minimierung des numerischen Aufwands wird mit Hilfe einer adaptiven Schrittweitensteuerung in **Kapitel 4** erreicht. Diese basiert auf einem geschätzten Integrationsfehler der Verschiebungen, der auf einen charakteristischen Wert bezogen wird. Zusätzlich wird für Kontaktsimulationen ein Steifigkeitsparameter einbezogen. Schließlich wird dargestellt, wie die Gleichgewichtsiteration bei stark nichtlinearen Berechnungen robust und effizient durchgeführt werden kann.

In **Kapitel 5** wird zur Beschleunigung der numerischen Simulation die parallele Programmierung herangezogen. Es wird ein neues Verfahren vorgestellt (Parallelisierung auf Elementebene), das sich besonders für spezielle Finite-Element-Methoden eignet. Für den flexiblen dynamischen Lastausgleich beim Einsatz auf einem heterogenen Rechner-Cluster wird ein einfach zu implementierendes Verfahren benutzt.

Die Leistungsfähigkeit der verwendeten Algorithmen wird in **Kapitel 6** mit Hilfe von numerischen Simulationen von unverankerten, flüssigkeitsgefüllten Behältern unter Erdbebeneinwirkung demonstriert. Darüberhinaus werden Probleme der Kopplung unterschiedlicher Elemente sowie die Verwendung geeigneter Lastmodelle für die nichtlineare dynamische Berechnung bei Erdbebeneinwirkungen diskutiert.

Anhand einer Parameterstudie werden besondere Effekte und die maßgeblichen Einflußgrößen des dynamischen Tragverhaltens der Behälter sowie die Auswirkungen des mechanischen Verhaltens auf die numerische Simulation gezeigt.

# Kapitel 2

## Nichtlineare Finite-Element-Berechnung

Im folgenden werden die Grundlagen der nichtlinearen Finite-Element-Berechnung kurz umrissen. Eine ausführliche Darstellung – in unterschiedlicher Notation und Vorgehensweise – findet man in der zahlreichen Literatur [Bat86, Bet93, Hug87, NH92, PW94, WR95, Zie84].

### 2.1 Grundlagen der Kontinuumsmechanik

Obwohl bei allen hier betrachteten mechanischen Vorgängen auch thermomechanische Prozesse (vgl. Hauptsätze der Thermodynamik) beteiligt sind, läßt sich zeigen, daß diese im Rahmen dieser Arbeit vernachlässigt werden können [Bet93].

#### 2.1.1 Kinematik

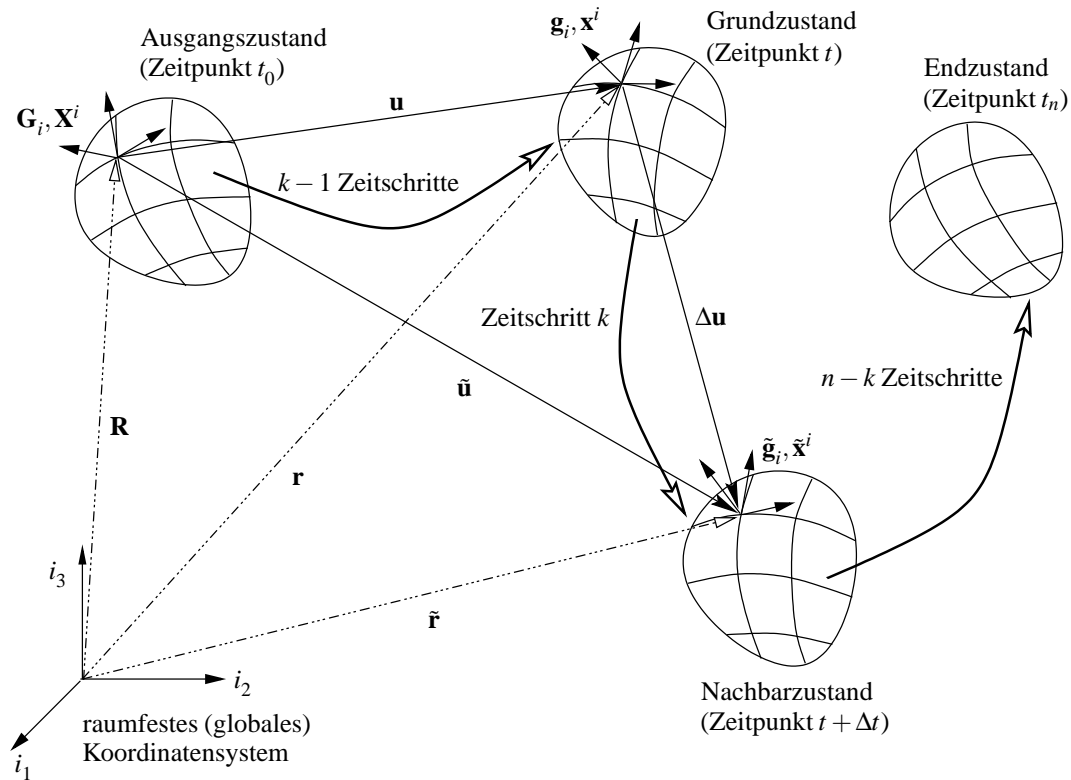
Die mathematische Beschreibung der Bewegung eines Körpers (Kontinuum) im Raum kann durch unterschiedliche Formulierungen erfolgen. Wird ein raumfestes Koordinatensystem verwendet, bei dem der materielle Fluß durch ein stationäres Kontrollvolumen bzw. Gitternetz betrachtet wird, so spricht man im Allgemeinen von einer *Euler-Formulierung*<sup>\*</sup>. Besonders vorteilhaft ist diese Vorgehensweise in der Fluidmechanik, während sich für die Strukturmechanik eine andere Formulierung wesentlich besser eignet: Die *Lagrange'sche*<sup>†</sup> Betrachtungsweise verwendet die Koordinaten des unverformten Körpers im Ausgangszustand zur Beschreibung der Zustandsgrößen (Verschiebungen und deren Ableitungen, Verzerrungen). Somit ist jeder Körperpunkt bei der gesamten Berechnung einem bestimmten Koordinatentripel zugeordnet, während sich die Koordinaten bezogen auf ein raumfestes Koordinatensystem ständig ändern. Auf die Finite-Element-Methode übertragen bedeutet das, daß die im Ausgangszustand gewählten Knoten (Gitterpunkte einer Netzdarstellung) bis zum Endzustand der Berechnung eindeutig einem Materiepunkt zugeordnet bleiben.

Eine Modifikation der *Lagrange'schen* Betrachtung ergibt sich, wenn die Zustandsgrößen nicht auf den Ausgangszustand, sondern auf den letzten errechneten Gleichgewichtszustand (Grundzustand) beziehen. In diesem Falle spricht man von einer *mitgehenden Lagrange* Formulierung. Die weiteren Betrachtungen beziehen sich jedoch auf die *totale Lagrange'sche* Darstellung.

---

<sup>\*</sup>benannt nach *Leonhard Euler*, schweizer Mathematiker, 1707-1783

<sup>†</sup>benannt nach *Joseph Louis de Lagrange*, französischer Mathematiker, 1736-1813



**Abb. 2.1:** Bewegung eines Körpers im Raum zu verschiedenen Zeitpunkten

Die Bewegungen und Verformungen eines Körpers im Raum (Abbildung 2.1) lassen sich für eine inkrementelle Berechnung in vier charakteristische Zustände einteilen:

- Der **Ausgangszustand** zu Beginn der Berechnung (Zeitpunkt  $t_0$ ) stellt den Körper in seiner unbelasteten, unverformten und spannungsfreien Lage dar (Koordinaten  $\mathbf{X}^i$ , Basisvektoren  $\mathbf{G}_i$ , Ortsvektoren  $\mathbf{R}$ ).
- Im **Grundzustand** (Zeitpunkt  $t$ ) sind die Verformungen und Spannungen aus der vorhergehenden Berechnung bekannt (Koordinaten  $\mathbf{x}^i$ , Basisvektoren  $\mathbf{g}_i$ , Ortsvektoren  $\mathbf{r}$ ).
- Der folgende **Nachbarzustand** (Zeitpunkt  $t + \Delta t$ ) stellt den zu ermittelnden Gleichgewichtszustand dar (Koordinaten  $\tilde{\mathbf{x}}^i$ , Basisvektoren  $\tilde{\mathbf{g}}_i$ , Ortsvektoren  $\tilde{\mathbf{r}}$ ).
- Schließlich wird der **Endzustand** (Zeitpunkt  $t_n$ ) erreicht, der dem Nachbarzustand des letzten Zeitschrittes entspricht.

Die Verformungszustände des Körpers werden in globalen Koordinaten beschrieben. Der Ortsvektor des Ausgangszustands  $\mathbf{R}$ , der Vektor des Grundzustands  $\mathbf{r}$  sowie der Vektor des Nachbarzustands  $\tilde{\mathbf{r}}$  stehen mit der Verformung  $\mathbf{u}$  bzw.  $\Delta \mathbf{u}$  in folgender Beziehung:

$$\begin{aligned}
 \mathbf{u} &= \mathbf{r} - \mathbf{R} \\
 \Delta \mathbf{u} &= \tilde{\mathbf{r}} - \mathbf{r} \\
 \tilde{\mathbf{u}} &= \mathbf{u} + \Delta \mathbf{u} = \tilde{\mathbf{r}} - \mathbf{R}
 \end{aligned} \tag{2.1}$$

Aus den partiellen Ableitungen (Notation  $_{,i}$ ) der Ortsvektoren ergeben sich die Basisvektoren:

$$\mathbf{G}_i = \frac{\partial \mathbf{R}}{\partial \mathbf{X}^i} = \mathbf{R}_{,i} \quad (2.2)$$

$$\mathbf{g}_i = \frac{\partial \mathbf{r}}{\partial \mathbf{X}^i} = \mathbf{r}_{,i} = \mathbf{G}_i + \mathbf{u}_{,i} = \mathbf{R}_{,i} + \mathbf{u}_{,i} \quad (2.3)$$

$$\tilde{\mathbf{g}}_i = \frac{\partial \tilde{\mathbf{r}}}{\partial \mathbf{X}^i} = \tilde{\mathbf{r}}_{,i} = \mathbf{g}_i + \Delta \mathbf{u}_{,i} = \mathbf{G}_{,i} + \mathbf{u}_{,i} + \Delta \mathbf{u}_{,i} = \mathbf{G}_i + \tilde{\mathbf{u}}_{,i} \quad (2.4)$$

Über die Metrik des Grundzustandes

$$g_{ij} = G_{ij} + \mathbf{G}_i \mathbf{u}_{,j} + \mathbf{G}_j \mathbf{u}_{,i} + \mathbf{u}_{,i} \mathbf{u}_{,j} \quad (2.5)$$

läßt sich der *Green-Lagrange'sche Verzerrungstensor* des Nachbarzustandes  $\tilde{\epsilon}_{ij}$  herleiten:

$$\tilde{\epsilon}_{ij} = \frac{1}{2} (\tilde{g}_{ij} - G_{ij}) = \frac{1}{2} \mathbf{G}_i \tilde{\mathbf{u}}_{,j} + \mathbf{G}_j \tilde{\mathbf{u}}_{,i} + \tilde{\mathbf{u}}_{,i} \tilde{\mathbf{u}}_{,j} \quad (2.6)$$

Wird nun die übliche tensorielle Schreibweise für die kovariante und die kontravariante Ableitung verwendet, erhält man:

$$\tilde{\epsilon}_{ij} = \frac{1}{2} (\tilde{u}_{i|j} + \tilde{u}_{j|i} + \tilde{u}^k|_j \tilde{u}_{k|i}) \quad (2.7)$$

Einfache Finite-Element-Konzepte gehen von kleinen Verformungen und kleinen Verzerrungen aus. Verläßt man diese Einschränkungen sukzessive, dann lassen sich sehr komplexe Systemzustände beschreiben. Nachdem hierdurch meist keine analytischen Lösungen mehr möglich sind, werden die Grundgleichungen durch eine stückweise Linearisierung einem iterativen Lösungsprozeß zugeführt [Cri91, Cri97, NH92].

## 2.1.2 Gleichgewicht

Definiert man einen Normalenvektor  $\mathbf{n}$ , der auf einer gedachten Schnittebene des Kontinuums senkrecht steht, dann erhält man mit Hilfe des physikalisch meßbaren *Cauchy'schen Spannungstensors*  $\sigma^{ij}$  den Spannungsvektor  $\mathbf{t}$ :

$$\mathbf{t} = \sigma^{ij} n_i \mathbf{g}_j \quad (2.8)$$

Für die zur Herleitung der Finite-Element-Methode verwendeten Arbeitsprinzipie wird jedoch ein dem *Green-Lagrange'schen Verzerrungstensor* konjugierter Spannungstensor benötigt. Hierzu kann der 2. *Piola-Kirchhoff'sche Spannungstensor*  $\mathbf{s}$  verwendet werden:

$$s^{mn} = \frac{\rho_0}{\tilde{\rho}} \frac{\partial X^m}{\partial \tilde{x}^i} \frac{\partial X^n}{\partial \tilde{x}^j} \sigma^{ij} \quad (2.9)$$

Der Term  $\frac{\rho_0}{\bar{\rho}}$  bezeichnet hierbei das Verhältnis der Dichte des Ausgangszustandes bezogen auf den Nachbarzustand.

Auf einen Körper wirken neben den inneren Kräften (Spannungen) auch äußere Kräfte  $\bar{\mathbf{P}}$  sowie Massenkräfte (Trägheitskräfte)  $\mathbf{T}$ . Das Gleichgewicht kann dann gebildet werden aus

$$\sigma^{ij}|_i + \bar{P}^j = 0 \quad (2.10)$$

und

$$t^{ij}|_i + T^j = 0. \quad (2.11)$$

### 2.1.3 Materialgesetz

Mit Hilfe des Materialgesetzes (auch als Stoffgesetz bezeichnet) kann die Beziehung zwischen Verzerrungen und Spannungen hergestellt werden. Unter Annahme von linear elastischem Materialverhalten definiert der Elastizitätstensor  $E^{ijkl}$  diesen Zusammenhang [SB95]:

$$\sigma^{ij} = E^{ijkl} \varepsilon_{kl} \quad (2.12)$$

Für nichtlineare Beziehungen benötigt man ein inkrementelles Stoffgesetz. Hierzu wird der Materialtensor  $C^{ijkl}$  definiert, der sowohl von den vorangegangenen Verzerrungen als auch von den Spannungen beeinflusst wird. Allerdings wird für die einzelnen Inkremente jeweils ein lineares Verhalten zwischen Spannungs- und Verzerrungstensor angenommen (hypoelastisches Materialgesetz):

$$\Delta s^{ij} = C^{ijkl} \Delta \varepsilon_{kl} \quad (2.13)$$

Wird auch die inkrementelle Linearität verlassen, so kann mit Hilfe von speziellen Fließfunktionen eine Beschreibung der Stoffeigenschaften erfolgen [Cri97, Gro95, dBM95].

Auch die Eigenschaften von Kontakt-Fugen können mit entsprechenden Materialformulierungen erfaßt werden, mittels derer sowohl Reibung als auch Separation zwischen zwei oder mehreren Körpern an vordefinierten Übergangsbereichen simuliert werden können (siehe z.B. [Cri97, Tem95]).

## 2.2 Finite-Element-Methode

Die bisherigen Betrachtungen gelten nur für einen infinitesimal kleinen Bereich. Sie lassen sich nur für wenige Ausnahmefälle in eine allgemeingültige Form mit Hilfe von analytischen Lösungen überführen.

Eine globale Formulierung (*schwache Form*) kann aus einer lokalen Darstellung mit Hilfe von Wichtungsfunktionen erfolgen. In der Strukturmechanik lassen sich diese Funktionen als *virtuelle Verschiebungen* deuten. Dabei können sie auch stückweise, also innerhalb eines lokalen Bereiches - dem Finiten Element - definiert werden.



### 2.2.1 Prinzip der virtuellen Verschiebungen

Die Grundlage bilden Energie- bzw. Arbeitsprinzipie. Hierzu werden immer zwei konjugierte Größen benötigt, die miteinander Arbeit leisten. In der Mechanik sind dies Verschiebungen und Kräfte bzw. Verzerrungen und Spannungen. Das System befindet sich im Gleichgewicht, wenn die durch die Variation hervorgerufene Arbeit verschwindet:

$$\delta W = \int_V \delta \tilde{\epsilon}_{ij} \tilde{s}^{ij} dV - \int_V \delta \tilde{u}_j \tilde{f}^j dV + \int_V \delta \tilde{u}_j \rho_0 \tilde{u}^j dV - \int_S \delta \tilde{u}_j \tilde{t}^j dS - \int_S \delta \tilde{u}_j \tilde{t}^j dS = 0 \quad (2.14)$$

Zur Lösung von Gleichung (2.14) muß diese linearisiert und somit einem iterativen Prozeß zugeführt werden. Das Verzerrungsinkrement  $\Delta \epsilon_{ij}$  wird in einen linearen Anteil  $\Delta e_{ij}$  und einen nichtlinearen Teil  $\Delta \eta_{ij}$  aufgeteilt. Unter Einbeziehung des Stofftensors  $C^{ijkl}$  erhält man schließlich die inkrementelle Form:

$$\begin{aligned} & \int_V \delta \Delta e_{ij} C^{ijkl} \Delta e_{kl} dV + \int_V \delta \Delta \eta_{ij} s^{ij} dV = \\ & \int_V \delta \Delta u_j \tilde{f}^j dV + \int_S \delta \Delta u_j \tilde{t}^j dS - \int_V \delta \Delta e_{ij} s^{ij} dV - \int_V \delta \Delta u_j \rho_0 \tilde{u}^j dV \end{aligned} \quad (2.15)$$

Hiermit ist nun die Basis für eine Formulierung der mechanischen Struktureigenschaften bezogen auf Weggrößen geschaffen.

### 2.2.2 Diskretisierung

Die globale Formulierung nach Gleichung (2.15) kann nicht allen Rand- und Anfangsbedingungen gleichzeitig an jedem Punkt des betrachteten Körpers genügen. Deshalb wird sie nur näherungsweise im Mittel erfüllt.

Die Verformungen der Struktur können im Verschiebungsvektor  $\mathbf{u}$  bzw. dessen inkrementeller Darstellung  $\Delta \mathbf{u}$  an diskreten Stellen beschrieben werden. Dazwischen erfolgt eine elementweise Approximation über Ansatzfunktionen  $\mathbf{N}$ :

$$\Delta \mathbf{u} = \mathbf{N} \Delta \hat{\mathbf{u}} \quad (2.16)$$

Bezogen auf Gleichung (2.15) bedeutet das, daß man für jeden Term einen speziellen Anteil erhält: Elementsteifigkeiten, Lastvektoren usw. Die Steifigkeitsmatrix setzt sich aus

$$\mathbf{K}_L + \mathbf{K}_{NL} = \mathbf{K}_T \quad (2.17)$$

zusammen, wobei  $\mathbf{K}_L$  die linear elastischen Steifigkeitsanteile enthält,  $\mathbf{K}_{NL}$  die geometrisch und physikalisch nichtlinearen Anteile und  $\mathbf{K}_T$  stellt schließlich die tangentielle Steifigkeitsmatrix dar. Damit ergibt sich das statische Knotengleichgewicht der gesamten Struktur zu:

$$\mathbf{K}_T \Delta \mathbf{u} = \Delta \bar{\mathbf{P}} \quad (2.18)$$

Diese inkrementelle Strukturgleichung muß nun einem iterativen Lösungsprozeß zugeführt werden, da  $\mathbf{K}_T$  von Zustandsgrößen abhängig ist, die sich von Inkrement zu Inkrement ändern [WR95].

Alternativ können die nichtlinearen Anteile auch als Pseudo-Lasten  $\Delta \mathbf{P}_{NL}$  in die Berechnung eingehen (“Iteration über die rechte Seite”):

$$\mathbf{K}_L \Delta \mathbf{u} = \Delta \bar{\mathbf{P}} - \Delta \mathbf{P}_{NL} \quad (2.19)$$

Die Diskretisierung hat einen wesentlichen Einfluß auf die Genauigkeit und den numerischen Aufwand der Berechnung. Die Erhöhung der Qualität der Ansatzfunktionen (Gleichung (2.16)) sowie die Verkleinerung der Elementabmessungen verbessert die Genauigkeit, führt jedoch auch zu einem größeren numerischen Aufwand. Mit Hilfe einer adaptiven Steuerung der Elementgröße (h-Adaption) bzw. der Ansatzfunktionen (p-Adaption) basierend auf a priori oder a posteriori Fehlerschätzern kann die Genauigkeit der Berechnung insgesamt auf ein gewünschtes Maß gebracht werden. Der Rechenaufwand steigt damit im Vergleich zu einer uniformen FE-Netzverfeinerung nur moderat.

Die Netzadaption stellt somit ein wichtiges Werkzeug bei der numerischen Berechnung großer Strukturen dar. Im Rahmen dieser Arbeit wird dieser Aspekt nicht weiter verfolgt, weshalb hier auf die entsprechende Literatur verwiesen sei (z.B.: [CRSW96, Ric97, Kla96, RRDB99, SSR94]).

### 2.2.3 Sonderfall: Rotationssymmetrische Finite Elemente

Die Methoden und Algorithmen der vorliegenden Arbeit sind weitgehend unabhängig von der verwendeten Elementformulierung – wo dies nicht der Fall ist, wird besonders darauf hingewiesen. Die detaillierte Herleitung der hier dargestellten Element-Formulierung findet man in den Arbeiten von Wunderlich et al [Wun67, WCO85, WST94, Tem95, WST92, WCST95], weshalb hier nur die methodische Vorgehensweise skizziert wird. Eine kurze Beschreibung der mechanischen Eigenschaften der verwendeten Elemente ist in Abschnitt 6.2 dargestellt.

Die Diskretisierung rotationssymmetrischer Strukturen kann äußerst effizient mit Hilfe von *Ring-Elementen* erfolgen. Da im Rahmen der vorliegenden Arbeit überwiegend solche Strukturen berechnet werden, kommen ausschließlich *semi-analytische Ringelemente* zum Einsatz. Unter Ausnutzung der Rotationssymmetrie erfolgt die geometrische Beschreibung mit Polarkoordinaten. Die Zustandsgrößen  $\mathbf{z}$  werden in Umfangsrichtung mit Hilfe von Fourierreihen dargestellt (vgl. Abbildung 2.2). Die Zustandsgrößen werden also aus einer Superposition der einzelnen Fourier-Reihenglieder ermittelt.

Damit die Harmonischen auch bei nichtlinearen Berechnungen entkoppelt sind, werden alle geometrischen und physikalischen Nichtlinearitäten in einem Pseudo-Lastvektor zusammengefaßt. Die jeweiligen Anteile werden auf Elementebene ermittelt (siehe Abschnitt 5.3.2). Dieses Vorgehen ist im Linearen äußerst effizient, wird jedoch bei stark nichtlinearem Strukturverhalten durch eine steigende Anzahl von Iterationsschritten bei der Gleichgewichtsiteration beeinträchtigt.

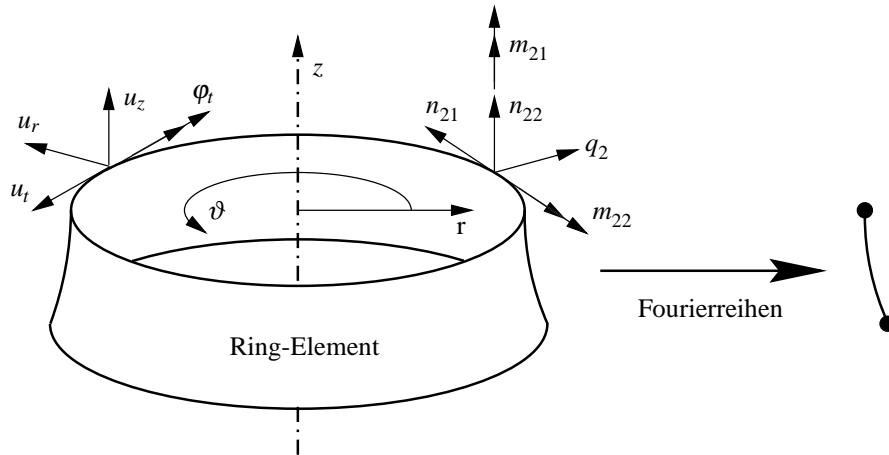


Abb. 2.2: Semi-analytische Ring-Elemente (Schalen-Element)

## 2.3 Nichtlineare Gleichungslösung

Die Lösung des nichtlinearen Gleichungssystems (2.18) stellt einen zentralen Punkt in der numerischen Simulation dar. Grundlage aller Verfahren ist die stückweise Linearisierung des nichtlinearen Gleichungssystems, d.h. aus dem bekannten Grundzustand soll der noch unbekannte Nachbarzustand iterativ ermittelt werden.

Da im Rahmen der vorliegenden Arbeit nur dynamische Berechnungen betrachtet werden, können keine Verfahren verwendet werden, die das Lastniveau sukzessive verändern (Bogenlängenverfahren) oder Verfahren, die verschiebungsgesteuert sind. Es kommen deshalb nur Algorithmen in Frage, bei denen im Prädiktorschritt die volle Last des aktuellen Zeitschrittes aufgebracht und in der Korrektorphase hierzu ein Gleichgewichtszustand gefunden wird.

Die numerische Effektivität der Verfahren hängt neben der Art und der Anzahl der notwendigen Matrizenoperationen auch von der Konvergenzordnung und damit von der Anzahl der erforderlichen Iterationsschritte ab.

### 2.3.1 Newton-Raphson-Verfahren

Das *Newton-Raphson-Verfahren* stellt eine weit verbreitete Methode zur iterativen Gleichgewichtsermittlung dar. Der Algorithmus wird in der Literatur ausführlich beschrieben (z.B.: [Cri91]). Zur Herleitung der Methode wird Gleichung (2.18) in eine Taylorreihe entwickelt, die nach dem linearen Glied abgebrochen wird. Nach [Bat86] entsteht daraus folgende inkrementelle Beziehung:

$${}^{t+\Delta t}\mathbf{K}_T^{i-1} \Delta \mathbf{u}^i = {}^{t+\Delta t}\bar{\mathbf{P}} - {}^{t+\Delta t}\mathbf{F}^{i-1} \quad (2.20)$$

Hierbei gibt  $i$  den aktuellen Iterationsschritt an (Zählvariable),  ${}^{t+\Delta t}\mathbf{K}_T^{i-1}$  stellt die Tangentialsteifigkeitsmatrix und  ${}^{t+\Delta t}\mathbf{F}^{i-1}$  stellt die inneren (Knoten-) Kräfte der Struktur jeweils zum

vorhergehenden Iterationsschritt dar. Schließlich bezeichnet  ${}^{t+\Delta t}\bar{\mathbf{P}}$  die äußeren (Knoten-) Lasten der Struktur. Die rechte Seite von Gleichung (2.20) kann mechanisch gesehen als nicht ausbalancierter Lastvektor bzw. als Residuum  $\Delta\mathbf{R}^i$  gedeutet werden.

Der Gesamtverschiebungsvektor wird sukzessive nach jedem Inkrement mit Hilfe der Lösung von Gleichung (2.20) verbessert:

$${}^{t+\Delta t}\mathbf{u}^i = {}^{t+\Delta t}\mathbf{u}^{i-1} + \Delta\mathbf{u}^i \quad (2.21)$$

Dieser Iterationsprozeß wird bis zum Erreichen einer vorgegebenen Genauigkeit durchgeführt, wobei als Iterationsschranken verschiedene Kriterien dienen können (siehe Abschnitt 2.3.3).

Der Vorteil dieses Verfahrens – die quadratische Konvergenzordnung – geht leider mit einem entscheidenden Nachteil einher: Die Berechnung der tangentialen Steifigkeitsmatrix führt insbesondere bei großen Systemen zu enormen Rechenzeiten. Mit der Verwendung der Anfangssteifigkeitsmatrix  ${}^{t=0}\mathbf{K}$  (*Anfangsspannungsmethode*) kann zwar dieses Problem umgangen werden, der Iterationsvorgang konvergiert dadurch aber sehr langsam und kann sogar zu einer divergenten Lösung führen [Bat86].

Einen Mittelweg zwischen Tangentialsteifigkeit und Anfangssteifigkeit stellt das *modifizierte Newton-Raphson-Verfahren* dar, bei dem die Steifigkeitsmatrix während der Berechnung nur manchmal (z.B. zu jedem Zeitpunkt  $t$ ) neu berechnet wird und während der Gleichgewichtsiteration unverändert bleibt. Bei stark nichtlinearem Strukturverhalten kann allerdings auch mit Hilfe begleitender Maßnahmen (*Zeitschrittverkürzung*, *Aitkenbeschleunigung*) nicht immer eine konvergente Lösung garantiert werden [Cri91].

### 2.3.2 Quasi-Newton-Verfahren

*Quasi-Newton-Algorithmen*, die einen Kompromiß zwischen modifiziertem und vollständigem *Newton-Raphson-Verfahren* darstellen, verändern die (inverse) Steifigkeitsmatrix ohne sie vollständig neu zu berechnen. Aus dieser Veränderung entsteht eine Sekantensteifigkeitsmatrix, also eine Annäherung an die eigentliche Tangentialsteifigkeit. Die bekannteste Methode ist in diesem Kontext die *BFGS-Methode* (benannt nach *Broyden*, *Fletcher*, *Goldfarb* und *Shanno*), die in [MS79] erstmals hergeleitet wird (siehe auch [Bat86] und [Cri91]). Hauptvorteil dieser Methode ist der relativ geringe numerische Aufwand, da nur *Rang-2 Korrekturen* vorgenommen werden.

Der Nachteil des Verfahrens besteht in der zusätzlichen Speicherung der update-Vektoren, was bei vielen Iterationen zu einem enormen Speicherplatzbedarf führen kann. Dies kann durch *limited memory quasi-Newton updates* vermieden werden [PP93].

Bei [Tem95] findet man eine modifizierte Version, bei der mit Hilfe von Pseudo-Lasten nicht-lineare Anteile erfaßt werden. Dieses Verfahren wird auch in den numerischen Berechnungen der vorliegenden Arbeit verwendet (siehe Kapitel 6).



- Der Zuwachs der nicht ausbalancierten Last (Residuum)  $\delta \mathbf{R}^i$
- Der Zuwachs an innerer Energie  $\delta E_{in}^i$

Bei einer konvergenten Lösung streben die Normen aller drei Parameter gegen Null. Die absoluten Werte können infolge der unterschiedlichen Einheiten sowie der Abhängigkeit bezüglich der Anzahl der Freiheitsgrade nur sehr schwer miteinander verglichen werden. Insbesondere kann kein absoluter Grenzwert angegeben werden.

Mit Hilfe von relativen Kriterien können unabhängig von der aktuellen Problemstellung Grenzwerte a-priori festgelegt werden. Als Bezugsgröße sind immer Werte festzulegen, die nicht gegen Null streben können. Dies ist z.B. bei der Norm des aktuellen Verschiebungsincrementes und bei der Norm des aktuellen Gesamtlastvektors möglich.

Auch ist zu beachten, daß die Vektoren teilweise unterschiedliche mechanische Größen beinhalten: Der Verschiebungsvektor enthält Verschiebungen und Verdrehungen, der Lastvektor enthält Kräfte und Momente. Bei der Berechnung einer Norm (z.B.  $L_2$ -Norm) werden dann unterschiedliche Größen miteinander verarbeitet [Bul93].

Abhilfe schaffen kann hier ein Aufspalten der Vektoren in zwei Vektoren gleicher mechanischer Größen, wobei auch die jeweilige Bezugsgröße entsprechend behandelt wird. Dies setzt jedoch voraus, daß mit Hilfe der Inzidenztafeln eine entsprechende Zuordnung möglich ist.

Die relativen Fehler ergeben sich somit zu:

$$\frac{\|\delta \mathbf{u}^i\|_2}{\|\mathbf{u}_{ref}\|_2} < \varepsilon_u \quad (\text{Verschiebung}) \quad (2.26)$$

$$\frac{\|\delta \mathbf{R}^i\|_2}{\|\mathbf{R}_{ref}\|_2} < \varepsilon_R \quad (\text{nicht ausbalancierte Lasten}) \quad (2.27)$$

$$\frac{\delta E_{in}^i}{E_{ref}} < \varepsilon_E \quad (\text{innere Energie}) \quad (2.28)$$

Als Referenzwert für die Verschiebung wird meist der aktuelle Verschiebungsvektor des letzten Iterationsschrittes  $\mathbf{u}^i$  verwendet [Bat86]. Um die oben angesprochene Problematik (Referenzwert strebt gegen Null) zu umgehen, wird hier eine Alternative vorgeschlagen:

$$\|\mathbf{u}_{ref}\|_2 = \frac{w\|\mathbf{u}^i\|_2 + \|\mathbf{u}_{max}\|_2}{w+1} \quad \text{wobei } w \geq 1 \quad (2.29)$$

Der Wert  $w$  dient als Wichtungparameter. Nach Erfahrung des Autors erzielt man mit  $w = 3$  gute Ergebnisse. Hierdurch ergibt sich für die aktuelle Berechnung ein charakteristischer Wert, der mit Sicherheit nicht gegen Null strebt. Analog kann mit den Referenzgrößen für die nicht ausbalancierten Lasten  $\mathbf{R}_{ref}$  sowie der inneren Energie  $E_{ref}$  verfahren werden.

# Kapitel 3

## Berechnung im Zeitbereich

### 3.1 Grundgleichungen

Die numerische Untersuchung des nichtlinearen dynamischen Strukturverhaltens eines komplexen Bauwerkes erfordert neben einer räumlichen auch eine zeitliche Diskretisierung. Mathematisch gesehen handelt es sich bei der Diskretisierung des Raumes um ein Randwertproblem, bei der Diskretisierung in der Zeit um ein Anfangswertproblem. Werden beide Probleme simultan in Raum und Zeit gelöst, so wird dies mittels sogenannter Raumzeit-Elemente durchgeführt (siehe Abschnitt 3.2.6).

Hier wird jedoch eine sequentielle Vorgehensweise verwendet, bei der zunächst das Randwertproblem und anschließend das Anfangswertproblem formuliert wird. Das Randwertproblem kann hierbei durch problemangepasste Finite-Element-Formulierungen erfaßt werden (vgl. Kapitel 2).

Zur Lösung des Anfangswertproblems muß die räumliche Formulierung um die Massen- und Dämpfungsterme erweitert werden (*d' Alembertsches Prinzip*, Abbildung 3.1). Hierzu wird ebenfalls die Finite-Element-Methode verwendet (siehe Abschnitt 2.2).

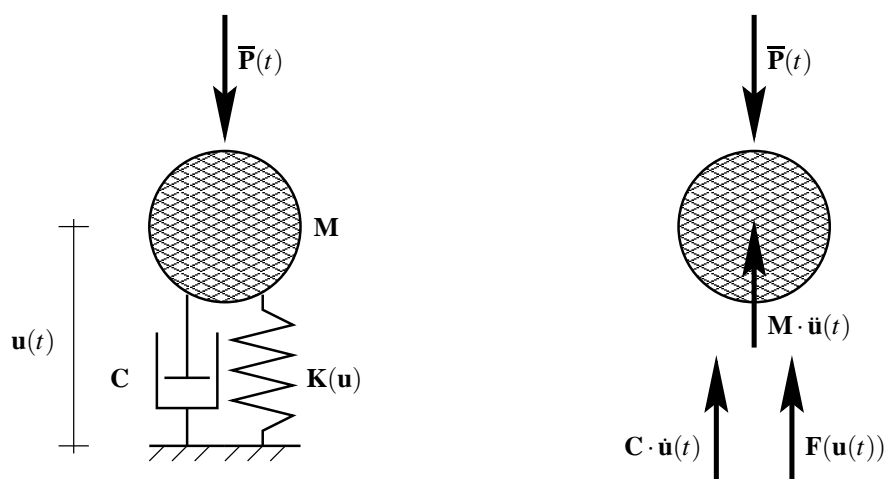


Abb. 3.1: Idealisiertes System zum *d' Alembertschen Prinzip*

Aus dieser Synthese erhält man durch Bildung des Gleichgewichts (vgl. Abschnitt 2.1.2) ein

nichtlineares algebraisches Gleichungssystem:

$$\mathbf{M} \cdot \ddot{\mathbf{u}}(t) + \mathbf{C} \cdot \dot{\mathbf{u}}(t) + \mathbf{F}(\mathbf{u}(t)) = \bar{\mathbf{P}}(t) \quad (3.1)$$

wobei  $\mathbf{F}(\mathbf{u})$  die inneren Kräfte der Struktur,  $\mathbf{M}$  die konstante Massenmatrix und  $\mathbf{C}$  die Dämpfungsmatrix darstellt\*.

Ziel ist nun die Ermittlung des Verschiebungsvektors  $\mathbf{u}(t)$  sowie seiner Zeitableitungen  $\dot{\mathbf{u}}(t)$  (Geschwindigkeitsvektor) und  $\ddot{\mathbf{u}}(t)$  (Beschleunigungsvektor) zu allen Zeitpunkten  $t$ .

Bei linear-elastischen Systemen wird angenommen, daß  $\mathbf{K}$  konstant ist. Somit gilt:

$$\mathbf{K} \cdot \mathbf{u}(t) = \mathbf{F}(\mathbf{u}(t)) \quad (3.2)$$

Daraus ergibt sich folgendes Differentialgleichungssystem 2. Ordnung in der Zeit [CP93]:

$$\mathbf{M} \cdot \ddot{\mathbf{u}}(t) + \mathbf{C} \cdot \dot{\mathbf{u}}(t) + \mathbf{K} \cdot \mathbf{u}(t) = \bar{\mathbf{P}}(t) \quad (3.3)$$

Für die Lösung des Gleichungssystems stehen äußerst effiziente Methoden zur Verfügung (Abschnitt 3.2.2). Es können somit auch sehr große Systeme mit einem relativ geringen numerischen Aufwand bearbeitet werden.

Die Dämpfung  $\mathbf{C}$  kann sich während des zu untersuchenden Zeitraums infolge plastischer Materialeigenschaften verändern, weshalb sie bei diesen Berechnungen durch implizite Materialformulierungen ersetzt werden muß. Insbesondere verändert sich aber die Steifigkeit  $\mathbf{K}$  des Gesamtsystems bei geometrisch und physikalisch nichtlinearen Systemen stark und kann Werte zwischen der linearen Ausgangssteifigkeit und nahezu verschwindender Steifigkeit annehmen.

Nimmt man nun die Steifigkeit in Abhängigkeit von der Verschiebung an und ersetzt die Dämpfungsmatrix durch das nichtlineare Materialverhalten, so lautet das Differentialgleichungssystem:

$$\mathbf{M} \cdot \ddot{\mathbf{u}}(t) + \mathbf{F}(\mathbf{u}(t)) = \bar{\mathbf{P}}(t) \quad (3.4)$$

Die Gleichung (3.4) stellt das gekoppelte Differentialgleichungssystem 2. Ordnung für die nichtlineare Bewegungsgleichung dar. Das Gleichungssystem kann nun generell nicht mehr vereinfacht gelöst werden. Es muß ein Verfahren gewählt werden, mit dem die Lösung numerisch ermittelt werden kann. Hierzu wird das zu untersuchende Zeitintervall  $[t_0, t_n]$  in  $n$  Stützstellen eingeteilt. Das Differentialgleichungssystem wird anschließend numerisch ausgewertet. Dieses Vorgehen wird allgemein als *Zeitintegration* oder *direkte numerische Integration* bezeichnet.

## 3.2 Zeitintegrationsverfahren – Überblick

Der folgende Überblick beinhaltet alle gängigen Verfahren zur Berechnung von dynamischen Systemen, also auch Verfahren, die auf lineare Systeme beschränkt sind.

\*Beim Einmassenschwinger reduzieren sich die Matrizen auf Skalare



### 3.2.1 Einschrittverfahren – Mehrschrittverfahren

Zeitintegrationsverfahren, bei denen lediglich der zuletzt ermittelte Gleichgewichtszustand (Zeitpunkt  $t$ ) zur Berechnung des neuen Zustandes (Zeitpunkt  $t + \Delta t$ ) verwendet wird, werden als *Einschrittverfahren* bezeichnet.

Werden dagegen mehrere Zeitpunkte miteinbezogen, so spricht man von einem *Mehrschrittverfahren* (*Linear Multistep Method*). Bekannte Mehrschrittverfahren sind *Houbolt's Methode*, die *Wilson- $\Theta$  Methode* und die Methode nach *Park* [Hug87].

Da diese Verfahren einige gravierende Nachteile besitzen (z.B.: aufwendiger Algorithmus, ungünstiges numerisches Dämpfungsverhalten, sehr großes *Overshooting* – siehe hierzu die ausführliche Beschreibung in [Hug87]), werden sie im Rahmen dieser Arbeit nicht weiter betrachtet. Im Folgenden werden also nur *Einschrittverfahren* untersucht.

### 3.2.2 Indirekte Verfahren

Bei indirekten Verfahren wird das Differentialgleichungssystem 2. Ordnung erst nach einer Umformung numerisch gelöst. Durch diese Umformung kann die ursprüngliche Struktur des Gleichungssystems völlig verändert werden. Die Anzahl der zu lösenden Gleichungen kann entweder reduziert werden (Modale Verfahren) oder stark anwachsen (Transformation in ein Differentialgleichungssystem 1. Ordnung).

#### Modale Verfahren

Insbesondere bei Systemen mit sehr vielen Freiheitsgraden ist es wünschenswert, die Anzahl der Unbekannten zu verringern, damit sich der Rechenaufwand im selben Maße verkleinert. Bei vielen baupraktischen Untersuchungen stellt man fest, daß bei schwingenden Systemen meist die niedrigen Frequenzen dominieren [Pfa89]. Dies führt zu der Überlegung, daß eine Reduktion des Ausgangssystems mit sehr vielen Eigenfrequenzen auf ein System mit wenigen charakteristischen Frequenzen eine enorme Rechenvereinfachung darstellt.

Da das Verfahren auf einer Eigenwertanalyse zur Bestimmung der Eigenformen beruht, muß vorausgesetzt werden, daß sowohl die Massen- als auch die Steifigkeitsmatrix konstant sind [Gru97]. Dies ist jedoch nur bei linear elastischen Systemen der Fall.

Es gibt allerdings auch Versuche, diese oder ähnliche Vorgehensweisen bei nichtlinearen Systemen mittels einer stückweisen Linearisierung anzuwenden [Goe91]. Hierzu ist immer dann eine aufwendige Transformation vom Ausgangssystem zum reduzierten System notwendig, wenn sich die Struktureigenschaften infolge von Nichtlinearitäten wesentlich ändern. Gerade bei stark nichtlinearen Berechnungen ist dies mit einem erheblichen Aufwand verbunden, da sich die dynamischen Eigenschaften sehr oft stark verändern. Insbesondere ist dies bei Kontaktproblemen der Fall.

### Transformation in ein Differentialgleichungssystem 1. Ordnung

Eine andere Möglichkeit ist die Transformation des bestehenden Differentialgleichungssystems 2. Ordnung (Gleichung 3.3) in ein System 1. Ordnung. Das System 1. Ordnung kann z.B. mit *Runge-Kutta-Verfahren* [Gea71] numerisch gelöst werden. Während die eigentliche numerische Integration mit äußerst robusten und gut erprobten Verfahren erfolgen kann, ist die Transformation vom System 2. Ordnung zum System 1. Ordnung mit dem Nachteil verbunden, daß sich hierdurch die Anzahl der Unbekannten und damit auch die Anzahl der Gleichungen verdoppelt. Außerdem erhält man vollbesetzte, unsymmetrische System-Matrizen, die nur mit speziellen Gleichungslösern behandelt werden können. Deshalb ist dieses Vorgehen nur für Systeme mit wenigen Freiheitsgraden sinnvoll.

Für dynamische Stabilitätsuntersuchungen leitet Redanz in [Red87] ein iteratives Verfahren her. Hierbei wird mit dem transformierten Differentialgleichungssystem 1. Ordnung eine Eigenwertuntersuchung der stückweise linearisierten Strukturgleichung durchgeführt. Dieses Verfahren ist jedoch sehr aufwendig und für dynamische Untersuchungen im Sinne der vorliegenden Arbeit nicht einsetzbar. Allerdings zeigt Redanz anhand einer Vergleichsrechnung, daß auch mit dem *Newmark* Verfahren dynamische Instabilitäten erfaßt werden können.

#### 3.2.3 Explizite Verfahren

Wird das Gleichgewicht zum Zeitpunkt  $t$  formuliert, also zu Beginn des jeweiligen Zeitinkrements, handelt es sich um ein explizites Verfahren. Dabei wird diese Art von Integrationsverfahren bevorzugt da verwendet, wo die Massenmatrix in diagonalisierter Form vorliegt – das Gleichungssystem entkoppelt sich und kann äußerst effektiv gelöst werden [CL94].

Leider sind explizite Verfahren nur bedingt numerisch stabil und müssen deshalb auf die höchsten Eigenformen der zu untersuchenden Struktur mit der Zeitschrittlänge angepaßt werden, was zu extrem kurzen Schritten führen kann. Deshalb werden diese Verfahren bevorzugt bei sehr kurzzeitigen Vorgängen wie Crash- oder Impact-Simulationen eingesetzt [Eib95].

Bei der Wahl des Parameters  $\beta = 0$  im *Newmark* Verfahren [New59], erhält man ein explizites Verfahren, allgemein als *zentrales Differenzenverfahren* bezeichnet [Hug87]\*.

#### 3.2.4 Implizite Verfahren

Bei impliziten Verfahren wird die Bewegungsgleichung zum Ende des jeweiligen Zeitschrittes ( $t + \Delta t$ ) in den Zeitintegrationsalgorithmus eingebracht. Diese große Gruppe von Verfahren ist in der Strukturdynamik sehr weit verbreitet, da sie insbesondere bei Berechnungen mit überwiegend niederfrequenten Strukturantworten äußerst effektiv und robust sind. Aber auch die sehr gute Einbindung von Elementformulierungen verschiedenster Art, wie sie bei der Berechnung gekoppelter Systeme auftreten, zeichnet implizite Verfahren aus, da sie von der eigentlichen Elementformulierung meist völlig entkoppelt sind.

\*Die Definition der verwendeten Bezeichnungen findet man in Abschnitt 3.3, Seite 20 ff

Nicht zuletzt ist der geringe numerische Aufwand dieser Algorithmen vor allem bei Systemen mit sehr vielen Freiheitsgraden zu nennen. Aus diesen Gründen werden im Rahmen der vorliegenden Arbeit implizite Verfahren eingehend betrachtet. Insbesondere stehen hierbei die sogenannten *Newmark* Verfahren im Mittelpunkt, die auf der Arbeit von *N. M. Newmark* beruhen [New59] und bis heute weiterentwickelt werden [HHT77, WBZ81, ZWH84, CH93]. Auch für nichtlineare Untersuchungen eignen sich diese Verfahren teilweise sehr gut. Es treten hierbei jedoch besondere Probleme auf, die in Kapitel 3.5 ausführlich diskutiert werden.

### 3.2.5 Energieerhaltende Verfahren

Die oben beschriebenen Verfahren können durch zusätzliche Maßnahmen oder implizit durch eine erweiterte Formulierung die Energie im Gesamtsystem vom Zeitpunkt  $t$  auf den Zeitpunkt  $t + \Delta t$  exakt übertragen. Man erhält dadurch numerisch äußerst robuste Algorithmen, die vor allem bei geometrisch nichtlinearen Berechnungen mit großen Verformungen verwendet werden. Dieses Vorgehen ist per Definition nur bei Systemen zulässig, bei denen kein Energieverlust\* im mechanischen Sinne auftritt.

Die Formulierung dieser Verfahren erfordert eine enge Kopplung von Zeitintegration und Elementformulierung, da beispielsweise die Ermittlung der aktuellen Gesamtenergie auch die Verzerrungsenergie auf Elementebene erfassen muß.

Die noch sehr neuen Verfahren basieren im wesentlichen auf Arbeiten, die um 1977 entstanden sind [NNDR77, HCL78] sowie neueren Veröffentlichungen [ST92, ST94, ST95, AP97, KR96, CJ98, KC99]. Diese Verfahren könnten auch für dissipative Vorgänge verwendet werden, wenn man die zu dissipierende Energie in jedem Zeitschritt a priori ermitteln könnte. Bisher ist ein solcher Ansatz nur für sehr spezielle Anwendungen gefunden worden (z.B. Reibkontakt) [CL98]. Deshalb wird diese Gruppe in der vorliegenden Arbeit nicht weiter untersucht, da neben physikalischer Materialdämpfung auch geometrische Dämpfung (z.B. infolge Wellenausbreitung in den Halbraum) untersucht werden soll.

Eine ähnliche Ausrichtung haben Verfahren, die nicht die Energieerhaltung erzwingen, sondern nur die Impulserhaltung. Hier liegen auch nur für elastische Probleme gesicherte Erkenntnisse bezüglich der numerischen Eigenschaften vor [AR99].

Derzeitiges Haupteinsatzgebiet energieerhaltender Verfahren ist die geometrisch hochgradig nichtlineare Elastodynamik. Als Referenzbeispiel wird meist ein frei im schwerelosen Raum fliegender Körper verwendet.

### 3.2.6 Raum–Zeit–Diskretisierung

Eine andere Vorgehensweise zur Diskretisierung im Zeitbereich stellen Verfahren dar, bei denen auch im Zeitbereich ein Finite-Element-Ansatz gewählt wird (Raum-Zeit-Diskretisierung) [HH88, HH90]. Für jedes Zeit-Inkrement werden dann zugleich für den Anfang

---

\*Energieverlust bedeutet hier eigentlich Dämpfung der dynamischen Struktur

und das Ende die jeweiligen Zustände mit Hilfe von Wichtungsfunktionen ermittelt. An den diskreten Zeitpunkten können Sprünge der Zustandsgrößen auftreten.

Eines der wichtigsten Vertreter dieser Gruppe ist das *Time-Diskontinuous-Galerkin (TDG)* Verfahren [WL96, Li96, WL99]. Diese Verfahren sind numerisch robust und es wird eine äußerst effektive *numerische Dissipation*\* ermöglicht [Hul94]. Darüberhinaus können mit dieser Methode auch Probleme im Zeitbereich behandelt werden, bei denen nicht dynamische sondern z.B. Effekte poröser Medien abgebildet werden [CFW99].

In der Dynamik werden sie derzeit vor allem bei der Berechnung von Wellenausbreitungs-Problemen mit Hilfe einer Adaption von Raum und Zeit eingesetzt. Hierbei kann das Maß der Sprünge der Zustandsgrößen (Verschiebungen, Geschwindigkeiten) als Fehlerindikator dienen [WL98]. Allerdings verdeutlichen die Beispielrechnungen in [NS99, WEL99] einen wichtigen Nachteil dieser Methoden: Die Form einer Welle wird relativ stark verfälscht. Vor allem rechteckförmige Wellen oder Sinushalbwellen werden an den "Ecken" stark abgerundet. Die Ursache für dieses Verhalten liegt in der starken Dissipation hoher Frequenzen begründet (siehe hierzu auch die Abbildungen 3.7 und 3.8 auf Seite 29 sowie die Diskussion auf Seite 33 ff und in Abschnitt 3.5.1).

Die gleichzeitige Raum–Zeit–Diskretisierung führt zu einem erheblichen numerischen Mehraufwand, da sich die Anzahl der Unbekannten verdoppelt bzw. bei einem zusätzlichen Ansatz der Geschwindigkeiten sogar vervierfacht. Das resultierende Gleichungssystem ist unsymmetrisch und weist keine Bandstruktur auf. Für sehr große FE–Systeme ist dies jedoch ein entscheidender Nachteil [NS99, WEL99].

Eine Reduktion der Anzahl der Unbekannten kann nach [WGK99] dadurch erfolgen, daß die Verschiebungen ohne Diskontinuitäten angesetzt werden. Lediglich bei den Geschwindigkeiten werden Diskontinuitäten zugelassen. Allerdings erhält man trotzdem sehr breitbandige, unsymmetrische Matrizen.

### 3.3 Modifizierter impliziter Zeitintegrationsalgorithmus

In Abschnitt 3.2.4 (Seite 18) wurde bereits erläutert, weshalb im Rahmen dieser Arbeit implizite Einschritt-Verfahren verwendet werden. Im folgenden wird zunächst die von Chung und Hulbert in [CH93] für lineare Anwendungen erstmals veröffentlichte *Generalized- $\alpha$*  Methode kurz dargestellt. Anschließend folgt eine an [Kuh96] angelehnte Herleitung für nichtlineare Probleme, wobei einige Änderungen vorgenommen werden.

Zunächst seien die wesentlichen Anforderungen an einen Zeitschritt-Algorithmus dargestellt:

- Unbedingte spektrale Stabilität (*unconditionally stable*) bei linearen Berechnungen.
- Energetische (numerische) Stabilität bei nichtlinearen Berechnungen.

---

\*Begriffserklärung: Siehe Abschnitt 3.4.3, Seite 27

- Physikalische und geometrische Nichtlinearitäten müssen einbezogen werden können. Es muß also eine beliebige physikalische Dämpfung möglich sein.
- Steuerbare numerische Dissipation sehr hoher Oszillationen zur Stabilisierung bei nicht-linearer Berechnung und zur Verwendung möglichst langer Zeitschritte.
- Der numerische Aufwand soll möglichst gering sein.
- Simultane Anwendung auf möglichst viele verschiedene Finite-Element-Formulierungen, um gemischte (gekoppelte) Systeme berechnen zu können.
- Möglichkeit zur Parallelisierung der Finite-Element-Berechnung

Ausgehend von diesen Überlegungen soll nun ein Verfahren weiterentwickelt werden, das möglichst alle genannten Punkte erfüllt. Grundgedanke des *Generalized- $\alpha$*  Verfahrens ist, daß die Verschiebungen, deren zeitliche Ableitungen sowie die äußeren Belastungen nicht am Ende des aktuellen Zeitinkrements in die Berechnung eingebracht werden. Stattdessen werden sie durch eine Linearkombination aus Anfangszustand und Endzustand der jeweiligen Größen ersetzt. Während die Beschleunigung mit dem Faktor  $\alpha_m$  skaliert wird, geschieht dies bei den restlichen Termen mit dem Faktor  $\alpha_f$ :

$$\begin{aligned}
& \mathbf{M}[(1 - \alpha_m)^{t+\Delta t} \ddot{\mathbf{u}} + \alpha_m {}^t \ddot{\mathbf{u}}] \\
& + \mathbf{C}[(1 - \alpha_f)^{t+\Delta t} \dot{\mathbf{u}} + \alpha_f {}^t \dot{\mathbf{u}}] \\
& + \mathbf{K}[(1 - \alpha_f)^{t+\Delta t} \mathbf{u} + \alpha_f {}^t \mathbf{u}] \\
& = {}^{t+\Delta t} \bar{\mathbf{P}} (1 - \alpha_f)^{-t} \bar{\mathbf{P}} \alpha_f
\end{aligned} \tag{3.5}$$

Bei nichtlinearen Berechnungen, wenn statt der Steifigkeitsmatrix  $\mathbf{K}$  der Vektor der inneren Kräfte  $\mathbf{F}(\mathbf{u})$  verwendet wird, kann dieser analog Gleichung (3.5) ebenso durch eine Linearkombination in das Rechenverfahren eingebracht werden:

$$\mathbf{F}({}^t \mathbf{u}) = \mathbf{F}({}^{t+\Delta t} \mathbf{u})(1 - \alpha_f) - \mathbf{F}({}^t \mathbf{u})\alpha_f \tag{3.6}$$

Sei nun angenommen, daß man die Verschiebungen zum Zeitpunkt  $t + \Delta t$  kennt, so kann man die Geschwindigkeiten und Beschleunigungen aus den *Newmark*-Ansätzen berechnen [New59]:

$${}^{t+\Delta t} \dot{\mathbf{u}} = \frac{\gamma}{\beta \Delta t} ({}^{t+\Delta t} \mathbf{u} - {}^t \mathbf{u}) - \left( \frac{\gamma}{\beta} - 1 \right) {}^t \dot{\mathbf{u}} - \frac{\Delta t}{2} \left( \frac{\gamma}{\beta} - 2 \right) {}^t \ddot{\mathbf{u}} \tag{3.7}$$

$${}^{t+\Delta t} \ddot{\mathbf{u}} = \frac{1}{\beta \Delta t^2} ({}^{t+\Delta t} \mathbf{u} - {}^t \mathbf{u}) - \frac{1}{\beta \Delta t} {}^t \dot{\mathbf{u}} - \left( \frac{1}{2\beta} - 1 \right) {}^t \ddot{\mathbf{u}} \tag{3.8}$$

Hierbei wird von einem linearen Verlauf der Beschleunigungen, von einem quadratischen Geschwindigkeitsverlauf und einem kubischen Verlauf der Verschiebungen ausgegangen, also entsprechend der mathematischen Integration. Die Parameter  $\beta$  und  $\gamma$  dienen analog den beiden  $\alpha$ -Werten zur Skalierung der jeweiligen Größen ( $\beta$  für die Verschiebungen,  $\gamma$  für die

Geschwindigkeit). Die Abhängigkeiten der Parameter untereinander werden in Abschnitt 3.4 (Seite 23 ff) hergeleitet und in Tabelle 3.1 (Seite 28) zusammengestellt.

Werden die Gleichungen (3.6),(3.7) und (3.8) in die Bewegungsgleichung (3.5) eingesetzt, so erhält man die *effektive Strukturgleichung*:

$$\begin{aligned}
& \mathbf{M} \left[ \frac{1 - \alpha_m}{\beta \Delta t^2} \left( {}^{t+\Delta t} \mathbf{u} - {}^t \mathbf{u} \right) + \frac{1 - \alpha_m}{\beta \Delta t} {}^t \dot{\mathbf{u}} + \left( \frac{1 - \alpha_m}{2\beta} - 1 \right) {}^t \ddot{\mathbf{u}} \right] \\
& + \mathbf{C} \left[ \left( \frac{(1 - \alpha_f) \gamma}{\beta} - 1 \right) {}^t \dot{\mathbf{u}} + (1 - \alpha_f) \frac{\Delta t}{2} \left( \frac{\gamma}{\beta} - 2 \right) {}^t \ddot{\mathbf{u}} \right] \\
& + \mathbf{F}({}^{t+\Delta t} \mathbf{u})(1 - \alpha_f) - \mathbf{F}({}^t \mathbf{u})\alpha_f \\
& = (1 - \alpha_f) {}^{t+\Delta t} \bar{\mathbf{P}} - \alpha_f {}^t \bar{\mathbf{P}}
\end{aligned} \tag{3.9}$$

Zunächst müssen die Verschiebungen  ${}^{t+\Delta t} \mathbf{u}$  ermittelt werden. Hierfür wird aus der effektiven Strukturgleichung (Gleichung 3.9) die *effektive Steifigkeitsmatrix*  $\mathbf{K}_{eff}$  ermittelt:

$$\mathbf{K}_{eff} = \mathbf{K} + \frac{1 - \alpha_m}{1 - \alpha_f} \frac{1}{\beta \Delta t^2} \mathbf{M} + \frac{\gamma}{\beta \Delta t} \mathbf{C}, \text{ mit } \alpha_f \neq 1, \beta \neq 0 \tag{3.10}$$

Die effektive Steifigkeitsmatrix enthält nun also auch Massen- und Dämpfungsanteile.

Die Steifigkeitsmatrix  $\mathbf{K}$  wird entsprechend des verwendeten Gleichgewichtsiterationsverfahrens modifiziert\*. Numerisch effizienter ist es, wenn für  $\mathbf{K}$  die lineare Steifigkeitsmatrix verwendet und entsprechend des *BFGS*-Verfahrens nur der Lastvektor verändert wird†, da hierbei nur bei einer Änderung des Zeitschrittes die effektive Steifigkeitsmatrix neu berechnet werden muß. Die physikalisch und geometrisch nichtlinearen Anteile werden dann in einem *Pseudo-Lastvektor* zusammengefaßt‡.

Voraussetzung für dieses Vorgehen ist jedoch die hier dargestellte Aufspaltung der Strukturgleichung in effektive Steifigkeitsmatrix und effektiven Lastvektor (Gleichung 3.11). Für die Implementierung einer Vorgehensweise nach *Wunderlich et al* (z.B. [WOSL87]) ist es darüberhinaus notwendig, die lineare Steifigkeitsmatrix ohne Vorfaktoren in Gleichung (3.10) einzubringen. Die implizite Darstellung nach *Kuhl* [Kuh96, KR96] ist hierfür nicht geeignet.

Der zugehörige effektive Lastvektor  $\mathbf{P}_{eff}$  wird ebenfalls aus der effektiven Strukturgleichung (Gleichung 3.9) gewonnen:

$$\begin{aligned}
{}^{\Delta t} \mathbf{P}_{eff} & = {}^{t+\Delta t} \bar{\mathbf{P}} - \frac{\alpha_f}{1 - \alpha_f} {}^t \bar{\mathbf{P}} - \frac{1 + \alpha_f}{1 - \alpha_f} {}^t \mathbf{F} \\
& + \mathbf{M} \left[ \frac{1 - \alpha_m}{\beta \Delta t (1 - \alpha_f)} {}^t \dot{\mathbf{u}} + \left( \frac{1 - \alpha_m}{2\beta (1 - \alpha_f)} - \frac{1}{1 - \alpha_f} \right) {}^t \ddot{\mathbf{u}} \right] \\
& + \mathbf{C} \left[ \left( \frac{\gamma}{\beta} - \frac{1}{1 - \alpha_f} \right) {}^t \dot{\mathbf{u}} + \frac{\Delta t}{2} \left( \frac{\gamma}{\beta} - 2 \right) {}^t \ddot{\mathbf{u}} \right]
\end{aligned} \tag{3.11}$$

\*z.B. *Newton-Raphson* Methode, siehe Abschnitt 2.3.1, Seite 11

†siehe Abschnitt 2.3.2, Seite 12

‡siehe Abschnitt 2.2.3, Seite 10

Der effektive Lastvektor wird für jeden Zeitschritt einmal berechnet, bleibt jedoch während der Gleichgewichtsiteration unverändert. Für die so gewonnenen Terme ergibt sich nun die Strukturgleichung zur Ermittlung des Gleichgewichts im aktuellen Zeitschritt:

$$\mathbf{K}_{eff} \Delta \mathbf{u} = \Delta^t \mathbf{P}_{eff} \quad (3.12)$$

oder wenn die nichtlinearen Anteile in einem Pseudo-Lastvektor erfaßt werden:

$$\mathbf{K}_{eff} \Delta \mathbf{u} = \Delta^t \mathbf{P}_{eff} - \Delta \mathbf{P}_{NL} \quad (3.13)$$

Mit Hilfe der Algorithmen zur Lösung nichtlinearer Gleichungssysteme (Abschnitt 2.3) wird schließlich die Verformungsänderung  $\Delta \mathbf{u}$  ermittelt. Der Verschiebungsvektor  $\mathbf{u}$  wird mit

$${}^{t+\Delta t} \mathbf{u} = {}^t \mathbf{u} + \Delta \mathbf{u} \quad (3.14)$$

berechnet. Die Geschwindigkeiten und Beschleunigungen werden aus den *Newmark* Ansätzen (Gleichungen 3.7 und 3.8) gewonnen.

Es wird deutlich, daß sich der numerische Aufwand der eigentlichen Zeitintegration auf wenige Matrizenmultiplikationen und -additionen beschränkt. Außerdem ist diese Methode völlig unabhängig von den verwendeten Elementformulierungen. Es können also beliebige Elemente gekoppelt werden.

### 3.4 Eigenschaften in der linearen Dynamik

Die Eigenschaften eines Zeitschrittverfahrens können für lineare Berechnungen mit Hilfe analytischer Untersuchungen dargestellt werden [BW73]. Hierzu wird der Algorithmus in eine allgemeine Form gebracht, die mechanisch als Beschreibung eines Einmassenschwingers gedeutet werden kann [HP88a]:

$$\mathbf{X}_{n+1} = \mathbf{A} \mathbf{X}_n + \mathbf{q}_{n+1} \quad (3.15)$$

mit

$$\mathbf{X} = \begin{bmatrix} \mathbf{u} \\ \Delta t \dot{\mathbf{u}} \\ \Delta t^2 \ddot{\mathbf{u}} \end{bmatrix}, \quad \mathbf{A} = \text{Vergrößerungsmatrix}, \quad \mathbf{q} = \text{Lastvektor.} \quad (3.16)$$

Die *Vergrößerungsmatrix*  $\mathbf{A}$  (*amplification matrix*) wird für das *Generalized- $\alpha$*  Verfahren durch Kombination der Arbeiten von *Hilber/Hughes* [HH78] und *Chung/Hulbert* [CH93] hergeleitet:

$$\mathbf{A} = \begin{bmatrix} 1 + \beta A_1 & 1 + \beta A_2 & \frac{1}{2} - \beta(1 - A_3) \\ \gamma A_1 & 1 + \gamma A_2 & 1 - \gamma(1 - A_3) \\ A_1 & A_2 & A_3 \end{bmatrix} \quad (3.17)$$



wobei:

$$\begin{aligned}
 A_1 &= -\frac{1}{D}\Omega^2 \\
 A_2 &= -\frac{1}{D}(2\xi\Omega + \Omega^2(1 - \alpha_f)) \\
 A_3 &= 1 - \frac{1}{D}(1 + 2\xi\Omega(1 - \alpha_f) + \frac{1}{2}\Omega^2(1 - \alpha_f)) \\
 D &= (1 - \alpha_m) + 2\xi\Omega(1 - \alpha_f)\gamma + \Omega^2(1 - \alpha_f)\beta \\
 \Omega &= \omega\Delta t \\
 \omega &= \sqrt{\frac{\mathbf{K}}{\mathbf{M}}} \text{ (Eigenfrequenz)} \\
 \xi &= \frac{\mathbf{C}}{2\sqrt{\mathbf{KM}}} \text{ (physikalische Dämpfung)}
 \end{aligned} \tag{3.18}$$

Führt man die Vergrößerungsmatrix einer Eigenwertanalyse zu, so ergeben sich aus

$$|\mathbf{A} - \lambda I| = \lambda^3 - 2A_1\lambda^2 + A_2\lambda - A_3 = 0 \tag{3.19}$$

die komplex-konjugierten Eigenwerte

$$\lambda_{1,2} = a \pm ib. \tag{3.20}$$

Aus diesen Eigenwerten läßt sich der *Spektralradius*  $\rho$

$$\rho = \sqrt{a^2 + b^2} \tag{3.21}$$

sowie die numerische Dämpfung (algorithmische Dissipation)  $\bar{\xi}$  berechnen:

$$\bar{\xi} = -\frac{1}{2 \arctan\left(\frac{b}{a}\right)} \ln(a^2 + b^2). \tag{3.22}$$

Mit diesen Herleitungen kann nun das numerische Verhalten der *Generalized- $\alpha$*  Methode für lineare dynamische Berechnungen mit anderen Verfahren verglichen werden. Dieses Vorgehen wurde bereits früher analog bei einigen der hier diskutierten Verfahren angewandt [HH78, Hug87, HP88b, HHHP89].

Es ist zu beachten, daß mit der hier dargestellten allgemeinen Herleitung andere Methoden einfach einbezogen werden können \*:

- Das *Newmark* Verfahren für  $\alpha_f = \alpha_m = 0$ .
- Das *Hilber-Hughes-Taylor- $\alpha$*  (*HHT- $\alpha$* ) Verfahren für  $\alpha_m = 0$ .
- Das *Wood-Bossak-Zienkiewicz- $\alpha$*  (*WBZ- $\alpha$* ) Verfahren für  $\alpha_f = 0$ .

---

\*siehe auch Tabelle 3.1, Seite 28



### 3.4.1 Spektrale Stabilität

Die numerische Stabilität eines Zeitschrittverfahrens ist im Linearen gewährleistet, wenn die *spektrale Stabilität* gewährleistet ist. In diesem Zusammenhang wird dann von Stabilität gesprochen, wenn die mit einem bestimmten Zeitschritt gewonnene Lösung für beliebige Anfangsbedingungen nicht über alle Grenzen wächst [Bat86]. Das Verfahren ist *unbedingt spektral stabil* wenn für den Spektralradius  $\rho$  gilt:

$$\rho \leq 1 \quad \forall \quad \Delta t \quad (3.23)$$

Hingegen spricht man von einem *bedingt spektral stabilen* Algorithmus, wenn gilt:

$$\rho \leq 1 \quad \forall \quad \Delta t < \Delta t_{krit} \quad (3.24)$$

Die maximale Zeitschrittlänge  $\Delta t_{krit}$  kann aus Gleichung (3.21) unter Berücksichtigung von Gleichung (3.24) gewonnen werden, wenn man die höchste Eigenfrequenz  $\omega$  kennt.

Nach [CH93] ist das *Generalized- $\alpha$*  Verfahren bei

$$\beta \geq \frac{1}{4}(1 - \alpha_m + \alpha_f)^2 \quad \text{und} \quad \alpha_m \leq \alpha_f \leq \frac{1}{2} \quad (3.25)$$

*unbedingt spektral stabil*. Für die anderen Verfahren der *Newmark-Familie* gilt dies bei:

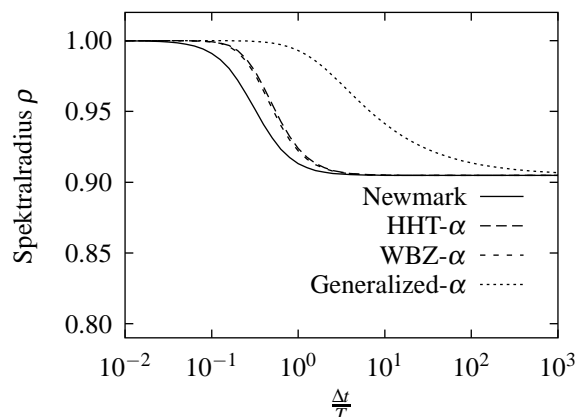
$$\beta \geq \frac{1}{4} \left( \gamma + \frac{1}{2} \right)^2 \quad (3.26)$$

Es sollte vermieden werden, einen bedingt spektral stabilen Algorithmus zu verwenden, wenn a priori nicht bekannt ist, welche maximalen Frequenzen bei der Berechnung auftreten und wie groß deshalb  $\Delta t_{krit}$  ist. Wird ein Zeitschritt verwendet, der Gleichung (3.24) nicht erfüllt, können die nachfolgenden Ergebnisse über alle Grenzen wachsen – also völlig falsche Werte liefern.

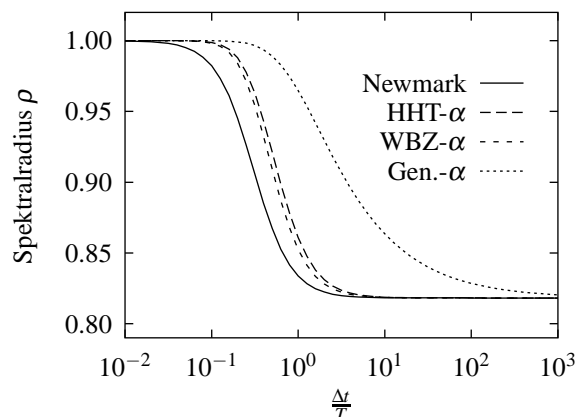
Je nach Wahl der Parameter verändert sich der Spektralradius für die verschiedenen Verfahren. Während bei der Trapezregel\* für alle Zeitschrittlängen der Spektralradius konstant bei 1.0 liegt, ist er bei den anderen *Newmark* Verfahren für hohe Frequenzen ( $\frac{\Delta t}{T}$  groß) kleiner als 1.0 (Abbildungen 3.2 und 3.3). Je größer der Parameter  $\gamma$  gewählt wird, desto kleiner wird der Spektralradius  $\rho$ . Wie in Abschnitt 3.4.3 (Seite 27) noch gezeigt wird, erhöht sich mit abnehmendem  $\rho$  die numerische Dissipation  $\xi$ .

Anhand der Abbildungen 3.2 und 3.3 sind zwischen den gezeigten Verfahren deutliche Unterschiede zu erkennen: Während bei *Newmark* und *WBZ* bzw. *HHT* Verfahren der Spektralradius bereits bei  $\frac{\Delta t}{T} \approx 10^{-1}$  stark abfällt, ist dies beim *Generalized  $\alpha$*  Verfahren erst bei  $\frac{\Delta t}{T} \approx 10^0$  der Fall. Zwischen der *WBZ* und der *HHT* Methode ist kaum ein Unterschied feststellbar – die Kurven sind nahezu deckungsgleich.

\**Newmark* Verfahren mit  $\beta = 0.25, \gamma = 0.5$



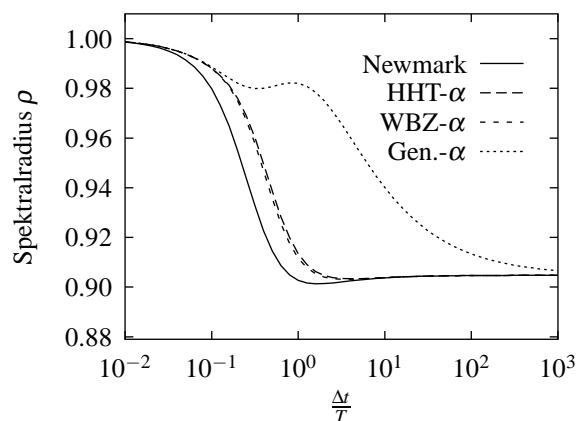
**Abb. 3.2:** Spektralradien verschiedener Verfahren für  $\gamma = 0.55$



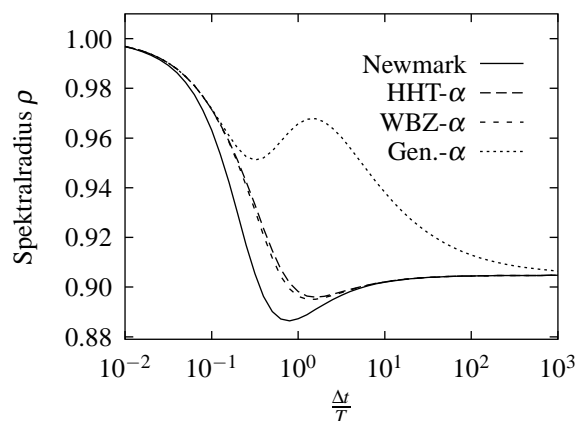
**Abb. 3.3:** Spektralradien verschiedener Verfahren für  $\gamma = 0.6$

Eine Änderung des Parameters  $\gamma$  (und der jeweils zugeordneten Werte für  $\alpha$  und  $\beta$ ) ergibt zwar einen veränderten minimalen Spektralradius ( $\rho \approx 0.9$  für  $\gamma = 0.55$  und  $\rho \approx 0.82$  für  $\gamma = 0.6$ ), die grundsätzliche Charakteristik der Funktionsverläufe bleibt jedoch gleich. Die wesentlichen Merkmale der hier untersuchten Algorithmen sind also unabhängig von der gewählten Parametrisierung.

Physikalische Dämpfung wird bei linearen Berechnungen mit der modalen Dämpfungsmatrix  $C$  berücksichtigt (siehe Gleichung 3.3, Seite 16). Dies wirkt sich auch auf den Spektralradius



**Abb. 3.4:** Spektralradien verschiedener Verfahren für  $\gamma = 0.55$  und 2 % physikalische Dämpfung



**Abb. 3.5:** Spektralradien verschiedener Verfahren für  $\gamma = 0.55$  und 5 % physikalische Dämpfung

aus. Auffällig ist hierbei, daß sich das *Generalized-α* Verfahren von den anderen Algorithmen wesentlich unterscheidet: Der Funktions-Verlauf hält sich bis  $\frac{\Delta t}{T} \approx 10^1$  auf einem relativ hohen Niveau und nähert sich erst ab  $\frac{\Delta t}{T} \approx 10^2$  dem Wert der anderen Verfahren (Abbildung 3.4). Bei hoher physikalischer Dämpfung ( $\xi \approx 5\%$ ) ist ein weiterer Effekt bei allen Verfahren erkennbar: Die Funktionsverläufe fallen nicht mehr monoton, sondern schwanken im mittleren Bereich ( $\frac{\Delta t}{T} \approx 10^0$ ) ausgeprägt (Abbildung 3.5).

### 3.4.2 Genauigkeit

Die Genauigkeit eines unbedingt spektral stabilen Einschritt-Zeitintegrationsverfahrens kann nach [HH78] maximal von 2. Ordnung sein. In [HP88b] und [CH93] wird eine Formel zur Berechnung des lokalen Integrationsfehlers  $\tau$  zum Zeitpunkt  $t_n$  angegeben:

$$\tau_n = \frac{1}{\Delta t^2} \sum_{i=0}^3 (-1)^i A_i \mathbf{u}(t_{n+1-i}) \quad (3.27)$$

Hieraus lassen sich die Parameter der Integrationsverfahren so berechnen, daß die gewünschte Genauigkeit 2. Ordnung erreicht wird:

$$\gamma = \frac{1}{2} - \alpha_m + \alpha_f \quad (3.28)$$

für die  $\alpha$ -Verfahren und

$$\gamma = \frac{1}{2} \quad (3.29)$$

für das *Newmark* Verfahren ( $\alpha_m = \alpha_f = 0$ ). Davon abweichende Parametrisierungen führen lediglich zu einer Genauigkeit 1. Ordnung. Das *Newmark* Verfahren ist also nur von 1. Ordnung genau, außer wenn die Trapezregel ( $\gamma = \frac{1}{2}$ ) verwendet wird.

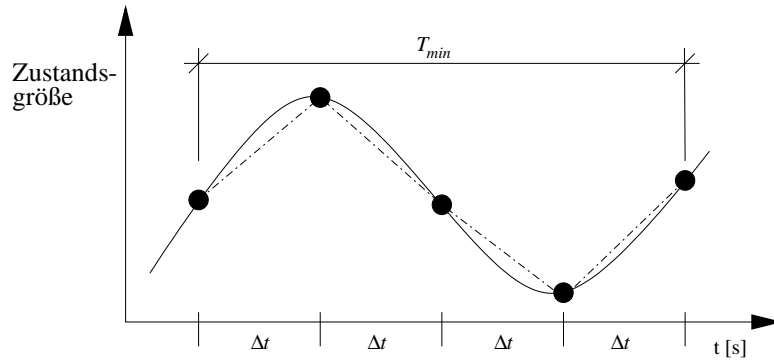
### 3.4.3 Numerische Dissipation

Die *numerische Dissipation* (auch als *algorithmische Dissipation* bezeichnet) ist eine besondere Eigenschaft mancher Zeitschrittverfahren, die zu einer Dämpfung der Strukturantwort führt, wobei insbesondere hohe Frequenzen beeinflusst werden. Diese Dämpfung ist rein numerisch bedingt und basiert nicht auf mechanischen Eigenschaften. Der Effekt ist also zu nächst unerwünscht, da hierdurch das zu ermittelnde Ergebnis verfälscht wird.

Frühere Untersuchungen zeigen, daß es Verfahren gibt, die keine numerische Dämpfung aufweisen (z.B. Trapezregel) und Verfahren, die zu einer sehr starken Dämpfung führen (z.B. *Houbolt* Methode, *Wilson- $\Theta$*  Verfahren) [HH78].

Obwohl das Simulationsergebnis durch die numerische Dämpfung beeinflusst wird, kann sie von großem Nutzen sein:

- Bei vielen Berechnungen können hochfrequente Oszillationen auftreten. Ursache können Störungen aus der numerischen Berechnung sein, z.B. infolge von Rundungsfehlern oder Kontaktbedingungen. Damit diese Oszillationen nicht den gesamten Rechengang stören, können sie sehr effektiv mit Hilfe der numerischen Dämpfung eliminiert werden [BH83, HHT77, Hug87, Hul94, Wri81].
- Geometrisch nichtlineare Berechnungen können durch die numerische Dämpfung stabilisiert werden (siehe hierzu die ausführliche Diskussion in Abschnitt 3.5, Seite 36).



**Abb. 3.6:** Höchste, näherungsweise abbildbare Frequenz für  $\Delta t = const$

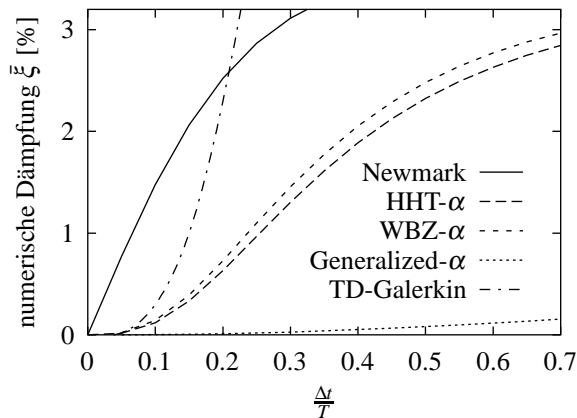
Zur Bestimmung der höchsten Frequenz (bzw. der minimalen Periode  $T_{min}$ ), die mit einem vorgegebenen Zeitschritt  $\Delta t = const$  gerade noch abgebildet wird, kann Abbildung 3.6 dienen. Offensichtlich benötigt man zur numerischen Diskretisierung einer annähernd sinusförmigen Schwingung mindestens vier Zeitschritte je Periodenlänge. Bei einer vorgegebenen Zeitschrittlänge bedeutet dies, daß kürzere Perioden als  $\frac{\Delta t}{T} \approx \frac{1}{4}$  nicht mehr berücksichtigt werden können.

Bezüglich der numerischen Dämpfung läßt sich damit folgender Schluß ziehen: Längere Perioden als  $\frac{\Delta t}{T} \approx \frac{1}{4}$  sollen möglichst nicht beeinflußt werden, kürzere Perioden können – falls notwendig – numerisch dissipiert werden, da sie ohnehin nicht mit der aktuellen Zeitschrittlänge abgebildet werden können. Aus diesen Bedingungen (möglichst geringe Dämpfung niedriger Frequenzen, starke Dämpfung hoher Frequenzen) lassen sich für die drei  $\alpha$ -Methoden eindeutige Zusammenhänge zwischen den Steuerparametern  $\alpha_f, \alpha_m, \beta$  und  $\gamma$  sowie dem Spektralradius  $\rho_\infty$  (für  $\frac{\Delta t}{T} \rightarrow \infty$ ) herleiten (Tabelle 3.1).

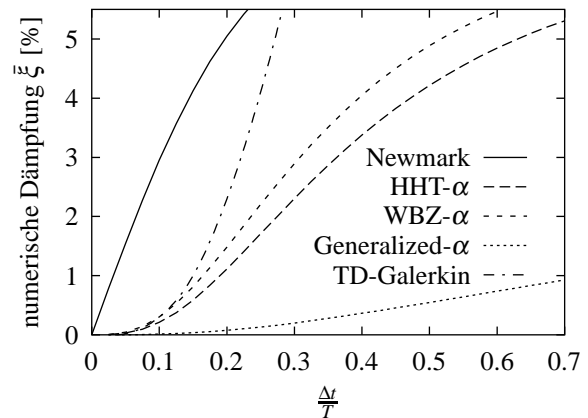
	Generalized- $\alpha$	HHT- $\alpha$	WBZ- $\alpha$
Zusammenhang von $\rho_\infty$ und $\alpha$ -Werten sowie minimale Dämpfung niedriger Frequenzen	$\alpha_f = \frac{\rho_\infty}{\rho_\infty + 1}$ $\alpha_m = \frac{2\rho_\infty - 1}{\rho_\infty + 1}$	$\alpha_f = \frac{1 - \rho_\infty}{\rho_\infty + 1}$ $\alpha_m = 0$	$\alpha_f = 0$ $\alpha_m = \frac{\rho_\infty - 1}{\rho_\infty + 1}$
maximale Dämpfung hoher Frequenzen	$\beta = \frac{1}{4}(1 - \alpha_m + \alpha_f)^2$	$\beta = \frac{1}{4}(1 + \alpha_f)^2$	$\beta = \frac{1}{4}(1 - \alpha_m)^2$
Genauigkeit 2. Ordnung	$\gamma = \frac{1}{2} - \alpha_m + \alpha_f$	$\gamma = \frac{1}{2} + \alpha_f$	$\gamma = \frac{1}{2} - \alpha_m$

**Tabelle 3.1:** Parameterwahl der  $\alpha$ -Verfahren

Wie bereits in Abschnitt 3.4, Gleichung (3.22) gezeigt wurde, läßt sich die algorithmische Dämpfung für lineare Systeme analytisch berechnen. Die Bilder 3.7 und 3.8 zeigen die Auswertung für verschiedene Verfahren. Hier ist – wie in Abschnitt 3.4.1 bereits erwähnt – auch der Zusammenhang zwischen Spektralradius  $\rho$  und numerischer Dämpfung  $\xi$  erkennbar: Je kleiner der Spektralradius, desto größer ist die algorithmische Dämpfung. Darüberhinaus wird auch hier wieder ein grundlegender Unterschied zwischen den Verfahren deutlich: Während das *Generalized- $\alpha$*  Verfahren höhere Moden erst ab ca.  $\frac{\Delta t}{T} \approx 0.3$  spürbar dissipiert, geschieht dies bei den anderen Verfahren – insbesondere beim *Newmark* Verfahren – schon



**Abb. 3.7:** Numerische Dämpfung verschiedener Verfahren für  $\gamma = 0.55$



**Abb. 3.8:** Numerische Dämpfung verschiedener Verfahren für  $\gamma = 0.6$

bei relativ niedrigen Frequenzen und zudem sehr stark. Auch das in Abschnitt 3.2.6 (Seite 19) erläuterte *Time-Discontinuous-Galerkin* Verfahren führt zu einer sehr starken Dissipation von Frequenzen ab  $\frac{\Delta t}{T} \approx 0.15$ . Zudem kann hierbei die numerische Dissipation nicht über Parameter gesteuert werden.

Die mit Hilfe von Abbildung 3.6 definierte Grenze für den unbeeinflussten Frequenzbereich wird somit nur vom *Generalized- $\alpha$*  Verfahren befriedigend erreicht. Die anderen Methoden dämpfen auch diejenigen Frequenzen, die von der aktuellen Zeitschrittlänge noch gut abgebildet werden können, was zu einem ungenauen Ergebnis der numerischen Simulation bei relativ hohen Frequenzen führt.

Mit dem *Generalized- $\alpha$*  Verfahren kann deshalb bei gleicher Genauigkeit eine größere Schrittweite gewählt werden, wodurch der numerischen Aufwand merklich verringert wird.

### Physikalische Dämpfung

Wie bereits in Kapitel 3.4.1 (Seite 25) gezeigt wurde, hat die physikalische Dissipation einen großen Einfluß auf die numerischen Eigenschaften der Zeitintegration (Abbildungen 3.4 und 3.5). Hier soll nun die Kombination aus numerischer und physikalischer Dämpfung untersucht werden.

Analog dem physikalisch ungedämpften Fall ergeben sich die Funktionen für physikalische Dissipation (Abbildungen 3.9–3.14). Da es wünschenswert ist, daß alle Frequenzen bei der numerischen Simulation gleich stark gedämpft werden, erhält man je nach Maß der physikalischen Dämpfung unterschiedliche optimale Parameter (Tabelle 3.2). Allerdings bestätigen sich für alle Parameter wieder die Ergebnisse aus dem vorhergehenden Abschnitt: Das *Newmark* Verfahren zeigt im Vergleich zu den  $\alpha$ -Verfahren deutlich schlechtere numerische Eigenschaften. Neben den hohen Frequenzen werden auch die tiefen Frequenzen ( $\frac{\Delta t}{T} \leq 0.1$ ) stark gedämpft. Selbst für sehr kleine  $\gamma$ -Werte ist die numerische Dämpfung deutlich höher als die physikalische Dämpfung.

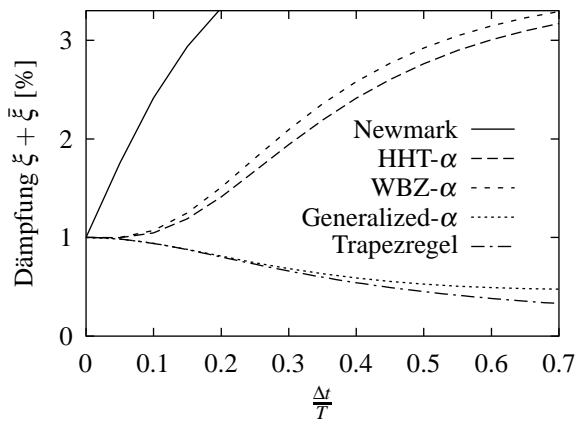


Abb. 3.9: Numerische und phys. Dämpfung ( $\gamma = 0.55$  und  $\xi = 0.01$ )

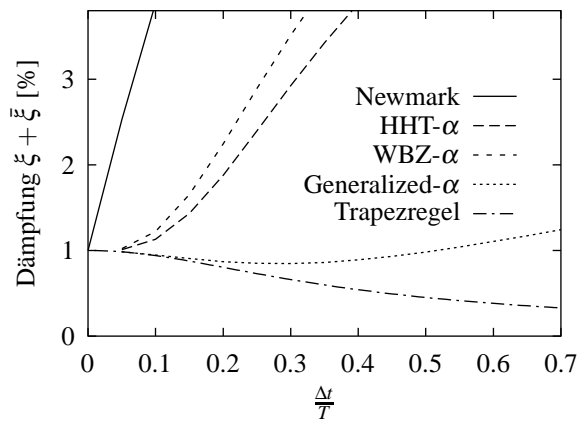


Abb. 3.10: Numerische und phys. Dämpfung ( $\gamma = 0.6$  und  $\xi = 0.01$ )

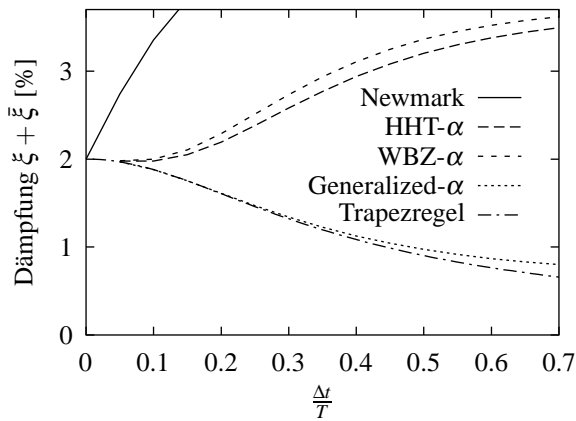


Abb. 3.11: Numerische und phys. Dämpfung ( $\gamma = 0.55$  und  $\xi = 0.02$ )

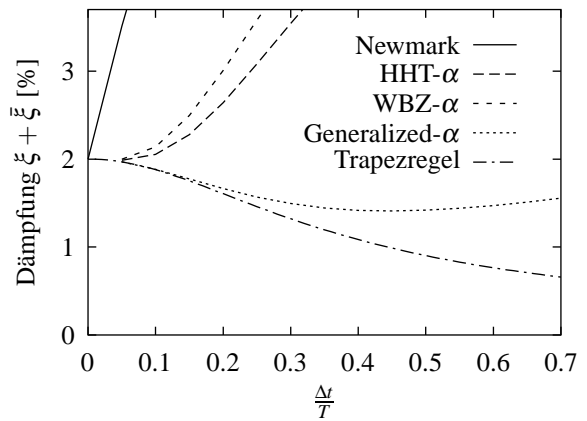


Abb. 3.12: Numerische und phys. Dämpfung ( $\gamma = 0.6$  und  $\xi = 0.02$ )

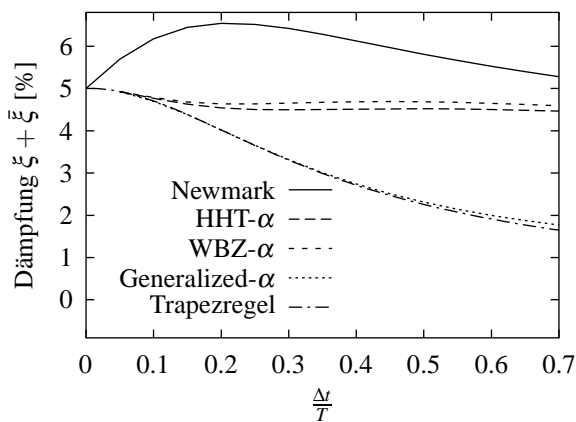


Abb. 3.13: Numerische und phys. Dämpfung ( $\gamma = 0.55$  und  $\xi = 0.05$ )

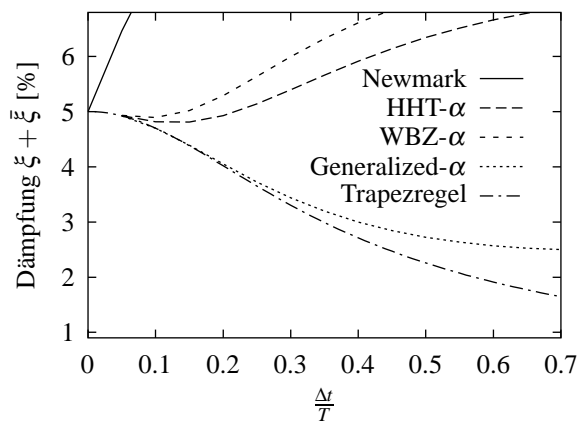


Abb. 3.14: Numerische und phys. Dämpfung ( $\gamma = 0.6$  und  $\xi = 0.05$ )

Dämpfung $\xi$	HHT/WBZ- $\alpha$	Generalized- $\alpha$
$\xi = 1\%$	$\gamma \approx 0.51$	$\gamma \approx 0.6$
$\xi = 2\%$	$\gamma \approx 0.52$	$\gamma \approx 0.65$
$\xi = 3\%$	$\gamma \approx 0.53$	$\gamma \approx 0.7$
$\xi = 5\%$	$\gamma \approx 0.55$	$\gamma \approx 0.8$

**Tabelle 3.2:** Optimale Werte für Parameter  $\gamma$  bei physikalischer Dämpfung

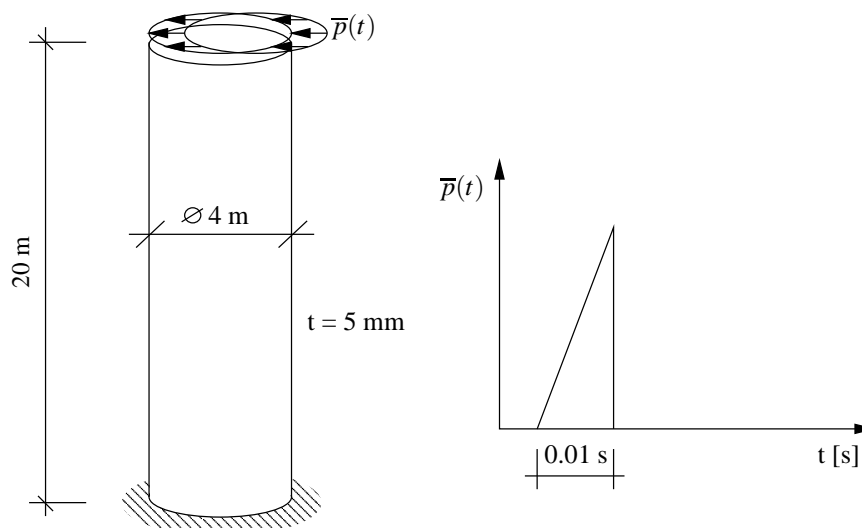
Die Trapezregel ( $\equiv$  ungedämpftes *Newmark* Verfahren mit  $\gamma = 0.5$ ) zeigt bei höheren Frequenzen eine merklich geminderte Dissipation. Die *Newmark* Methode ist daher für alle Parametrisierungen ungeeignet, physikalische Dissipation zu unterstützen.

Lediglich bei einer starken physikalischen Dämpfung ( $\xi = 0.05$ ) bieten das HHT- $\alpha$  sowie das WBZ- $\alpha$  Verfahren für die Parametrisierung mit  $\gamma = 0.55$  eine optimale, gleichmäßige Dämpfung (Abbildung 3.13). Dies kann auch mit dem *Generalized- $\alpha$*  Verfahren erreicht werden, wenn eine entsprechend hohe numerische Dämpfung (Parameter  $\gamma \approx 0.8$ ) verwendet wird.

Das *Generalized- $\alpha$*  Verfahren stellt also in dem hier untersuchten Kontext die flexibelste und damit am besten geeignete Methode dar.

### Numerische Berechnung: Schwingendes System (lineare Dynamik)

Zur Überprüfung der analytischen Untersuchungen wird ein unten eingespanntes, zylinderförmiges Stahlrohr mit einer dreiecksförmigen Impulslast am obersten Rand belastet (Abbildung 3.15). Das Rohr schwingt anschließend physikalisch ungedämpft in mehreren Eigenfrequenzen.



**Abb. 3.15:** Zylinderförmiges Rohr unter Impulsbelastung

Die räumliche Diskretisierung erfolgt über rotationssymmetrische Ringelemente nach *Wunderlich et al* [WCO85] (siehe hierzu auch die Beschreibung in Abschnitt 2.2.3, Seite 10).

Alle Berechnungen werden mit konstantem Zeitschritt ( $\Delta t = 0.001s$ ) durchgeführt. Die relativ kurze Schrittweite gewährleistet auch die Abbildung sehr hoher Eigenfrequenzen: Bis etwa 150 Hz können die Zustandsgrößen sehr genau erfasst werden, während bis etwa 250 Hz eine grobe Approximation möglich ist. Höhere Frequenzen sind weder mit der räumlichen Diskretisierung\* noch mit der gewählten Zeitschrittlänge zu erfassen.

Aus einer Eigenwertanalyse ergeben sich folgende Eigenfrequenzen:

$$f_1 = 7.39 \frac{1}{s}, f_2 = 41.06 \frac{1}{s}, f_3 = 96.08 \frac{1}{s}, f_4 = 153.6 \frac{1}{s}.$$

Es wird eine geringe numerische Dämpfung angenommen (Parameter  $\gamma = 0.55$ ). Für die numerische Simulation wird geometrisch und physikalisch lineares Verhalten angenommen, um Nebeneffekte zu vermeiden. Zur Beurteilung der Ergebnisse wird jeweils die Längskraft in Meridianrichtung an der Einspannung sowie die numerisch dissipierte Energie  $E_{num}$  verwendet. Diese läßt sich über

$$E_{num} = E_{gesamt} - E_{kin} - E_{pot} \quad (3.30)$$

berechnen.

Der Verlauf der Meridiankraft aus einer numerisch ungedämpften Berechnung (Trapezregel) und der Verlauf aus dem *Generalized- $\alpha$*  Verfahren ist nahezu deckungsgleich. Dagegen zeigt

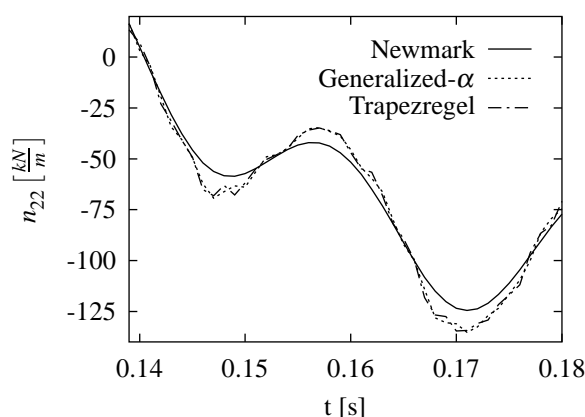


Abb. 3.16: Meridiankraft am Rohrfuß

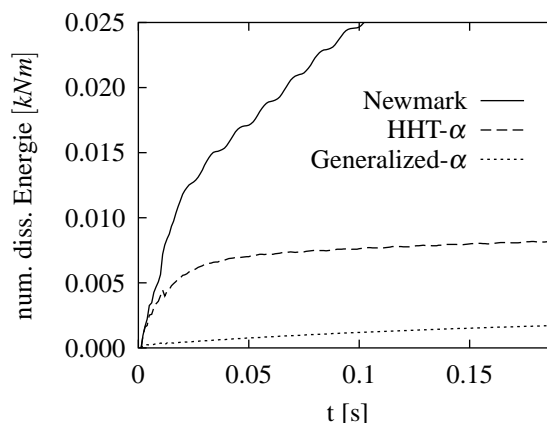


Abb. 3.17: Numerisch dissipierte Energie

sich der Verlauf aus dem *Newmark* Verfahren stark geglättet (Abbildung 3.16). Offensichtlich sind besonders die hohen Frequenzen gedämpft worden.

Die Zeitverläufe der numerisch dissipierten Energie zeigen, daß das *Newmark* Verfahren im Vergleich zu den anderen Algorithmen sehr stark numerisch dämpft (Abbildung 3.17). Aber auch das *Hilber-Hughes-Taylor-alpha* Verfahren liegt weit über dem *Generalized-alpha* Verfahren. Die Trapezregel ist hier nicht gezeigt, da sie keine numerische Dissipation aufweist.

Aus der Spektralanalyse des Meridiankraftverlaufs (Abbildung 3.18) lassen sich diese Erkenntnisse genauer quantifizieren. Das *Newmark* Verfahren zeigt bereits ab einer Frequenz von etwa 50 Hz ( $\frac{\Delta t}{T} = 0.05$ ) deutliche geringere Amplituden im Frequenzband. Etwas besser

\*Elementlänge im Verhältnis zur Wellengeschwindigkeit



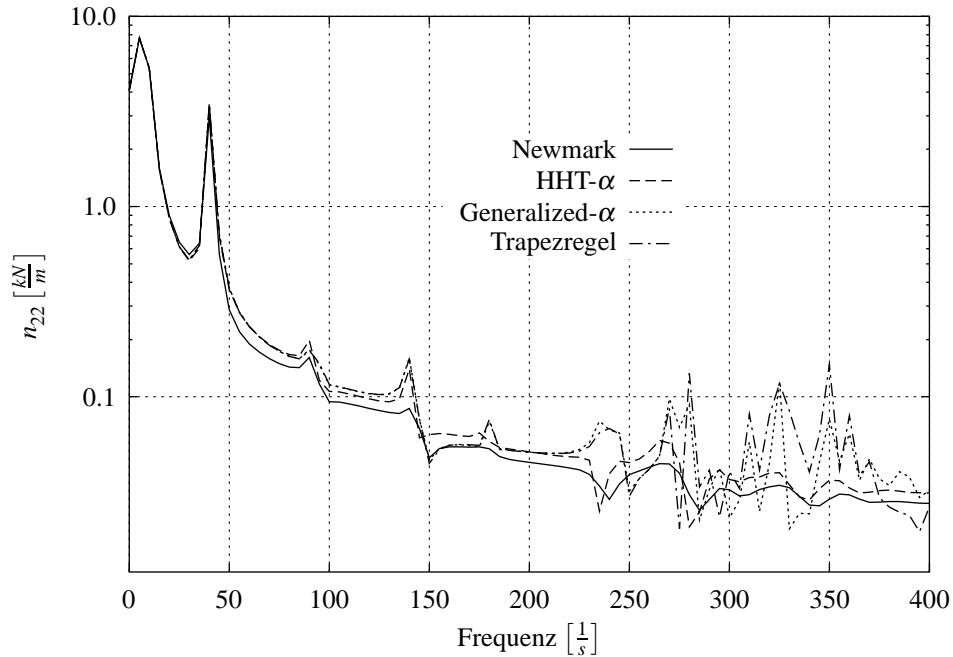


Abb. 3.18: Amplitudenspektrum des Meridiankraftverlaufs

wiederum das *HHT- $\alpha$*  Verfahren: Erst ab etwa 100 Hz ( $\frac{\Delta t}{T} = 0.1$ ) ist eine Glättung des Frequenzbandes erkennbar. Es werden also mit beiden Methoden Frequenzen beeinflusst, die mit der gewählten Zeitschrittlänge noch sehr gut approximiert werden können.

Das *Generalized- $\alpha$*  Verfahren ist wiederum nahezu deckungsgleich mit der Trapezregel: Auch bei sehr hohen Frequenzen tritt kaum eine Veränderung des Spektrums auf. Die Frequenzen ab etwa 250 Hz ( $\frac{\Delta t}{T} \geq \frac{1}{4}$ ) sind mit der gewählten Schrittweite nicht mehr zu erfassen, weshalb die unterschiedlichen Amplituden in diesem Bereich unerheblich sind.

Diese numerischen Berechnungen bestätigen die oben analytisch durchgeführten Untersuchungen: Das *Generalized- $\alpha$*  Verfahren ist den anderen dissipativen Algorithmen in Bezug auf die numerische Dämpfung weit überlegen. Es lassen sich trotz der numerischen Dämpfung hochfrequente Moden noch gut abbilden. Es werden nur solche Frequenzen dissipiert, die mit der jeweiligen Zeitschrittlänge ohnehin nicht erfaßt werden können.

### Numerische Berechnung: Wellenausbreitung

Beim nun folgenden Beispiel wird der Einfluß der gewählten Zeitintegrationsmethode auf die Wellenausbreitung in einem elastischen Körper untersucht. Hierzu wird ein elastischer Stab mit einer kurzzeitig wirkenden, sinusförmigen Belastung verwendet (Abbildung 3.19). Die räumliche Diskretisierung erfolgt über 80 Finite Elemente mit linearem Verschiebungsansatz.

Für eine ähnliche Problemstellung hat *Wriggers* in [Wri81] eine optimale Zeitschrittlänge ( $\Delta t_{opt}$ ) für das *Newmark* Verfahren empirisch ermittelt:

$$\Delta t_{opt} \approx 0.7 \frac{\Delta x}{c}, \text{ mit } c = \sqrt{\frac{E}{\rho}}. \quad (3.31)$$

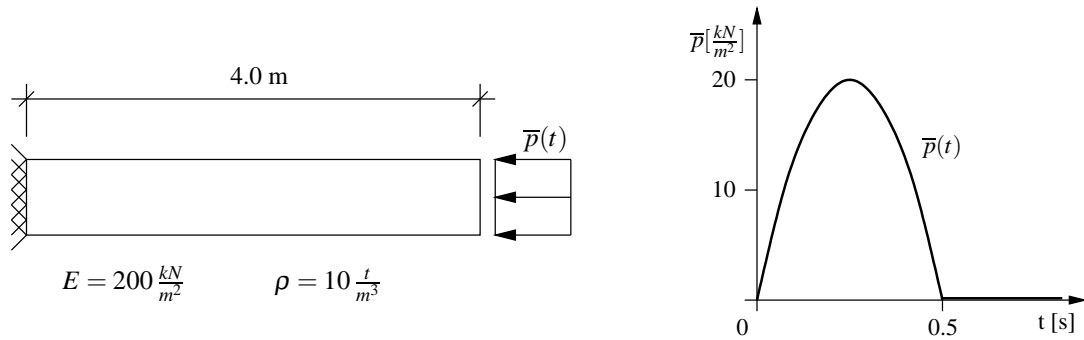


Abb. 3.19: Elastischer Stab, belastet mit einer Sinushalbwelle

Der Wert  $\Delta x$  gibt hierbei die Elementlänge und  $c$  die Wellengeschwindigkeit im elastischen Körper an. Zunächst wird hier die Formel überprüft und getestet, ob auch für das *Generalized- $\alpha$*  Verfahren diese optimale Schrittweite gilt.

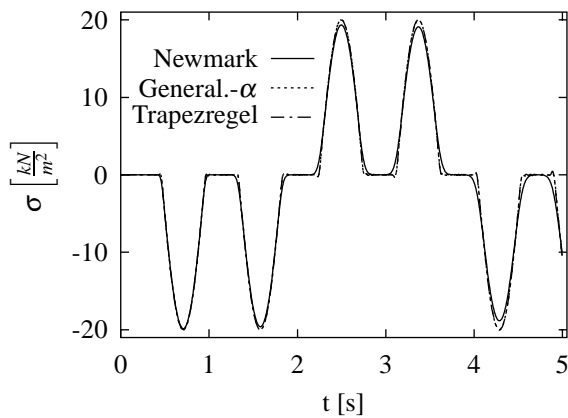


Abb. 3.20: Zeitverlauf der Spannung in der Stabmitte mit optimaler Zeitschrittweite ( $\Delta t_{opt} = 0.0075s$ )

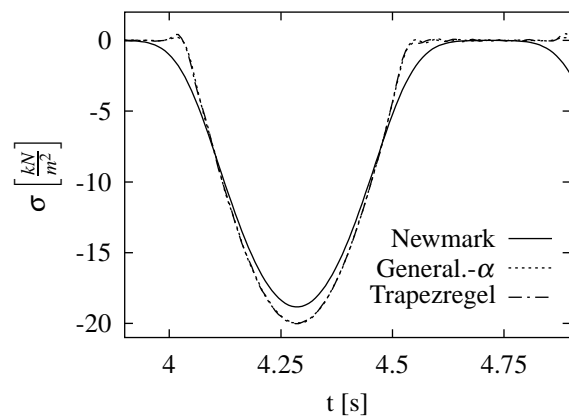


Abb. 3.21: Detail aus Abbildung 3.20 ( $\gamma = 0.6$  für Newmark und Generalized- $\alpha$ )

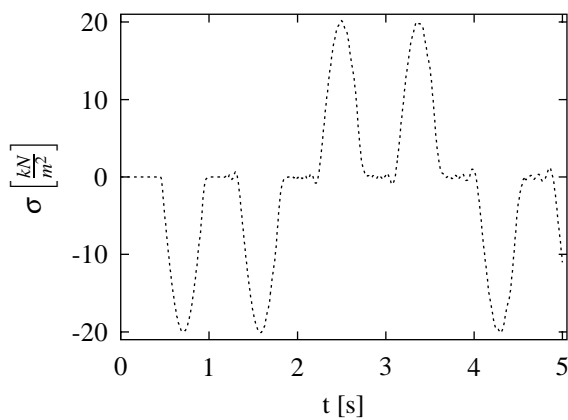


Abb. 3.22: Zeitverlauf der Spannung mit  $\Delta t = \frac{\Delta t_{opt}}{2} = 0.0038s$  (Generalized- $\alpha$  mit  $\gamma = 0.6$ )

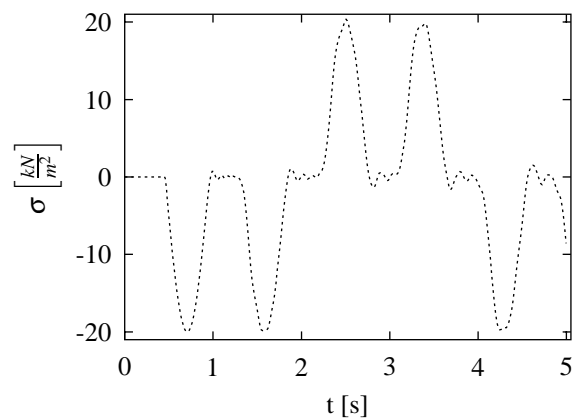
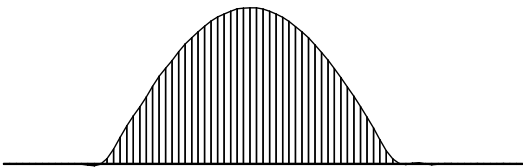


Abb. 3.23: Zeitverlauf der Spannung mit  $\Delta t = 2\Delta t_{opt} = 0.015s$  (Generalized- $\alpha$  mit  $\gamma = 0.6$ )

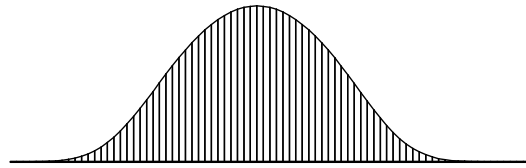
Die Ergebnisse der numerischen Simulation zeigen deutlich, daß man mit der optimalen Zeitschrittlänge  $\Delta t_{opt}$  nach Gleichung (3.31) sowohl mit der Trapezregel als auch dem *Generalized- $\alpha$*  Verfahren sehr gute Ergebnisse erzielt. Auch nach einigen Reflexionen wird die Welle nahezu ungestört abgebildet (Abbildung 3.20 und 3.21).

Eine kürzere bzw. längere Schrittweite führt zu den von *Wriggers* beschriebenen Effekten: Bei einer zu kurzen Schrittweite (Abbildung 3.22) treten am Ende der Welle Oszillationen auf, bei einer zu großen Schrittweite (Abbildung 3.23) erhält man diese Oszillationen am Beginn der Wellenfront (Hier ist nur das *Generalized- $\alpha$*  Verfahren dargestellt). Leider reagieren die Berechnungsergebnisse sehr sensitiv auf die Zeitschrittlänge. Schon eine Änderung um 10 % kann zu einer merklichen Oszillation führen. Mit der *Generalized- $\alpha$*  Methode lassen sich im Vergleich zur Trapezregel geringfügig bessere Ergebnisse erzielen, da die störenden Oszillationen etwas gedämpft werden.

Wie zu erwarten ist, dämpft auch hier das *Newmark* Verfahren die Strukturantwort zu stark. Bei dem gezeigten Beispiel wird dadurch nicht nur die maximale Amplitude allmählich geringer, sondern es geht vor allem die ursprüngliche Form der Welle (Sinushalbwelle) verloren (Abbildungen 3.21 und 3.25). Die unerwünschten Oszillationen an der Wellenfront und am Wellenende können mit dem *Newmark* Verfahren etwas korrigiert werden (hier nicht dargestellt).



**Abb. 3.24:** Räumliche Verteilung der Welle zum Zeitpunkt  $t = 4.3 \text{ s}$  mit dem *Generalized- $\alpha$*  Verfahren



**Abb. 3.25:** Räumliche Verteilung der Welle zum Zeitpunkt  $t = 4.3 \text{ s}$  mit dem *Newmark* Verfahren

Unter Berücksichtigung der optimalen Schrittlänge  $\Delta t_{opt}$  nach Gleichung 3.31 bietet das *Generalized- $\alpha$*  Verfahren also auch bei der numerischen Simulation einer Wellenausbreitung einen guten Kompromiß zwischen Genauigkeit und der Elimination unerwünschter Nebeneffekte.

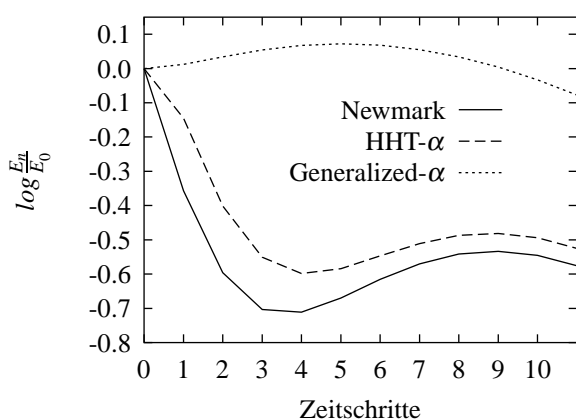
Die Berechnung dieses Beispiels mit dem *Time-Discontinuous-Galerkin* Verfahren führt nach [NS99] zu ähnlichen Ergebnissen wie mit dem *Newmark* Verfahren. Auch hier wird die Form der Welle infolge der zu starken numerischen Dissipation verfälscht, wobei insbesondere die "Ecken" ausgerundet werden.

Manche Autoren verwenden bei Beispielrechnungen Schrittweiten, die teilweise stark von Gleichung (3.31) abweichen und erhalten dadurch sehr ungenaue Ergebnisse [Li96, NS99, Wib99]. Nach Meinung des Autors kann hieraus jedoch nicht gefolgert werden, daß die *Newmark-Familie* für Wellenausbreitungsphänomene generell ungeeignet ist.

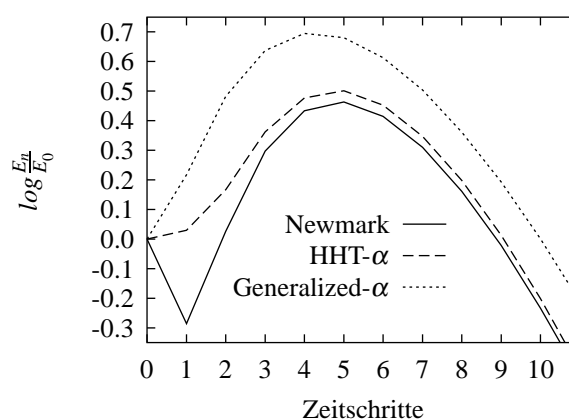
### 3.4.4 Overshooting

Eine weitere Eigenschaft von Zeitschrittverfahren ist das *Overshooting*-Verhalten. Unter *Overshooting* versteht man ein "überschiessen" einer Kenngröße in den ersten Schritten einer Berechnung bei bestimmten Ausgangsbedingungen. Ausführlich untersucht wurde dieses Phänomen in [HH78] für die Verfahren nach *Park*, *Houbolt*, *Wilson*, *Newmark* und *Hilber-Hughes-Taylor*.

Im Vergleich zu den anderen genannten Methoden spielt das *Overshooting* bei den hier untersuchten Verfahren eine untergeordnete Rolle, zumal es nur bei sehr hochfrequenten Moden auftritt. Da es für das *Generalized- $\alpha$*  Verfahren in der Literatur keine genauen Angaben hierüber gibt, seien hier einige Ergebnisse dargestellt.



**Abb. 3.26:** Overshooting für  $\frac{\Delta t}{T} = 10$  und  $\gamma = 0.55$



**Abb. 3.27:** Overshooting für  $\frac{\Delta t}{T} = 10$  und  $\gamma = 0.6$

Die Untersuchung geht von einem Einmassenschwinger mit den Anfangsbedingungen  $\mathbf{u}_0 = 1$ ,  $\dot{\mathbf{u}}_0 = 0$ ,  $\ddot{\mathbf{u}}_0 = -1$  aus. Als Kenngröße wird die Summe aus kinetischer und potentieller Energie des aktuellen Zeitschrittes ( $E_n$ ) im Verhältnis zur Summe aus kinetischer und potentieller Energie zu Beginn der Berechnung ( $E_0$ ) verwendet.

Aus den Abbildungen 3.26 und 3.27 wird deutlich, daß das Verhalten der hier untersuchten Verfahren sehr sensitiv auf eine Veränderung des Parameters  $\gamma$  reagiert. Dieser Effekt ist nur bei relativ hohen Frequenzen (ab ca.  $\frac{\Delta t}{T} > 5$ ) erkennbar. Bei numerischen Berechnungen mit den im Rahmen dieser Arbeit verwendeten Methoden kann deshalb das *Overshooting* vernachlässigt werden.

## 3.5 Eigenschaften in der nichtlinearen Strukturdynamik

Die bisherigen Betrachtungen gelten nur für den geometrisch und physikalisch linearen Fall. Während physikalisch schwach nichtlineare Effekte über globale Dämpfungsterme (Dämpfungsmatrix  $\mathbf{C}$ ) erfaßt werden können, ist dies bei ausgeprägt plastischem und geometrisch nichtlinearem Verhalten unmöglich.

Wie bereits in Abschnitt 3.2.2 erläutert, muß zur korrekten Erfassung aller nichtlinearen Effekte eine direkte Berechnungsmethode im Zeitbereich angewendet werden. Da die Betrachtungen und Ergebnisse zum linearen Verhalten (Abschnitt 3.4) nicht ohne genaue Überprüfung auf nichtlineare Systeme übertragen werden können, scheint eine eingehende Untersuchung angezeigt.

Eigenschaften wie numerische Stabilität, Dissipation oder Genauigkeit können im nichtlinearen Fall nicht stringent mathematisch abgeleitet werden. Es ist deshalb notwendig, daß für jede zu untersuchende Finite-Element-Formulierung ein aussagefähiger Test unternommen wird. Bisher gibt es allerdings keinen universellen *Benchmark*, der zuverlässige Aussagen zu den einzelnen Punkten liefern würde. Auch sind sowohl die räumlichen als auch die zeitlichen Diskretisierungen im weiten Feld der nichtlinearen Strukturdynamik zu unterschiedlich, um in ein starres Schema zu passen.

Hier werden zunächst die besonderen Probleme bei der nichtlinearen dynamischen Berechnung von Stoß- und Kontaktproblemen beleuchtet, wie sie bei gekoppelten Strukturen auftreten können.

Anschließend wird exemplarisch ein komplexes System untersucht, das mit den in Abschnitt 6.2 beschriebenen Elementen diskretisiert ist. Im Mittelpunkt steht dabei die Frage, ob mit Hilfe der im Rahmen dieser Arbeit eingehend untersuchten Zeitintegrationsverfahren eine numerisch robuste Simulation durchgeführt werden kann.

### 3.5.1 Stoß- und Kontaktsimulation

Bei der Stoß- und Kontaktsimulation steht weniger die numerische Stabilität des Zeitintegrationsverfahrens im Mittelpunkt. Hauptproblem ist hierbei, daß mit der Berechnung ‘‘Sprünge’’ bestimmter Zustandsgrößen abgebildet werden sollen. Hierdurch können stark oszillierende, mitunter auch völlig falsche Ergebnisse entstehen.

Ein in der Literatur verwendetes Beispiel ist ein elastischer Stab, der auf eine starre Lagerung prallt (Abbildung 3.28), [Wri81]. Aus der analytischen Lösung weiß man, daß der Zeitver-

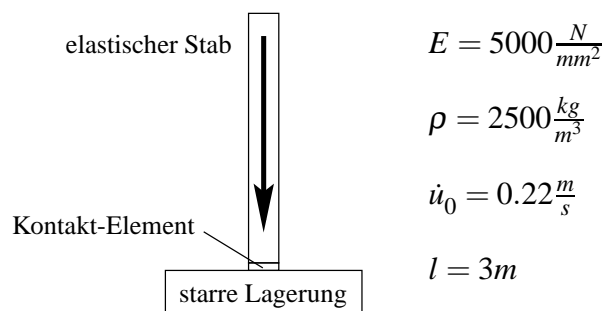


Abb. 3.28: Modell zur Kontaktsimulation

lauf der Kontaktspannung rechteckförmig ist und sich im Stab entsprechend fortsetzt. Die

Kontaktspannung  $\sigma_c$  kann über den Impulssatz ermittelt werden:

$$p = m \cdot \dot{u}_0 = \rho \cdot A \cdot l \cdot \dot{u}_0 \text{ und}$$

$$p = T \cdot F, \text{ mit } T = \frac{2l}{c}, \text{ wobei } c = \sqrt{\frac{E}{\rho}}. \quad (3.32)$$

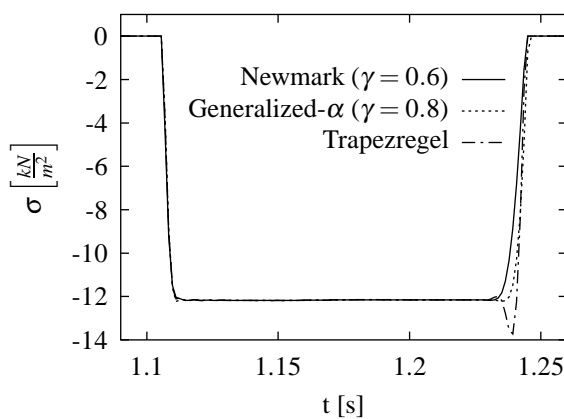
Nach einer Umformung ergibt sich die Kontaktkraft  $F$  sowie die Kontaktspannung  $\sigma_c$  zu:

$$F = \frac{\rho \cdot A \cdot \dot{u}_0 \cdot c}{2}, \quad \sigma_c = \frac{\rho \cdot \dot{u}_0 \cdot c}{2} \quad (3.33)$$

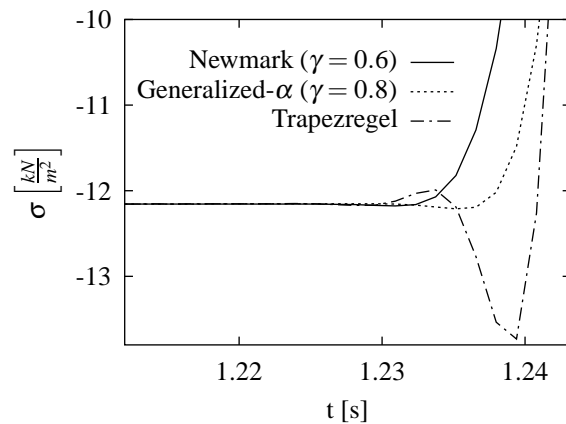
Im Stab ergibt sich die Normalspannung entsprechend dem Fortschreiten der Kompressionswelle. Am Stabende tritt während des gesamten Vorgangs aus Gleichgewichtsgründen keine Normalspannung auf.

Bei der numerischen Simulation kann dieser Vorgang nicht exakt abgebildet werden, da ein Sprung in den Zeitverläufen nicht möglich ist. Im Idealfall wird dieser Sprung innerhalb eines Zeitschritts linear approximiert. Zunächst erscheint es also angebracht, mit möglichst kurzen Zeitschritten zu rechnen, um den Vorgang möglichst genau zu erfassen. Wie bereits *Wriggers* in [Wri81] festgestellt hat, gibt es eine optimale Zeitschrittlänge für die *Trapezregel* und das *Newmark* Verfahren (siehe Gleichung 3.31, Seite 33). Nach Erfahrung des Autors sollte diese Schrittlänge auch für die  $\alpha$ -Methoden verwendet werden.

Zunächst ist zu untersuchen, ob die Kontaktspannung  $\sigma_c$  möglichst genau, d.h. rechteckförmig abgebildet wird. Aus Abbildung 3.29 wird deutlich, daß der Beginn des Stoßvorganges bei allen drei Verfahren befriedigende Ergebnisse liefert. Der Zeitpunkt, wo sich der Stab



**Abb. 3.29:** Zeitverlauf der Kontaktspannung ( $\Delta t_{opt} = 0.0015s$ )



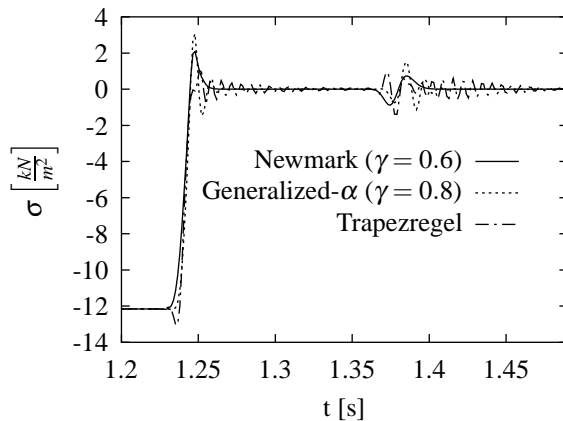
**Abb. 3.30:** Detail aus Abb. 3.29

wieder von der Lagerung trennt ( $t \approx 1.24s$ ), wird nur vom *Generalized-alpha* Verfahren gut abgebildet. Die der Verlauf der Kontaktspannung bei  $t \approx 1.24s$  hängt offensichtlich stark vom verwendeten Verfahren, also von der numerischen Dämpfung ab.

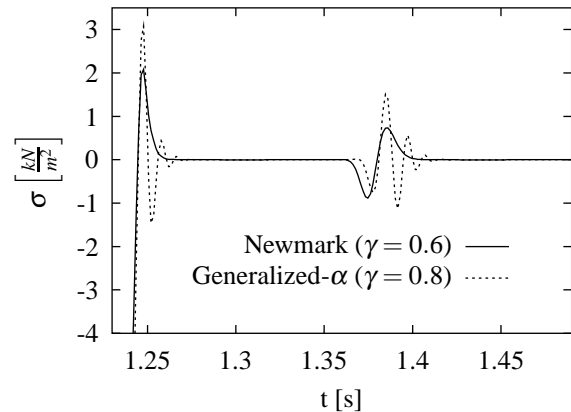
Das *Newmark* Verfahren dämpft die Strukturantwort zu stark, wodurch hier eine starke Ausrundung entsteht. Mit der *Trapezregel* entsteht ein leichtes "überschießen" der Spannung, was

auf eine (geringfügig) zu kurze Zeitschrittlänge hindeutet. Das *Generalized- $\alpha$*  Verfahren stellt einen guten Kompromiß aus zu starker und zu geringer numerischer Dämpfung dar.

Während die Kontaktspannung sehr gut abgebildet werden kann, ist die Berechnung der Normalspannung im Stab wesentlich schwieriger. Der Verlauf der Kompressionswelle, die durch



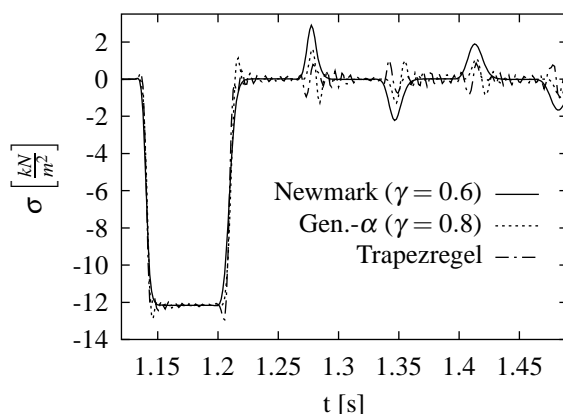
**Abb. 3.31:** Zeitverlauf der Längsspannung im Stab an der Kontaktstelle



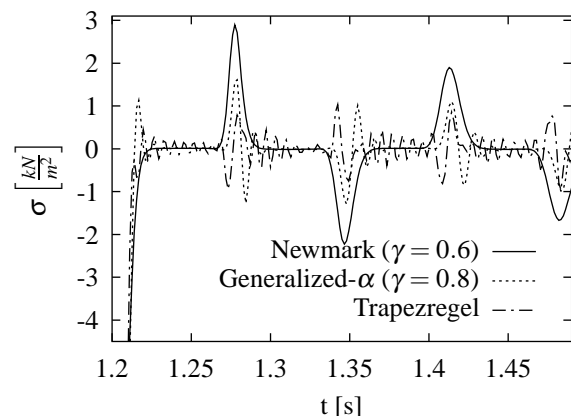
**Abb. 3.32:** Detail aus Abb. 3.31

den Stoßvorgang hervorgerufen wird, kann im gesamten Stab mit dem *Generalized- $\alpha$*  Verfahren sehr gut simuliert werden (Abbildungen 3.31 und 3.33).

Nach dem Stoßvorgang sollte der Stab wieder unbelastet sein, d.h. es dürfen im gesamten Stab keine Spannungen mehr vorhanden sein. Die Abbildungen 3.32 und 3.34 zeigen, daß sich noch hochfrequente Spannungen geringer Amplitude im Stab befinden. Diese treten bei der



**Abb. 3.33:** Zeitverlauf der Längsspannung in Stabmitte



**Abb. 3.34:** Detail aus Abb. 3.33

*Trapezregel* besonders hochfrequent auf – die numerisch dissipativen Methoden zeigen ein geringfügig besseres Verhalten. Manche Autoren verwenden deshalb das *Newmark* Verfahren mit  $\gamma \geq 0.6$ , um diese Oszillationen zu dämpfen [VV90, Wri81]. Dadurch wird der gesamte Stoßvorgang negativ beeinflusst, da sehr viel Energie numerisch dissipiert wird.

Die Ursache dieser hochfrequenten Störungen dürfte nicht nur in der Zeitintegration liegen, sondern auch von der Art der verwendeten Elemente zur Abbildung des Stabes sowie des Kontaktelements abhängen. Hier wurde das von *Temme* in [Tem95] beschriebene Element verwendet (siehe Abschnitt 6.2). Insbesondere die stark unterschiedlichen Eigenschaften der Kontaktschicht und der Kontinuums-Elemente (z.B. E-Modul) können eine beachtliche Störung hervorrufen.

Die numerische Behandlung dieses Problems ist auch mit anderen Verfahren problematisch. Die Veröffentlichungen von *Neumann/Schweizerhof* [NS99] und *Li/Ekevid/Wiberg* [Li96, WEL99] zeigen deutlich, daß es auch mit dem *Time-Discontinuous-Galerkin (TDG)* Verfahren kaum möglich ist, rechteckförmige Wellen korrekt abzubilden. Vor allem wird die korrekte Darstellung der Wellenform mit zunehmender Simulationsdauer erschwert (vgl. auch Abschnitt 3.2.6).

Ein ähnliche Beobachtung haben *Orden/Goicolea* in [OG99] bei der Simulation von elastischen und starren Mehrkörpersystemen gemacht. Sie vergleichen das *Hilber-Hughes-Taylor- $\alpha$*  Verfahren mit einer modifizierten *Energy-Momentum-Methode*. Aufgrund der äußerst hochfrequenten Strukturantwort erhalten sie mit dem *Hilber-Hughes-Taylor- $\alpha$*  Verfahren sehr unbefriedigende Ergebnisse im Hinblick auf die Abbildung hoher Frequenzen. Allerdings verwenden auch sie nach Erfahrung des Autors zu lange Zeitschritte ( $\Delta t \approx 1.5\Delta t_{opt}$ ).

### 3.5.2 Energetische Stabilität

Spektrale Stabilität bei linearen Berechnungen kann von vielen Algorithmen gewährleistet werden. Dies ist kein Garant für numerisch stabiles Verhalten bei geometrisch nichtlinearen Berechnungen [ST94].

Eine Besonderheit ergibt sich bei Berechnungen, bei denen die Steifigkeit der Struktur negativ wird (z.B. Durchschlagen). Hierdurch wird der unter Abschnitt 3.4.1 definierte Begriff der spektralen Stabilität unbrauchbar. Er kann durch eine *relative Stabilität* ersetzt werden (vgl. *Xie/Wood* [XW93]). *Xie/Wood* geben für diesen Fall auch eine Parametrisierung für das *Newmark* Verfahren an ( $\gamma = 0.5, \beta = \frac{1}{12}$ ), die jedoch nach Meinung des Autors noch genauer verifiziert werden müßte.

Zur allgemeinen Beurteilung von numerisch stabilem Verhalten eignet sich nach Auffassung vieler Autoren nur eine Betrachtung des energetischen Verhaltens des Zeitintegrationsverfahrens [ST95, AP97, KR96, Kuh96]. Aus der Forderung, daß die Gesamtenergie zum Ende des aktuellen Zeitschrittes kleiner oder gleich der Gesamtenergie zu Beginn des Schrittes sein muß, läßt sich bei Abwesenheit physikalischer Energie-Dissipation folgende Beziehung formulieren:

$${}^{t+\Delta t}E_{pot} - {}^tE_{pot} + {}^{t+\Delta t}E_{kin} - {}^tE_{kin} \leq W_{ext}, \quad \forall \Delta t \quad (3.34)$$

Diese Bedingung kann durch verschiedene Maßnahmen gewährleistet werden:

- Algorithmen, die eine Energieerhaltung implizit gewährleisten



- Erzwingen der Energieerhaltung durch Nebenbedingungen
- Numerische Dämpfung (algorithmische Dissipation)

### Implizite Energieerhaltung

Algorithmen, die eine Energieerhaltung implizit gewährleisten, wurden im wesentlichen von *Simo/Tarnow* für elastische Stäbe und Schalen entwickelt [ST92, ST94, ST95]. Für Kontakt-simulationen elastischer Körper gibt es verschiedene Weiterentwicklungen (z.B.: [AP97], [BDT95] und [OG99]). Diese Verfahren basieren auf einer Kombination von Energie- und Drehimpulserhaltung\* sowie einer Mittelpunktsregel. Ihr Haupteinsatzgebiet ist derzeit im Bereich der dynamischen Analyse von Mehrkörpersystemen sowie bei der Berechnung von großen geometrischen Nichtlinearitäten zu finden [CJ98, Dam98].

Zur Verhinderung von numerischen Instabilitäten bei der Gleichgewichtsiteration infolge hochfrequenter Störfrequenzen haben *Kuhl, Crisfield* und *Ramm* eine Kombination von impliziter Energieerhaltung und der *Generalized- $\alpha$*  Methode vorgestellt (*Generalized Energy Momentum Method, GEMM*) [KC99, KR98]. Dadurch geht die ursprüngliche Energieerhaltung verloren, da dem System entsprechend dem *Generalized- $\alpha$*  Verfahren numerisch Energie entzogen wird. Einen wesentlichen Nachteil dieser Algorithmen erwähnen *Crisfield/Shi* in [CS94]: Die Zeitintegration ist hiermit stark an die Finite-Element-Formulierung gekoppelt. Es ist dadurch kaum möglich, sehr unterschiedliche Elemente in einer Berechnung zu verwenden. Darüberhinaus führt die programmtechnische Umsetzung zu einem erhöhten numerischen Aufwand, da u.a. unsymmetrische Systemmatrizen auftreten [Mey98].

### Erzwungene Energieerhaltung

Ein alternatives Vorgehen wird bei einer erzwungenen Energieerhaltung verwendet. Hierbei wird die Energieerhaltung durch eine Nebenbedingung (z.B. *Lagrange Multiplikatoren*) eingebracht. Diese Methode wurde ursprünglich in [HCL78] veröffentlicht. *Kuhl* und *Ramm* kombinierten diesen Algorithmus mit der *Generalized- $\alpha$*  Methode zum *Constraint Energy Momentum Algorithm (CEMA)*, der neben den ursprünglichen Eigenschaften der *Generalized- $\alpha$*  Methode (vgl. Abschnitt 3.4) somit auch impuls- und drallerhaltend ist. Dies hat zur Folge, daß bei einer Dissipation hoher Frequenzen ein Energietransfer von den hohen Frequenzen zu den niedrigen Frequenzen stattfindet [Kuh96]. Hierdurch kann eine Verfälschung der Ergebnisse eintreten, wie ein Vergleich von *EMM* und *CEMA* in der Arbeit von *Kuhl* zeigt.

Einen weiteren Problempunkt stellt die Zunahme des numerischen Aufwandes dar: Nach [KR96] ist ein Anstieg der Rechenzeit von ca. 50 % im Vergleich zum ursprünglichen *Generalized- $\alpha$*  Verfahren festzustellen.

Alle energieerhaltenden Verfahren sind zwar meist numerisch sehr robust, doch kann mit ihnen kaum eine physikalische Dissipation einbezogen werden. Lediglich für genau definierte

---

\*daher auch der Name: *Energy-Momentum-Method, EMM*

Zustände (z.B. Reibung) läßt sich eine Modifikation durchführen [CL98]. *Crisfield et al* verwenden hierzu die Arbeit von *Armero* und *Petöcz* [AP97]. Für spezielle Balken-Elemente kann hiermit eine physikalische Dämpfung mit dem ursprünglich energieerhaltenden Verfahren kombiniert werden [CJG99].

Das Einsatzgebiet dieser Verfahren ist sehr beschränkt, zumal ihre Formulierung eng an die eigentliche Finite-Element-Formulierung gekoppelt ist, wodurch jeweils nur eine bestimmte Art einer Struktur simuliert werden kann. Für den in der vorliegenden Arbeit gewünschten Einsatz bei gekoppelten Strukturen mit einer beliebigen physikalischen Dämpfung sind diese Verfahren also nicht geeignet. Physikalische Dämpfung könnte nur dann einbezogen werden, wenn man die jeweils im aktuellen Zeitschritt dissipierte Energie a priori ermitteln könnte. Dies ist jedoch vor allem bei komplexen Systemen unmöglich.

### Numerische Dämpfung

Wie in Kapitel 3.4.3 bereits ausführlich gezeigt wurde, gibt es zahlreiche Algorithmen, die eine kontrollierbare numerische Dämpfung erlauben. Im folgenden ist zu untersuchen, ob diese Eigenschaft zu einem uneingeschränkt numerisch stabilen Verhalten führt.

In einer frühen Veröffentlichung wurde mit Hilfe von analytischen Untersuchungen versucht, eine sichere Aussage zum numerischen Stabilitätsverhalten der Trapezregel für nichtlineare elastische Problemstellungen zu finden [WH81]. Mittlerweile ist unstrittig, daß die Trapezregel (ungedämpftes *Newmark* Verfahren) bei geometrisch nichtlinearen Berechnungen zu einem numerisch instabilen Verhalten neigt. *Simo* und *Tarnow* [ST94] und viele andere (z.B.: [Mey98]) zeigen dies: Nach einem anfänglich stabilen Verhalten steigt sukzessive die Gesamtenergie im System an und wächst plötzlich über alle Grenzen (*energy blow up*).

Dieser Prozeß ist stark vom Umfang der Nichtlinearitäten abhängig. Während bei Systemen mit großen Rotationen (z.B.: sich im Raum drehende Körper) die Gesamtenergie nahezu schlagartig explodiert, kann dieser Vorgang bei schwach nichtlinearen Systemen meist nicht beobachtet werden.

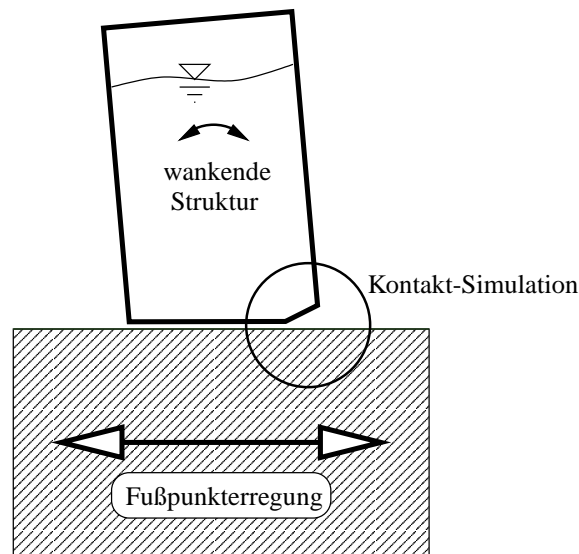
Eine weitere negative Eigenschaft der Trapezregel stellt das *locking* dar [CS94]. Hierbei springt das Systemverhalten plötzlich in einen völlig anderen Zustand. Dies geht mit einem drastischen Energiezuwachs einher und tritt vor allem bei relativ langen Zeitschritten auf.

Am Beispiel eines “einfachen Pendels” (Punktmasse mit Freiheitsgraden in Radial- und Umfangsrichtung) untersuchen *Kuhl* und *Crisfield* in [KC99] geeignete Parametrisierungen für verschiedene Verfahren. Für das gewählte Beispiel erhalten sie erst mit einem Spektralradius von  $\rho \approx 0.6$  ( $\gamma \approx 0.75$ ) für alle dissipativen Verfahren ein numerisch stabiles Ergebnis. Bei einer so starken numerischen Dämpfung ist nur noch das *Generalized- $\alpha$*  Verfahren einsetzbar, da die anderen Methoden die Strukturantwort stark verfälschen (siehe Abschnitt 3.4.3).

Numerisch instabiles Verhalten äußert sich nach [KC99] bei allen Verfahren in ähnlicher Weise. Ausgehend von stetig anwachsenden Oszillationen wächst die Gesamtenergie des Systems plötzlich über alle Grenzen. Charakteristisch ist für diese Oszillationen, daß ihre Periodendauer kleiner als zwei Zeitschritte ist. Nach Abbildung 3.6 (Seite 28) können diese Frequenzen also nicht mehr eindeutig abgebildet werden.

In der angegebenen Literatur zur erzwungenen oder impliziten Energieerhaltung werden meist “akademische” Problemstellungen untersucht (z.B. frei im Raum fliegende Körper mit großen Rotationen und Translationen [CJ98, Kuh96, ST94]), ohne daß ein allgemeines, numerisches Stabilitätskriterium für geometrisch nichtlineare Berechnungen gefunden wurde.

Im Folgenden wird deshalb das Verhalten der wichtigsten impliziten Verfahren bei einer komplizierten, gekoppelten Struktur untersucht. Im Gegensatz zur erwähnten Literatur wird hier nicht von großen Rotationen ausgegangen. Vielmehr werden die Berechnungen mit einem im Bauwesen üblichen Ansatz moderater Rotationen vorgenommen [WCO85, WOSL87]. Das physikalische System (Behälter-Flüssigkeit-Boden) sowie das verwendete Strukturmo-

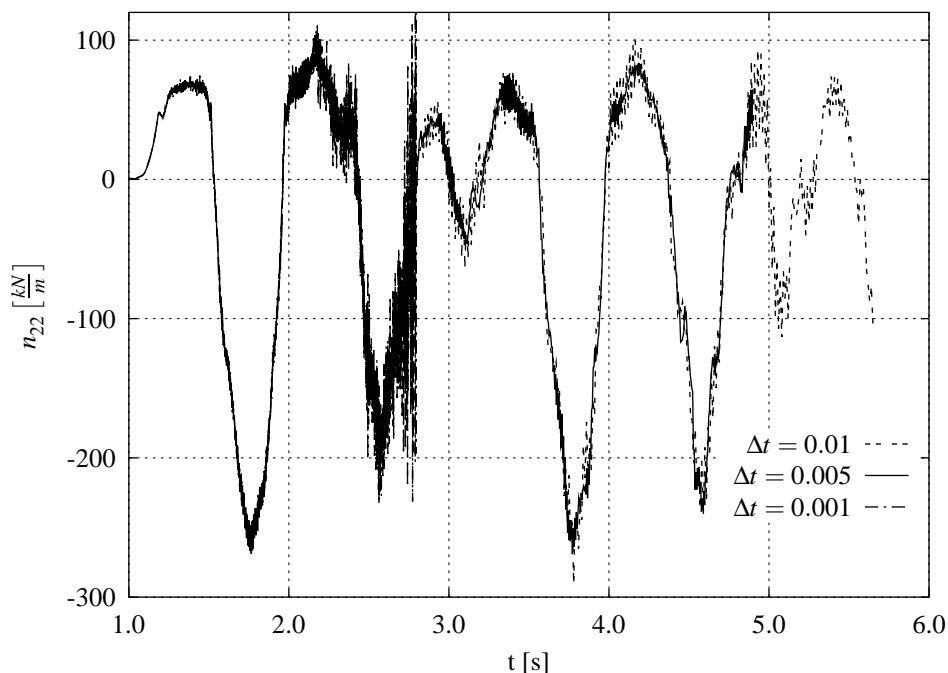


**Abb. 3.35:** Flüssigkeitsgefüllter Behälter unter Fußpunkterregung

dell wird ausführlich in Kapitel 6 dargestellt. Für die Simulation wird ein Tank der Reihe “T9” (vgl. Tabelle 6.1, Seite 105) mit linear-elastischem Boden und einer harmonischen Fußpunkterregung mit einer Frequenz von 1.5 Hz verwendet.

Zunächst ist zu untersuchen, ob bereits eine geringe numerische Dämpfung den Rechenprozeß ausreichend stabilisieren kann. Neben einer charakteristischen Antwortgröße (hier: Meridiankraft  $n_{22}$  am Behälterfuß) wird die während der Simulation numerisch dissipierte Energie gemessen. Die Gesamtenergie des Systems ist hier ein ungeeigneter Indikator, da ständig äußere Energie (Anregung) zugeführt wird – gleichzeitig jedoch Energie infolge von Wellenabstrahlung in den Halbraum und plastischer Effekte dissipiert wird.

Die Meridiankraft zeigt für die Trapezregel einen typischen Verlauf: Beim ersten Auftreten starker Nichtlinearitäten bei  $t \approx 1.3s$  entstehen erste Oszillationen (Abbildung 3.36). Im weiteren Verlauf der Berechnung nehmen sie teilweise stark zu und scheinen dann wieder abzuklingen. Dies könnte eine Folge von physikalischer und geometrischer Dämpfung sein. Für den gewählten Zeitschritt ( $\Delta t = 0.005s$ ) wird die numerische Simulation bei  $t \approx 5s$  instabil, wobei die Gleichgewichtsiteration versagt. Verkürzt man die Zeitschrittlänge, so kann meist ein noch früheres Versagen der Berechnung beobachtet werden. Bei einer Schrittlänge von  $\Delta t = 0.001s$  wird die Simulation bereits bei  $t \approx 2.8s$  instabil. Eine Verlängerung des Schrittweite ( $\Delta t = 0.01s$ ) führt hingegen erst bei  $t \approx 5.6s$  zu einem Berechnungsabbruch infolge



**Abb. 3.36:** Meridiankraft am Behälterfuß mit Trapezregel

numerischer Instabilität. Das Verfahren ist also nicht im Sinne von Gleichung (3.24) bedingt numerisch stabil, sondern kann bei nichtlinearen Berechnungen sogar bei einer Verkürzung der Zeitschrittlänge instabil werden.

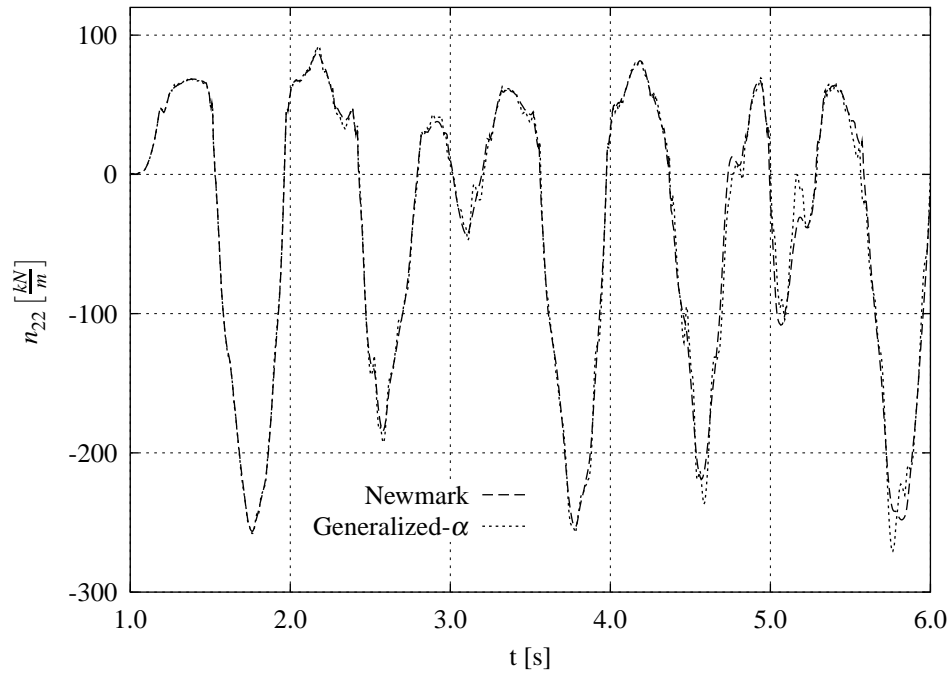
Die beiden anderen Verfahren weisen trotz der relativ geringen numerischen Dämpfung ( $\gamma = 0.55$ ) einen numerisch robusten Verlauf auf (Abbildung 3.37). Allerdings ist eine höhere numerische Dämpfung sinnvoll ( $\gamma \geq 0.6$ ), da hierdurch die Simulation auch bei starken geometrischen Nichtlinearitäten stabil bleibt [KC99]. Bei einer starken numerischen Dämpfung ist das *Generalized- $\alpha$*  Verfahren empfehlenswert, damit das Simulationsergebnis – insbesondere die hohen Frequenzen – nicht zu stark verfälscht wird.

Mit Hilfe von Abbildung 3.38 wird der Zusammenhang zwischen numerischer Energie-Dissipation und *energy blow up* deutlich. Während bei der Trapezregel sukzessive Energie in Form von hochfrequenten Moden zugeführt wird (negative dissipierte Energie), wird bei den dissipativen Verfahren hochfrequente Energie entzogen.

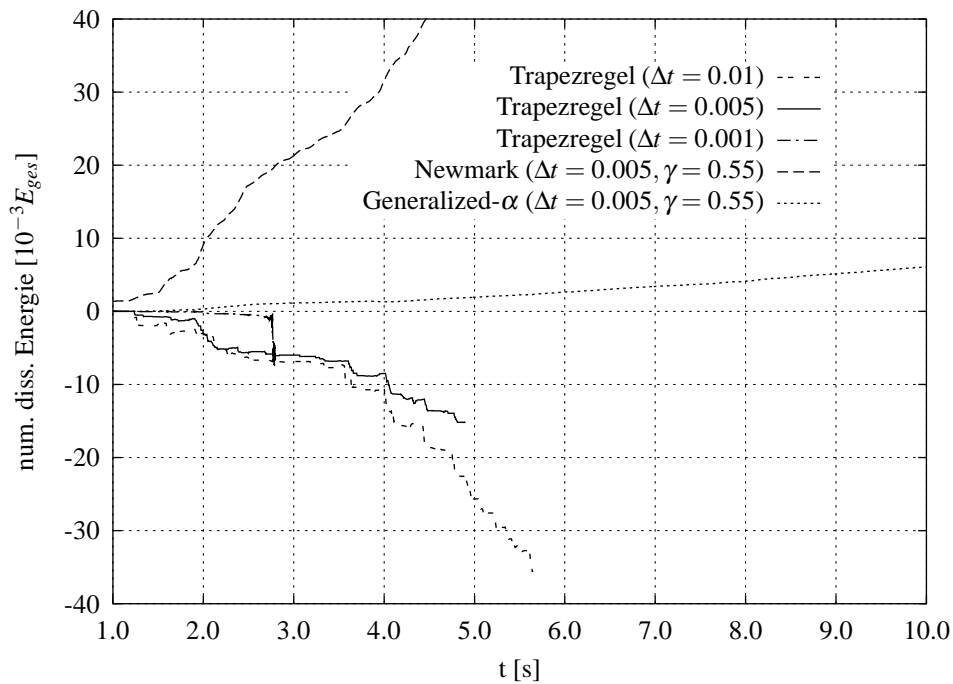
Wie bereits mehrfach erwähnt, wird auch hier mit dem *Newmark* Verfahren sehr viel Energie numerisch dissipiert. In dem gezeigten Beispiel sind dies am Ende der Simulation ca. 10 % der zugeführten äußeren Energie. Das *Generalized- $\alpha$*  Verfahren dissipiert dagegen weniger als 0.7 %.

Die numerische Instabilität zeigt sich letztendlich in einer divergenten Gleichgewichtssiteration. Die Simulation mit der Trapezregel benötigt bereits lange vor dem letzten Zeitschritt an einigen Stellen außergewöhnlich viele Iterationsschritte, wodurch die Summe der Iterationen bei gleicher Zeitschrittlänge stärker anwächst als bei den dissipativen Verfahren (Abb. 3.39).

Insgesamt läßt sich feststellen, daß die Anzahl der Iterationen stark von der Höhe der numerischen Dissipation abhängt: Je stärker die numerische Dämpfung, umso geringer ist die

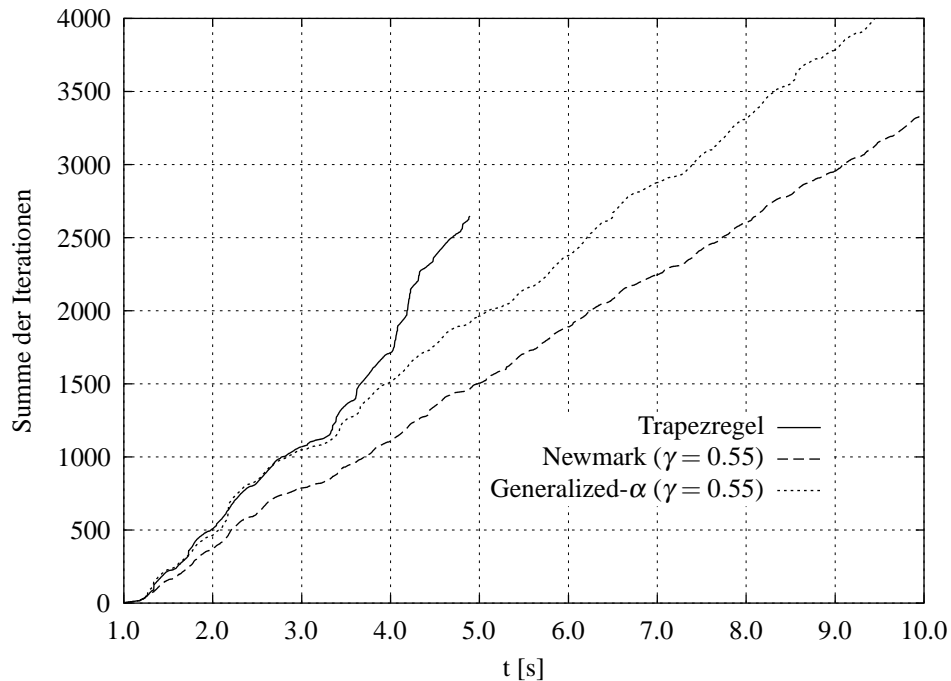


**Abb. 3.37:** Meridiankraft am Behälterfuß mit *Newmark* und *Generalized- $\alpha$*  Verfahren ( $\Delta t = 0.005$ ,  $\gamma = 0.55$ )



**Abb. 3.38:** Numerisch dissipierte Energie

Anzahl der Iterationsschritte. Eine numerische Dämpfung führt deshalb nicht nur zu einer robusten Simulation, sondern auch zu einer effizienten Berechnung. Mit dem *Generalized- $\alpha$*  Verfahren kann hier ein optimaler Kompromiß aus effizienter und genauer Berechnung gefunden werden.



**Abb. 3.39:** Anzahl der summierten Gleichgewichtsiterationen für verschiedene Verfahren ( $\Delta t = 0.005\text{ s}$ )

Bei geometrisch nichtlinearen Berechnungen mit moderaten Rotationen führt also eine numerische Dämpfung zu einer numerisch stabilen Simulation. Es ist jedoch möglich, daß bei Verwendung von Elementen mit beliebig großen Nichtlinearitäten (z.B. [CS94, Kuh96, ST94]) ein Verfahren mit erzwungener oder impliziter Energieerhaltung notwendig ist.

### 3.5.3 Genauigkeit

Im Gegensatz zur analytischen Genauigkeitsuntersuchung bei linearen Berechnungen (siehe Abschnitt 3.4.2) können im Nichtlinearen Genauigkeitsschätzungen nur anhand von numerischen Untersuchungen vorgenommen werden. Neben dem Phasenfehler kann als globaler Vergleichsparameter die numerisch dissipierte Energie dienen. Weitergehende Untersuchungen beinhalten charakteristische Antwortgrößen der Simulation (z.B. Spannungen).

Im linearen Bereich wurde bereits festgestellt, daß die unerwünscht starke Dissipation des *Newmark* Verfahrens das Rechenergebnis negativ beeinflusst (siehe Abschnitt 3.4.3). Die Ergebnisse der nichtlinearen Berechnung zeigen ein nahezu identisches Verhalten (vgl. Abbildungen 3.17 und 3.38). Auch hier dämpft die *Generalized-α* Methode das System nur sehr schwach, wodurch das Rechenergebnis kaum verfälscht wird.

Aufgrund der positiven Eigenschaften des *Generalized-α* Verfahrens wird diese Methode bei allen nun folgenden Berechnungen verwendet.

# Kapitel 4

## Adaptive Zeitschrittsteuerung

Die Zeitschrittlänge stellt einen wesentlichen Parameter bei der nichtlinearen dynamischen Strukturanalyse dar. Sie steuert nicht nur die Genauigkeit der Berechnung, sondern kann auch die numerische Stabilität der Simulation beeinflussen (vgl. Abschnitt 3.5). Darüberhinaus hängt vor allem der numerische Aufwand von der Anzahl der Zeitschritte ab.

Die Zeitschrittlänge muß deshalb aus einem sinnvollen Kompromiß zwischen möglichst genauer Berechnung (kurze Schrittweite) und möglichst geringem Rechenaufwand (meist möglichst lange Schrittweite) gefunden werden.

Da sich das Strukturverhalten während der Simulation laufend ändert, kann eine dem aktuellen Zustand angepaßte adaptive Schrittweite sinnvoll sein. Gerade im Hinblick auf eine effiziente Berechnung großer FE-Systeme ist eine adaptive Schrittweitensteuerung von sehr großer Bedeutung. Hierzu sind jedoch geeignete Verfahren erforderlich, um eine zuverlässige Steuerung zu erhalten. Diese Algorithmen lassen sich in drei Teile gliedern:

- Fehlerschätzer (bzw. -indikatoren) zur Ermittlung des Integrationsfehlers
- Eine darauf aufbauende Steuerung der Integrationsschrittweite
- Problemangepaßte Zusatzverfahren, um evtl. vorhandene Schwächen des Fehlerschätzers auszuschließen

Eine alternative Vorgehensweise basiert auf dem *Rayleigh-Quotienten* [BM84, BM85, CP93, WGS85]:

$$\omega_s = \sqrt{\frac{\Delta \mathbf{u}^T \mathbf{K}_T \Delta \mathbf{u}}{\Delta \mathbf{u}^T \mathbf{M} \Delta \mathbf{u}}} \quad (4.1)$$

Hierbei stellt  $\omega_s$  die *signifikante Frequenz* dar, deren Periodenlänge mit ca. 10 Integrationschritten abgebildet werden soll (vgl. Abbildung 3.6, Seite 28). Nach Wunderlich, Goebel und Springer [WGS85] muß allerdings zusätzlich ein *Steifeparameter*  $s$  [BM84] verwendet werden, um bei abnehmender Systemsteifigkeit ( $\omega_s$  wird kleiner) nicht zu lange Schrittweiten zu erhalten:

$$t + \Delta t_s = \frac{\Delta \mathbf{u}^T \mathbf{K}_T \Delta \mathbf{u}}{\Delta \mathbf{u}^T \mathbf{K}_L \Delta \mathbf{u}} \quad (4.2)$$

Hieraus kann die Schrittweite für den nächsten Integrationsschritt ermittelt werden:

$$\Delta t = \frac{1}{n} {}^{t+\Delta t} s \frac{2\pi}{\omega_s}, \text{ mit } n \geq 10 \quad (4.3)$$

Damit bei einem starken Steifigkeitsverlust keine allzu kurzen Integrationsschritte oder bei einem Anstieg nicht zu lange Schritte verwendet werden, schlagen *Wunderlich, Goebel* und *Springer* vor, daß man eine untere und eine obere Schranke der Schrittweite a-priori angibt. In Abschnitt 4.4 wird diese Methode mit dem in den folgenden Abschnitten dargestellten Verfahren verglichen.

Ein ähnliches Vorgehen verwendet *Dinkler* [Din88]. Er schätzt die maßgebende Eigenfrequenz im aktuellen Zeitschritt über

$$\omega_D = \sqrt{{}^t \omega + \frac{1}{2} ({}^t \omega - {}^{t-\Delta t} \omega)} \quad (4.4)$$

ab. Die Zeitschrittlänge ergibt sich aus

$$\Delta t = \frac{2}{\omega_D} \sqrt{\frac{\varepsilon_\psi}{1 + \varepsilon_\psi}}, \text{ mit } \varepsilon_\psi \approx 10. \quad (4.5)$$

Die ermittelte charakteristische Frequenz  $\omega_s$  bzw.  $\omega_D$  ist jedoch ein fragwürdiger Indikator. Beispielsweise liefert er bei einem ungedämpften Einmassenschwinger mit linearer Federsteifigkeit immer einen konstanten Wert. Es läßt sich aber zeigen, daß der tatsächliche Integrationsfehler stark zyklisch schwankt [Ric97]. Auch bei ausgeprägt nichtlinearem Strukturverhalten ist die Abschätzung mit Unsicherheiten verbunden [WGS85, Kuh96].

Es ist deshalb zu untersuchen, ob mit einem geschätzten Integrationsfehler eine bessere Adaption der Schrittweiten erreicht werden kann.

Es soll hier nur die adaptive Zeitschrittsteuerung, nicht jedoch eine räumliche Netzanpassung untersucht werden. Die Netzadaption kann mit den bekannten Methoden (h- oder p-Adaption) nachgeschaltet werden [Li96, Ric97].

## 4.1 Integrationsfehler

Da die exakte Lösung des zu berechnenden Problems nicht bekannt ist, muß ein Verfahren gefunden werden, mit dem die Genauigkeit des aktuell berechneten Schritts abgeschätzt werden kann. Hierzu könnte eine zweite, simultane Berechnung mit einer kürzeren Schrittweite hilfreich sein. Allerdings ist der numerische Aufwand hierfür inakzeptabel. Ebenso ist eine parallele Berechnung mit einem Verfahren höherer Ordnung nicht geeignet. Trotzdem lassen sich für die hier untersuchten impliziten Einschnitt-Verfahren a-posteriori Fehlerindikatoren entwickeln.



Für linear elastische Problemstellungen gibt es für die *Newmark-Familie* einige Ansätze, die im wesentlichen auf einer Taylor-Reihenentwicklung der Verschiebungen bzw. Geschwindigkeiten beruhen.

*Zienkiewicz* und *Xie* leiten in [ZX91] einen einfachen Fehlerindikator für das *Newmark* Verfahren her:

$$\varepsilon_{ZX} = \left( \beta - \frac{1}{6} \right) \Delta t^2 \left\| {}^{t+\Delta t} \ddot{\mathbf{u}} - {}^t \ddot{\mathbf{u}} \right\|_2 \quad (4.6)$$

Dieser Fehlerschätzer für die Verschiebungen ist von dritter Ordnung genau [Ric97] und kann damit für Zeitintegrationsverfahren eingesetzt werden, die von zweiter Ordnung genau sind. Offensichtlich liefert er für  $\beta = \frac{1}{6}$  kein Ergebnis. Diese Parametrisierung wird ohnehin meist nicht verwendet, da sie nur bedingt spektral stabil ist (vgl. Abschnitt 3.4.1, Seite 25 ff).

Mit einem sehr ähnlichen Vorgehen erhalten *Li*, *Zeng* und *Wiberg* ebenfalls einen Fehlerschätzer für Verschiebungen [LZW93]:

$$\varepsilon_{LZW} = \frac{1}{12} \Delta t^3 \left\| \left( 6\beta - \frac{1}{2} \right) {}^{t+\Delta t} \dot{\mathbf{u}} - \left( \frac{3}{2} - 6\beta \right) {}^t \dot{\mathbf{u}} \right\|_2 \quad (4.7)$$

Die Ableitungen der Beschleunigung  ${}^{t+\Delta t} \dot{\mathbf{u}}$  können über Differenzenquotienten ermittelt werden:

$${}^{t+\Delta t} \dot{\mathbf{u}} \approx 2 \frac{{}^{t+\Delta t} \ddot{\mathbf{u}} - {}^t \ddot{\mathbf{u}}}{\Delta t} - {}^t \ddot{\mathbf{u}} \quad (4.8)$$

oder alternativ über:

$${}^{t+\Delta t} \dot{\mathbf{u}} \approx \frac{{}^{t+\Delta t} \ddot{\mathbf{u}} - {}^{t-\Delta t} \ddot{\mathbf{u}}}{2\Delta t} \quad \text{vorausgesetzt, daß } \Delta t = \text{const} \quad (4.9)$$

Der Ansatz nach Gleichung (4.8) ist für eine adaptive Berechnung besser geeignet, da hier nur ein konstanter Zeitschritt erforderlich ist, während bei Gleichung (4.9) zwei konstante Schritte vorhanden sein müssen. Durch diese Approximation ist nach [Kuh96] auch dieser Fehlerindikator nur von dritter Ordnung genau.

Durch die Einbeziehung des Fehlers der Geschwindigkeiten

$$\dot{\varepsilon}_{LW} = \frac{1}{6} \Delta t^2 \left\| (3\gamma - 1) {}^{t+\Delta t} \dot{\mathbf{u}} + (3\gamma - 2) {}^t \dot{\mathbf{u}} \right\|_2 \quad (4.10)$$

erreichen *Li* und *Wiberg* einen Fehler der Energienorm [Li96, WL94]:

$$\varepsilon_E = \sqrt{\dot{\varepsilon}_{LW}^T \cdot \mathbf{M} \cdot \dot{\varepsilon}_{LW} + \varepsilon_{LZW}^T \cdot \mathbf{K}_T \cdot \varepsilon_{LZW}} \quad (4.11)$$

Hierbei ist zu beachten, daß die Terme  $\varepsilon_{LZW}$  und  $\dot{\varepsilon}_{LW}$  nicht als  $L_2$ -Norm, sondern als Vektoren einzusetzen sind. Nach *Kuhl* [Kuh96] ist der so ermittelte Fehler der Energienorm nur

von zweiter Ordnung genau, da der Fehler der Geschwindigkeiten (Gleichung 4.10) diese Genauigkeitsordnung aufweist.

*Kuhl* zeigt mit Hilfe von Vergleichsrechnungen, daß der einfache Fehlerschätzer nach *Zienkiewicz* und *Xie* bessere Ergebnisse liefert als die von *Wiberg et al.*

Mit einem verallgemeinerten Vorgehen bestätigt *Riccus* [Ric97] nochmals die hier gezeigten Zusammenhänge. Er leitet darüberhinaus Fehlerschätzer für die Verschiebungen (Genauigkeit vierter Ordnung) und die Geschwindigkeiten (Genauigkeit dritter Ordnung) auf der Basis von jeweils drei Beschleunigungsvektoren zu verschiedenen Zeitpunkten her (siehe auch [NS99]):

$$\begin{aligned}\varepsilon_R &= \frac{\Delta t^2}{24} \left( {}^{t-\Delta t}\ddot{\mathbf{u}} + (2 - 24\beta)^t \ddot{\mathbf{u}} + (24\beta - 3)^{t+\Delta t} \ddot{\mathbf{u}} \right) \\ \dot{\varepsilon}_R &= \frac{\Delta t}{12} \left( {}^{t-\Delta t}\ddot{\mathbf{u}} + (4 - 12\gamma)^t \ddot{\mathbf{u}} + (12\gamma - 5)^{t+\Delta t} \ddot{\mathbf{u}} \right), \quad \Delta t = \text{const.} \quad (4.12)\end{aligned}$$

Diese beiden Indikatoren können analog Gleichung (4.11) zu einem Fehler der Energienorm zusammengefaßt werden. Auch *Choi* und *Chung* verwenden in [CC99] diese Vorgehensweise für die *Newmark* und die *Wilson* Methode.

Auch *Riccus* zeigt, daß der Fehlerschätzer nach *Zienkiewicz* und *Xie* (Gleichung 4.6) sehr gute Ergebnisse liefert. Die relativ aufwendige Vorgehensweise nach Gleichung (4.12) ist deshalb nicht notwendig. Sie kann ohnehin nur dann eingesetzt werden, wenn im betrachteten Intervall eine konstante Zeitschrittlänge verwendet wird. Gerade dies ist jedoch bei einer adaptiven Zeitschrittsteuerung nicht immer gegeben.

Alle Fehlerindikatoren zeigen bei einer hochfrequenten Strukturantwort (relativ zur aktuellen Schrittweite) sehr große Integrationsfehler an. Dies führt bei Verwendung der Trapezregel in Verbindung mit einer adaptiven Schrittweitensteuerung zu immer kleineren Integrationslängen. Ein numerisch dissipatives Verfahren ist deshalb auch im Zusammenhang mit einer automatischen Schrittweitensteuerung von großem Vorteil [Kuh96].

Im folgenden wird nur der Fehlerschätzer nach *Zienkiewicz* und *Xie* (Gleichung 4.6) verwendet, da er aufgrund der erwähnten Vorteile (einfach zu implementieren, Genauigkeit dritter Ordnung, gutes numerisches Verhalten, nur ein Zeitintervall notwendig) den anderen Indikatoren überlegen ist.

## 4.2 Relativer Fehler - Bezugsgröße

Die im vorhergehenden Abschnitt dargestellten Fehlerschätzer liefern jeweils einen absoluten Wert, der je nach Art (Translationen – Rotationen) und Anzahl der Freiheitsgrade, Art der Elementformulierung sowie abhängig von der Strukturantwort stark unterschiedlich ist. Der Wert ist also in dieser Form unbrauchbar, um als allgemeiner Steuerparameter zu dienen.

Darüberhinaus werden bei der Bildung der  $L_2$ -Norm die Translationen mit den Rotationen vermischt. Diese haben jedoch unterschiedliche physikalische Einheiten. Als Lösung dieses

Problems bietet sich an, die Translationen und Rotationen jeweils getrennt voneinander zu bearbeiten. Alternativ kann eine Energiegröße über

$$\varepsilon_{En} = \sqrt{\varepsilon_{ZX}^T \cdot \mathbf{K}_T \cdot \varepsilon_{ZX}} \quad (4.13)$$

gebildet werden.

Um eine allgemein verwendbare Fehlerkenngröße zu erhalten, muß der absolute Fehler auf eine zugeordnete Größe bezogen werden. Man erhält hiermit einen relativen Fehler:

$$\varepsilon_{rel} = \frac{\varepsilon_{ZX}}{\|\mathbf{u}_{ref}\|_2} \quad (4.14)$$

Als Referenzwert sind verschiedene Möglichkeiten vorhanden:

- Die Norm des aktuellen Verschiebungsincrementes  $\|\Delta\mathbf{u}\|_2$
- Die Norm des Gesamtverschiebungsvektors im aktuellen Zeitpunkt  $\|\mathbf{u}\|_2$
- Die maximale Verschiebungsnorm  $\|\mathbf{u}_{max}\|_2$ , die bis zum aktuellen Zeitpunkt aufgetreten ist

Um während der gesamten Berechnung ein möglichst gleichmäßiges Genauigkeitsniveau zu erhalten, wäre die Norm des aktuellen Verschiebungsincrementes die beste Wahl. Allerdings ergibt sich hier das Problem, daß bei manchen Zeitpunkten diese Norm gegen Null strebt. Der somit berechnete relative Fehler ist dann unbrauchbar.

Die Norm des Gesamtverschiebungsvektors im aktuellen Zeitpunkt ist mit derselben Problematik behaftet. Diese Norm ist ohnehin eine zweifelhafte Größe, da der damit ermittelte relative Fehler bei großen Verformungen zu kleine, bei kleinen Verformungen zu große Fehler anzeigt.

Der relative Fehler, der mit der maximalen Verschiebungsnorm (die bis zum aktuellen Zeitpunkt aufgetreten ist) ermittelt wird, zeigt am Anfang der Berechnung zu kleine Werte an. Erst nach dem auftreten der maximalen Verschiebungsnorm wird ein einheitliches Genauigkeitsniveau innerhalb der aktuellen Berechnung erreicht. Diese in [Kuh96, WL94] beschriebene Vorgehensweise ist deshalb nur bei harmonischen Strukturantworten sinnvoll.

Eine Alternative zu den drei hier dargestellten Normen wird in [RW99] angegeben. Aus den  $L_2$ -Normen der Verschiebungsincremente wird eine Summe gebildet, die bezogen auf eine a priori vorzugebende Referenzschrittweite  $\Delta t_{ref}$  sowie auf das bisher berechnete Zeitintervall  $[t_0 : t_n]$  eine mittlere inkrementelle Verschiebungsnorm ergibt:

$$\|\mathbf{u}_{ref}\|_2 = \sum_{i=0}^n \|\Delta\mathbf{u}_i\|_2 \cdot \frac{\Delta t_{ref}}{t_n - t_0} \quad (4.15)$$

Die Größe der Referenzschrittweite  $\Delta t_{ref}$  muß der Anwender entsprechend der zu erwartenden Strukturantwort vorgeben.

Wie die nachfolgenden Beispiele zeigen, erreicht die mit Gleichung (4.15) ermittelte Bezugsgröße erst nach einigen Zeitschritten ein relativ konstantes Niveau. Deshalb wird zu Beginn der Simulation mit einer konstanten Schrittweite gerechnet. Erst ab einer vorgegebenen Anzahl an Zeitschritten wird die nachfolgend dargestellte Schrittweitenanpassung aktiviert.

### 4.3 Schrittweitenanpassung

Die Modifikation der Zeitschrittlänge erfolgt entsprechend dem relativen Fehler aus Gleichung (4.14). Überschreitet der Fehler eine vorgegebene Schranke, dann muß der aktuelle Berechnungsschritt mit einem kleineren Inkrement wiederholt werden. Ist der Fehler sehr klein, dann kann der nächste Zeitschritt mit einem größeren Inkrement berechnet werden. Für die Anpassung der Schrittweite wird in allen hier diskutierten Veröffentlichungen eine analoge Vorgehensweise verwendet [LZW93, RW99]. Hier wird nun eine Modifikation für nichtlineare Berechnungen vorgestellt.

Die Verkürzung des aktuellen Zeitschrittes kann über

$$\Delta t_{neu} = \tau \sqrt[3]{\frac{\varepsilon_{max}}{\varepsilon_{rel}}} \Delta t_{alt}, \quad \text{für } \varepsilon_{rel} > \varepsilon_{max} \quad (4.16)$$

erfolgen. Hierbei gibt  $\tau$  einen zusätzlichen Sicherheitsparameter an, der bei nichtlinearen Berechnungen verhindern soll, daß die Schrittweite mehrmals hintereinander verkürzt werden muß. Für  $\tau$  können Werte zwischen 0.8 (große Nichtlinearität) und 1.0 (lineares Verhalten) verwendet werden. Die Bestimmung von  $\tau$  kann entweder a priori erfolgen oder anhand der Anzahl der Iterationen des aktuellen Zeitschrittes abgeschätzt werden: Bei wenigen Iterationen wird  $\tau = 1.0$  gesetzt, bei vielen Iterationen wird  $\tau = 0.8$  gesetzt.

Die Verlängerung der Schrittweite erfolgt analog über

$$\Delta t_{neu} = \tau \sqrt[3]{\frac{\varepsilon_{min}}{\varepsilon_{rel}}} \Delta t_{alt}, \quad \text{für } \varepsilon_{rel} < \varepsilon_{min} \quad (4.17)$$

Mit dem hier gezeigten Vorgehen ist es auch möglich, für beliebige Systeme feste Fehlerschranken für  $\varepsilon_{max}$  und  $\varepsilon_{min}$  anzugeben. Nach Erfahrung des Autors sollte für  $\varepsilon_{max} \approx 0.5\%$  und für  $\varepsilon_{min} \approx 0.1\%$  verwendet werden. Kleinere Werte führen zu einer genaueren, größere Werte zu einer weniger genauen Berechnung. Der Nutzer kann mit der Referenzschrittweite  $\Delta t_{ref}$  die gewünschte Genauigkeit der Berechnung vorgeben.

Die Verlängerung der Schrittweite sollte nach einem Vorschlag in [LZW93] nur dann erfolgen, wenn die Fehlerschranke  $\varepsilon_{min}$  mehrmals hintereinander unterschritten wurde (z.B. fünf mal). Dies wird mit der Bedingung ergänzt, daß der relative Fehler in diesem Intervall monoton fallen soll, wenn er nur wenig unterhalb von  $\varepsilon_{min}$  liegt. Andernfalls wird keine Verlängerung der Schrittlänge vorgenommen.

Zur Steuerung der Schrittweite können zusätzlich auch Parameter verwendet werden, die aus der Gleichgewichtsiteration direkt ermittelt werden. Manche Autoren verwenden z.B. die

Anzahl der Iterationsschritte des vorhergehenden Inkrementes zur Bestimmung der neuen Inkrementlänge bei statischen Berechnungen [Bul93, Ram81]. Dies kann analog auch auf die Dynamik übertragen werden [Tem95]. Es wäre auch möglich, den dargestellten Fehler-schätzer mit dem Verfahren der Abschätzung des *Rayleigh-Quotienten* zu kombinieren. Dies erfordert jedoch eine schwer steuerbare Abstimmung der beiden unterschiedlichen Vorgehensweisen.

Die dargestellte adaptive Schrittweitensteuerung kann nur dann eingesetzt werden, wenn der aktuelle Zeitschritt mit einem konvergenten Iterationsprozeß beendet wird. Allerdings kann es bei stark nichtlinearem Verhalten (Stabilitätsversagen, Kontaktsimulation) vorkommen, daß der gewählte Zeitschritt zu groß ist, um Konvergenz bei der Gleichgewichtsiteration zu erreichen. In diesen Fällen muß der Iterationsvorgang abgebrochen und mit einem verkürzten Inkrement neu gestartet werden (siehe hierzu Abschnitt 4.5 sowie die ausführliche Zusammenstellung in [Bul93]).

Die mit diesem Vorgehen erhaltene Schrittweitenanpassung erfaßt nur Effekte, die aus dem Integrationsfehler der gesamten, globalen Struktur resultieren. Vorgänge, wie z.B. die Wellenausbreitung in einem eindimensionalen Kontinuum (2. Beispiel im Abschnitt 3.4.3) können nicht mit einer adaptiven Zeitschrittsteuerung erfaßt werden, da sich der Fehler zwar räumlich verändert, sich jedoch nicht im globalen Zeitintegrationsfehler bemerkbar macht.

Die Beispiele zur Wellenausbreitung in elastischen Körpern in [Li96, WEL99] zeigen diese Problematik deutlich: Aufgrund einer zu kurz gewählten Start-Schrittweite wird in der ersten Phase der Berechnung eine Anpassung vorgenommen. Im verbleibenden Simulationszeitraum bleibt sie nahezu konstant. Hier ist entweder ein ausreichend genaues uniformes Netz vorzugeben oder die Welle wird über eine sukzessiv veränderte Netzadaption möglichst genau erfaßt. Eine Schrittweitenanpassung ist jedoch unnötig.

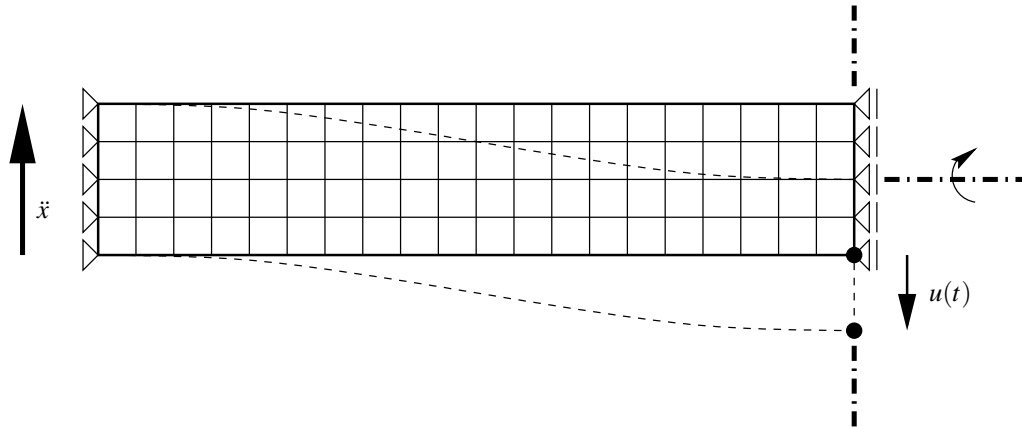
### 4.3.1 Beispiel: Gedämpft schwingender Stab

Das folgende Beispiel wurde in ähnlicher Form bereits von Li [Li96] und Riccius [Ric97] verwendet. Während Riccius den beidseitig eingespannten Balken mit einem uniformen Netz aus Scheibenelementen diskretisiert, nutzt Li zusätzlich eine Netzadaption. In der vorliegenden Arbeit wird ein runder Stab (Zylinder) mit linear elastischen Kontinuums-Elementen berechnet\*. Der beidseitig eingespannte Stab wird unter Ausnutzung der Symmetrie nur zur Hälfte diskretisiert (Abbildung 4.1).

Die feste Lagerung des Stabes (linke Seite) wird zu Beginn der Berechnung kurz mit Hilfe einer Fußpunkterregung beschleunigt ( $0 \leq t \leq 2.0s$ ). Anschließend schwingt der Stab in seiner niedrigsten Eigenfrequenz mit einer physikalischen Dämpfung (*Rayleigh-Dämpfung*:  $\alpha_R = 0.01, \beta_R = 0.001$ ). Die Berechnung wird mit einem konstanten Zeitschritt ( $\Delta t = 0.05$ ) sowie einem adaptiv gesteuerten Zeitschritt ( $0.05 \leq \Delta t \leq 0.2$ ) durchgeführt. Die Zeitintegration erfolgt mit dem *Generalized- $\alpha$*  Verfahren.

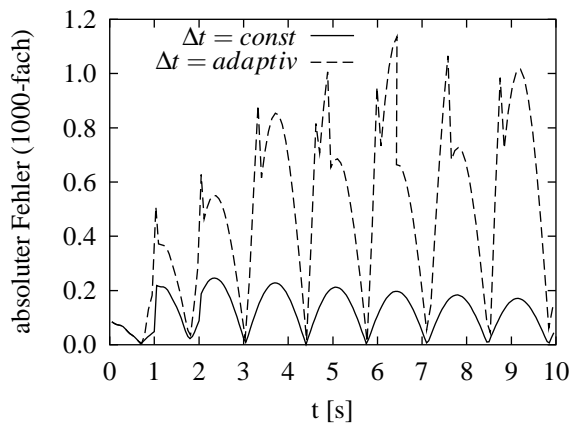
---

\*Element-Beschreibung: siehe Abschnitt 6.2

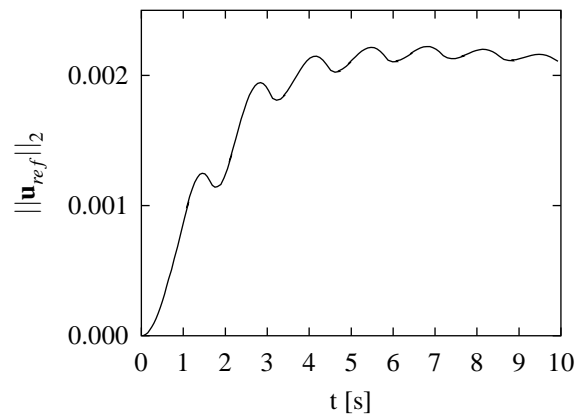


**Abb. 4.1:** Eingespannter Zylinder mit Fußpunktbeschleunigung (schematischer Schnitt)

Die von *Riccius* durchgeführten analytischen Untersuchungen am Einmassenschwinger zeigen, daß der Integrationsfehler bei konstanter Schrittweite am Umkehrpunkt der Verschiebung gegen Null geht und beim Nulldurchgang ( $u(t) = 0, \dot{u}(t) = \max$ ) maximal wird. Das hier untersuchte Beispiel verhält sich ähnlich, da der Stab nur in einer Frequenz schwingt. Der absolute Integrationsfehler zeigt diesen Zusammenhang deutlich (Abbildungen 4.2 und



**Abb. 4.2:** Absoluter Zeitintegrationsfehler



**Abb. 4.3:** Fehler-Bezugsgröße  $\|\mathbf{u}_{ref}\|_2$

4.4). Darüberhinaus nimmt mit zunehmender Dauer der Berechnung der Integrationsfehler (bei konstanter Integrationslänge) allmählich ab, da die Strukturantwort gedämpft wird und somit die maximale Verschiebungsamplitude kleiner wird.

Der Unterschied des Integrationsfehlers zwischen adaptiver und konstanter Schrittweitensteuerung ist abhängig von der jeweils verwendeten Schrittweite (Abbildungen 4.2 und 4.7). Dagegen ist die Fehler-Bezugsgröße  $\|\mathbf{u}_{ref}\|_2$  (Abbildung 4.3) nur von der Referenzschrittweite  $\Delta t_{ref}$  abhängig (Gleichung 4.15, hier:  $\Delta t_{ref} = \Delta t_{konst}$ ).

Wie bereits erwähnt, ist zu Beginn der Simulation die Bezugsgröße  $\|\mathbf{u}_{ref}\|_2$  zu klein, wodurch der relative Fehler zu große Werte ergibt. Deshalb wird die adaptive Schrittweitensteuerung erst nach einer vorgegebenen Zeitspanne aktiviert.

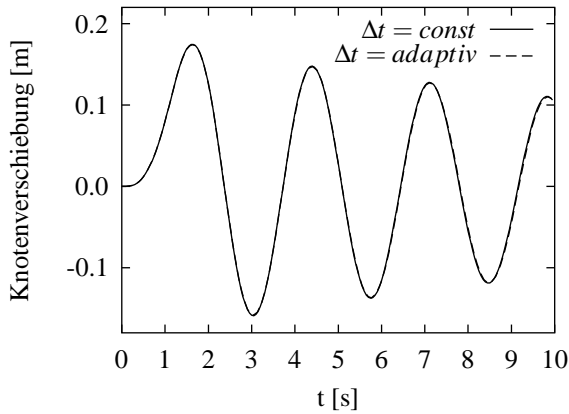


Abb. 4.4: Verschiebung Stabmitte

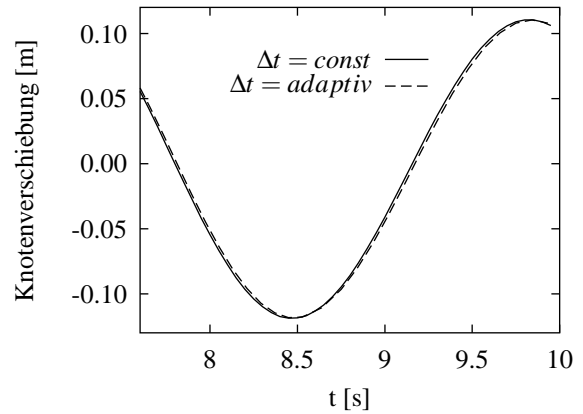


Abb. 4.5: Detail aus Abbildung 4.4

Der relative Fehler (Abbildung 4.6) nimmt bei einer konstanten Schrittweite analog dem absoluten Fehler sukzessive ab. Es wird also mit einer zunehmend höheren Genauigkeit gerechnet, wodurch ein unnötiger Rechenaufwand anfällt.

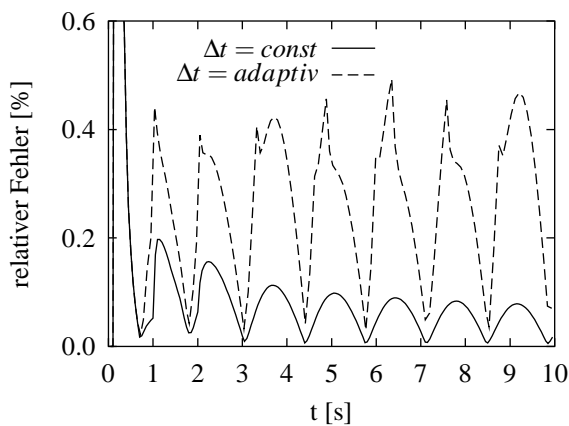


Abb. 4.6: Relativer Zeitintegrations-Fehler

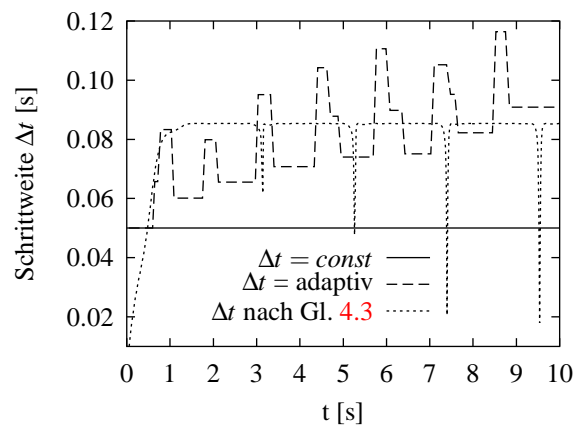


Abb. 4.7: Schrittweiten mit verschiedenen Steuerungen

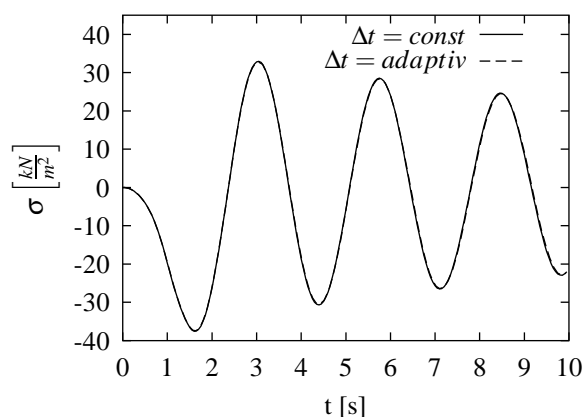
Mit der adaptiven Schrittweitensteuerung schwankt der Fehler in den vorgegebenen Grenzen und führt damit zu einer gleichmäßigen Genauigkeit während der gesamten Simulation. Legt man die obere und die untere Schranke enger zusammen, so wird das Genauigkeitsniveau zwar noch gleichmäßiger, die Anzahl der Schrittweitenanpassungen steigt jedoch stark an. Jede Schrittweitenänderung führt allerdings zu einem zusätzlichen Rechenaufwand, da jedesmal die effektive Steifigkeitsmatrix neu berechnet werden muß.

Die Länge der adaptiven Zeitschritte zeigt einen für das gewählte Beispiel typischen Verlauf (siehe Abbildung 4.7): Analog dem schwankenden relativen Fehler wird die Integrationslänge ständig verlängert oder verkürzt. Insgesamt ist aber auch ein steigender Trend erkennbar. Es wird also nicht nur die unterschiedliche Genauigkeitsanforderung der schwingenden Struktur in einer Periode erfaßt, sondern zusätzlich auch die abklingende Schwingung infolge der physikalischen Dämpfung mit einer allmählichen Verlängerung der Schrittweite automatisch berücksichtigt (siehe hierzu auch [Li96, Ric97]).

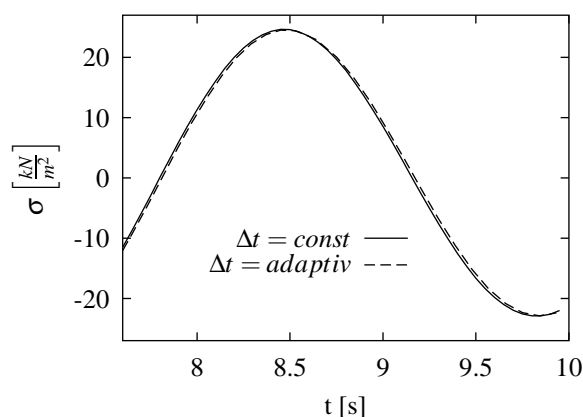


Im Gegensatz hierzu bleibt die Schrittweite nach Gleichung (4.3) (Abschätzung über *Rayleigh-Quotienten*) während der gedämpften Schwingung konstant (abgesehen von kurzen, numerisch bedingten Störungen). Die Berechnung mit konstantem Zeitschritt während einer Periode ist sicher vertretbar. Die fehlende Anpassung der Schrittweite an die degressive Schwingungsamplitude zeigt jedoch, daß der *Rayleigh-Quotient* nur bedingt für eine adaptive Schrittweitensteuerung geeignet ist.

Die Abbildungen 4.4 und 4.8 zeigen deutlich, daß mit der adaptiv gesteuerten Berechnung ausreichend genaue Ergebnisse erzielt werden können. Der numerische Aufwand kann bei diesem Beispiel um ca. 40% verringert werden.



**Abb. 4.8:** Biegespannung an der Einspannung



**Abb. 4.9:** Detail aus Abbildung 4.8

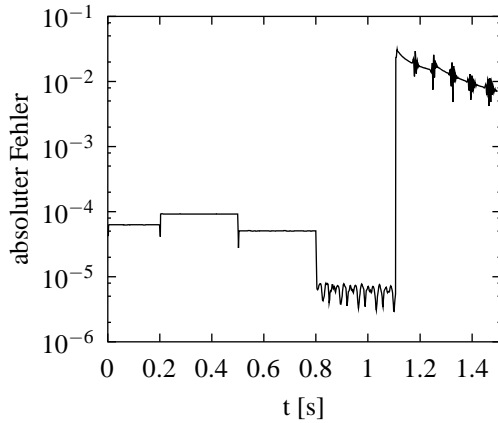
Die adaptive Schrittweitensteuerung ist deshalb besonders für Finite-Element-Berechnungen großer Systeme sinnvoll. Der zusätzliche Aufwand zur Ermittlung des Integrationsfehlers sowie zur Schrittweitenänderung ist im Vergleich zum numerischen Aufwand bei konstanter Integrationslänge vernachlässigbar.

## 4.4 Zusatzmaßnahmen für Kontaktsimulation

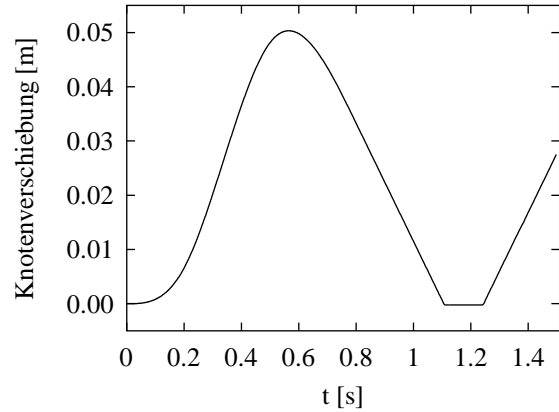
Bei der Kontaktsimulation werden in den Elementen große Spannungsgradienten wirksam, die mit relativ kleinen Verformungsgradienten einhergehen. Es ist deshalb zunächst fraglich, ob mit dem verwendeten Fehlerschätzer (Gleichung 4.6) die Schrittweite so gesteuert wird, daß auch die Spannungen möglichst genau abgebildet werden. Am Beispiel “elastischer Stab prallt auf starre Lagerung” (siehe Kapitel 3.5.1) soll diese Problematik untersucht und eine allgemeine Verbesserung der Schrittweitensteuerung für diesen Sonderfall gefunden werden.

Der Fehlerschätzer nach *Zienkiewicz* und *Xie* (Gleichung 4.6) spiegelt jede Phase der Simulation wider (Abbildungen 4.10 und 4.11): Die drei Phasen Beschleunigen-Abbremsen-Beschleunigen zu Beginn der Berechnung ( $t_0 = 0$  bis  $t = 0.8s$ ) werden jeweils mit einem relativ kleinen Fehler angezeigt. Die gewählte Zeitschrittlänge\* ist also im Vergleich zur ab-





**Abb. 4.10:** Absoluter Zeitintegrations-Fehler  $\varepsilon_{ZX}$  ( $\Delta t = \Delta t_{opt}$ )



**Abb. 4.11:** Knotenverschiebung am Stab (Kontaktstelle)

zubildenden Strukturantwort sehr klein gewählt. Auch die Phase der gleichförmigen Bewegung ( $t = 0.8$  bis  $t \approx 1.1$  s) geht mit einem sehr kleinen Fehler einher. Erst beim Aufprall des Stabes auf die Lagerung ( $t \approx 1.1$  s) vergrößert sich der Fehler um den Faktor 1000.

Beim Abstoß des Stabes von der Lagerung wird keine Erhöhung des Fehlers angezeigt. Allerdings haben viele Beispielrechnungen gezeigt, daß die (Kontakt-)Spannungen nur dann mit großer Genauigkeit abgebildet werden, wenn auch beim Abstoß der Zeitschritt verkürzt wird. Der Fehlerindikator kann je nach gewählter Referenzschrittweite  $\Delta t_{ref}$  jedoch bereits wieder eine größere Schrittweite zulassen. Die Aussage des Fehlerschätzers ist also in diesem engen Zeitbereich problematisch [Arm99].

Die Systemsteifigkeit ist dagegen sowohl für den Aufprall als auch für den Abstoß ein zuverlässiger, charakteristischer Parameter, da sie sich hier jeweils um den Wert 1.0 zwischen frei fliegendem Stab und Kontakt ändert. Deshalb wird der bereits dargestellte Steifigkeitsparameter (Gleichung 4.2, Seite 47) als zusätzlicher Indikator einbezogen. Hier wird nicht der absolute Wert der Systemsteifigkeit wie in [WGS85] verwendet, sondern dessen Änderung im aktuellen Zeitschritt:

$$\Delta s = |{}^{t+\Delta t}s - {}^t s|, \quad \forall \Delta t \text{ so, daß } \Delta s \leq \Delta s_{max} \quad (4.18)$$

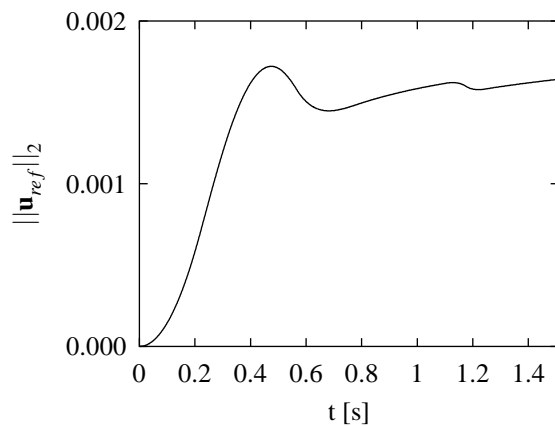
Die maximal zulässige Änderung  $\Delta s_{max}$  der Steifigkeit wird auf einen vorgegebenen Wert (z.B.  $\Delta s_{max} = 0.2$ ) begrenzt. Wird dieser Wert überschritten, dann muß der letzte Zeitschritt mit einer verkürzten Schrittweite wiederholt werden. Zur Abschätzung der neuen Integrationslänge kann die Änderung der Systemsteifigkeit dienen.

Beim bereits dargestellten Beispiel springt im Idealfall die Differenz der Steifigkeit um den Wert 1.0. Es ist deshalb a priori eine minimale Schrittweite vorzugeben, bei der die Bedingung der Gleichung (4.18) nicht erfüllt sein muß. Die maximal zulässige Änderung von  $\Delta s_{max}$  gewährleistet bei vielen anderen Kontaktsimulationen, daß der eigentliche Kontakt- oder Abhebevorgang mit ausreichend kleinen Zeitschritten erfaßt wird.

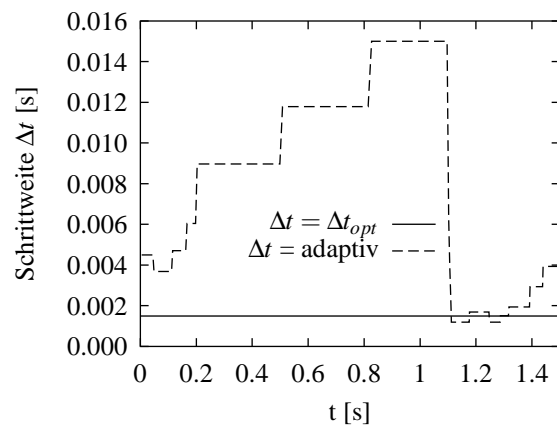
\*  $\Delta t = \Delta t_{opt} = 0.015$  s nach Gleichung (3.31), Seite 33

Als Alternative zum Steifigkeitsparameter könnte hier auch die maximale Anzahl von Integrationspunkten festgelegt werden, die in einem Zeitschritt ihren Deformationsmode ändern (z.B.: Übergang Kontaktmode in Separationsmode). Bei der dynamischen Berechnung von verschmierten Rißmodellen (vgl. [dBM95]) kann analog vorgegangen werden, wenn die maximale Anzahl von neu gerissenen Integrationspunkten in einem Zeitschritt begrenzt wird.

Obwohl der Aufprall das gesamte System stark verändert, bleibt der ermittelte Referenzwert der Verschiebung  $\|\mathbf{u}_{ref}\|_2$  (Gleichung 4.15) nach der “Initialisierungsphase” auf einem nahezu konstanten Niveau (Abbildung 4.12). Er ist deshalb auch für die Kontaktsimulation optimal geeignet.



**Abb. 4.12:** Fehler-Bezugsgröße  $\|\mathbf{u}_{ref}\|_2$   
( $\Delta t = \Delta t_{opt} = \Delta t_{ref}$ )



**Abb. 4.13:** Zeitschrittweite bei konstanter und adaptiver Schrittweite

Der hier dargestellte Algorithmus zur adaptiven Zeitschrittsteuerung liefert automatisch eine problemangepaßte Schrittweite (Abbildung 4.13). Für die gesamte Simulation (Abheben – Beschleunigung – Umkehrpunkt – Gleichförmige Bewegung – Aufprall – Kontakt – Abheben – Gleichförmige Bewegung, siehe Abbildung 4.11) kann mit der adaptiven Schrittweitensteuerung ca. 70 % der Rechenzeit im Vergleich zu einer Berechnung mit konstanter Schrittweite eingespart werden.

Anhand der Kontaktspannung (Abbildung 4.14) kann gezeigt werden, daß die Genauigkeit der adaptiven sowie der Berechnung mit konstanter Schrittweite sehr ähnlich ist. Hier zeigt sich auch nochmals die Notwendigkeit, daß der Steifigkeitsparameter in die Schrittweitensteuerung miteinbezogen wird: Ohne ihn wäre die Ausrundung am Ende der Kontaktzeit (Abbildung 4.15) noch stärker, da der Fehlerschätzer hier bereits wieder eine größere Schrittweite zulassen würde.

Die bereits erwähnten Oszillationen im Stab nach dem Abstoß von der Lagerung treten nahezu unverändert bei der adaptiven Zeitintegration auf (Abbildung 4.16). Hier ist also keine Verbesserung festzustellen. Allerdings wird insgesamt deutlich, daß es nicht zwingend erforderlich ist, mit der nach [Wri81] (Gleichung 3.31) ermittelten optimalen Schrittweite zu rechnen, sondern daß eine adaptive Zeitintegration ebenso geeignet ist.

Die adaptive Steuerung sollte jedoch nicht mit der Abschätzung des *Rayleigh-Quotienten* nach Gleichung (4.3) erfolgen, da hierdurch keine befriedigenden Ergebnisse erzielt werden

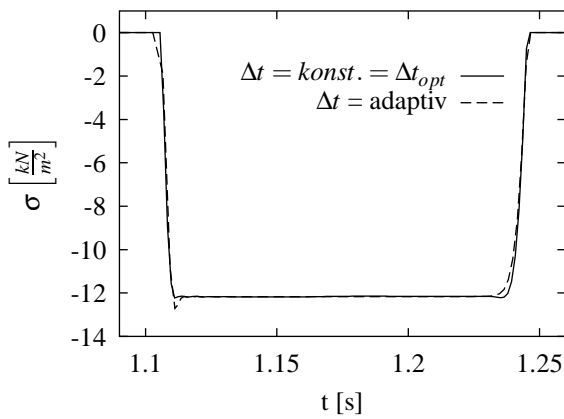


Abb. 4.14: Kontaktspannung mit konstanter und adaptiver Schrittweite

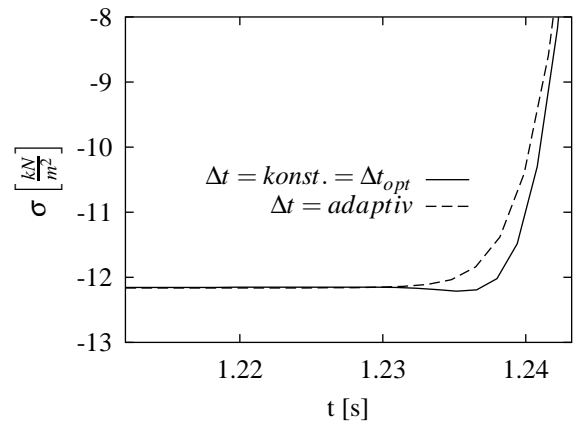


Abb. 4.15: Detail aus Abb. 4.14

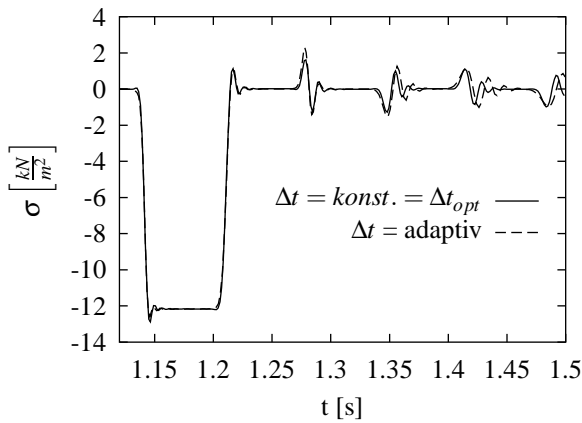


Abb. 4.16: Längsspannung im Stab (konst. und adaptive Schrittweite)

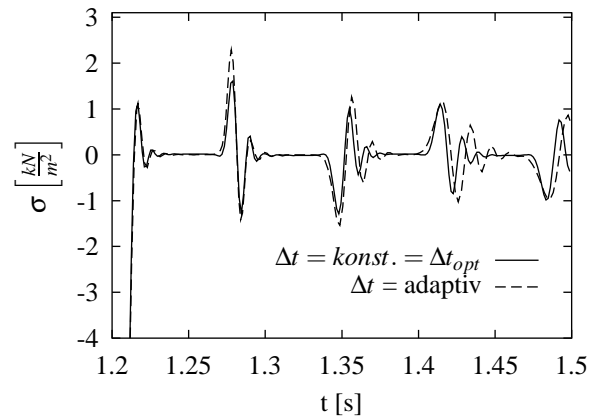


Abb. 4.17: Detail aus Abb. 4.16

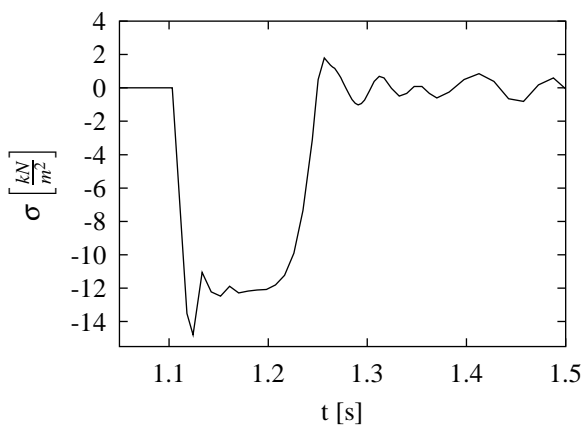


Abb. 4.18: Längsspannung im Stab mit Steuerung nach Gleichung (4.3)

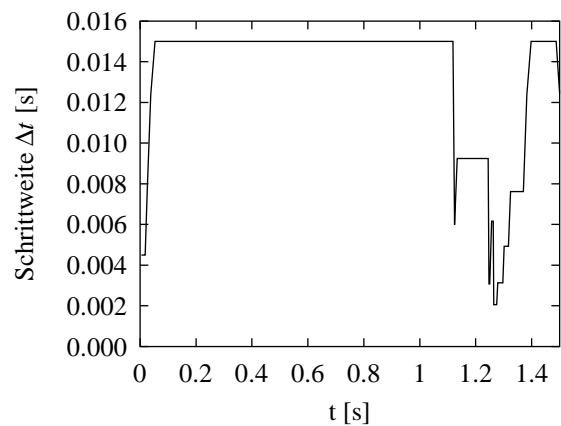


Abb. 4.19: Adaptive Schrittweite nach Gleichung (4.3) (Abschätzung über Rayleigh-Quotienten)

können (Abbildung 4.18). Die hier gezeigten Ergebnisse werden sogar noch wesentlich ungenauer, wenn die maximal zulässige Schrittweite vergrößert wird.

Die adaptive Schrittweitensteuerung ist also auch für die Kontaktsimulation geeignet, wenn die hier gezeigten Zusatzmaßnahmen verwendet werden. Es wird nicht nur der numerische Aufwand drastisch verringert, sondern auch die Genauigkeit der Berechnung wesentlich verbessert.

## 4.5 Begleitende Maßnahmen

Bei sprunghaften Steifigkeitsänderungen während der Berechnung können darüberhinaus zusätzliche Maßnahmen notwendig werden, damit die Gleichgewichtsiteration stabil gehalten werden kann [Ram81, RW99]. Der Iterationsprozeß des aktuellen Zeitschrittes kann nur sehr langsam konvergieren oder sogar divergieren. Während bei langsamer Konvergenz nur unnötig viel Rechenzeit verloren geht, führt divergentes Verhalten zum Abbruch der gesamten Berechnung und muß deshalb unbedingt vermieden werden.

Beide Probleme können gelöst werden, wenn während der Iteration mit Hilfe geeigneter Parameter entschieden wird, ob  $\Delta t$  zu groß gewählt wurde und deshalb der aktuelle Schritt mit einem verkleinerten Inkrement neu gestartet werden muß.

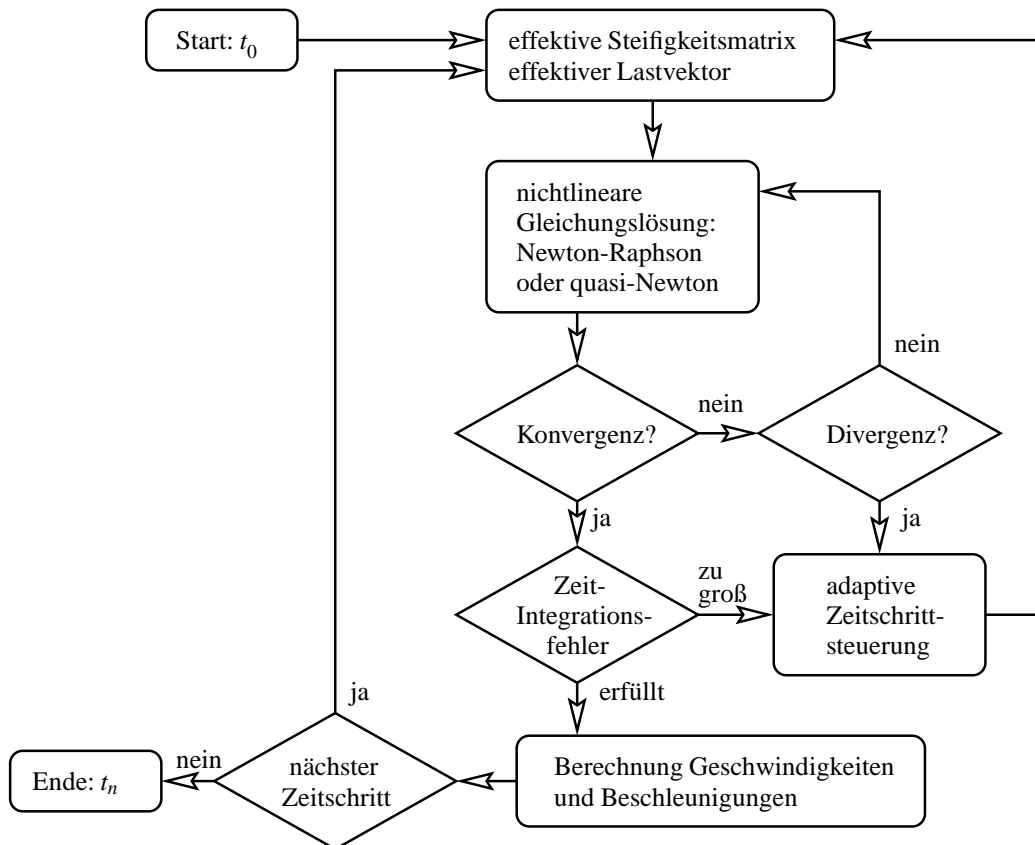


Abb. 4.20: Schema der erweiterten Zeitintegration

Bei der im Rahmen der vorliegenden Arbeit verwendeten Software wird die Anzahl der Iterationen je Zeitschritt begrenzt. Außerdem wird anhand steigender oder oszillierender Konvergenzkriterien (Abschnitt 2.3.3) geprüft, ob die Iteration konvergiert. Ist entweder die Anzahl der maximal zulässigen Iterationen überschritten oder divergentes Verhalten festzustellen, wird der aktuelle Zeitschritt halbiert. Ähnliche Methoden sind in [Bat86, Bul93, Cri91, Sch89] zu finden.

Abbildung 4.20 zeigt schematisch den Ablauf der Zeitintegration und der adaptiven Schrittweitensteuerung sowie die begleitenden Maßnahmen. Dieses Ablaufdiagramm gilt jedoch nur für eine Gleichgewichtsiteration, bei der die nichtlinearen Anteile über Pseudo-Lasten erfaßt werden. Andernfalls muß in jedem Iterationsschritt eine neue effektive Steifigkeitsmatrix ermittelt werden (vgl. Abschnitt 3.3, Seite 22).

Neben einer numerisch robusten Berechnung ergibt sich durch die hier gezeigten Zusatzmaßnahmen auch ein positiver Nebeneffekt: Statt vieler Iterationen bei einem zu groß gewählten  $\Delta t$  benötigt man mit kleineren Zeitschritten insgesamt weniger Iterationen, da in diesen Fällen deren Anzahl überproportional abnimmt. Gerade bei Systemen mit sehr vielen Freiheitsgraden und hochgradig nichtlinearem Strukturverhalten ist dies ein wesentlicher Aspekt zur Verringerung des Rechenaufwandes.

## 4.6 Numerische Simulation: Schalenbeulen

Mit dem folgenden Beispiel wird die Leistungsfähigkeit der adaptiven Schrittweitensteuerung bei der Simulation eines numerisch sehr empfindlichen Systems demonstriert. Die Abmessungen und die Art der Belastung (siehe Abbildung 4.21) sind in ähnlicher Form auch in der Literatur zu finden [Din88, Kuh96].

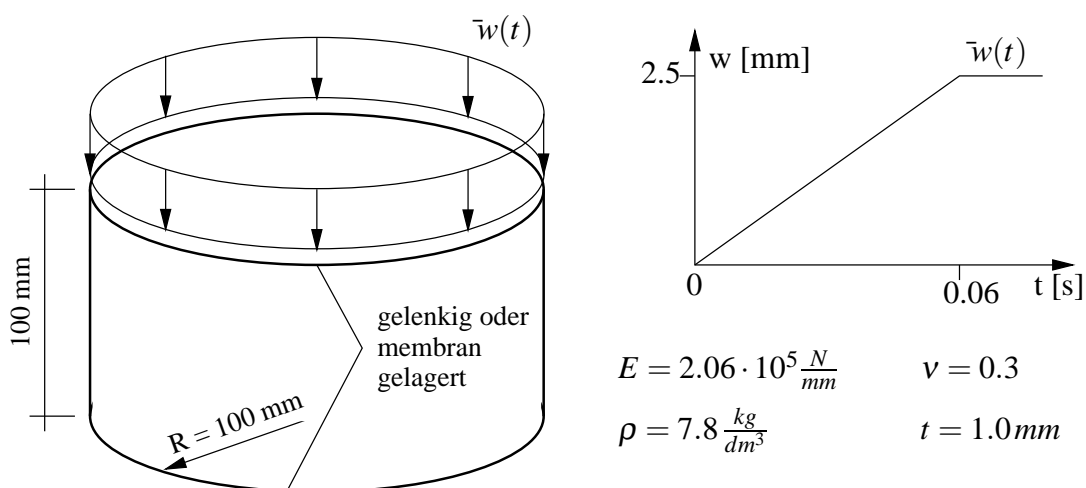


Abb. 4.21: Kreiszylinderschale im weggesteuerten Belastungsversuch

Die ideale Beullast der dargestellten Zylinderschale läßt sich unter der Annahme einer per-

fechten Geometrie und linear-elastischem Materialverhalten mit

$$p_{krit} = \frac{Et^2}{\sqrt{3(1-\nu^2)}R} \quad (4.19)$$

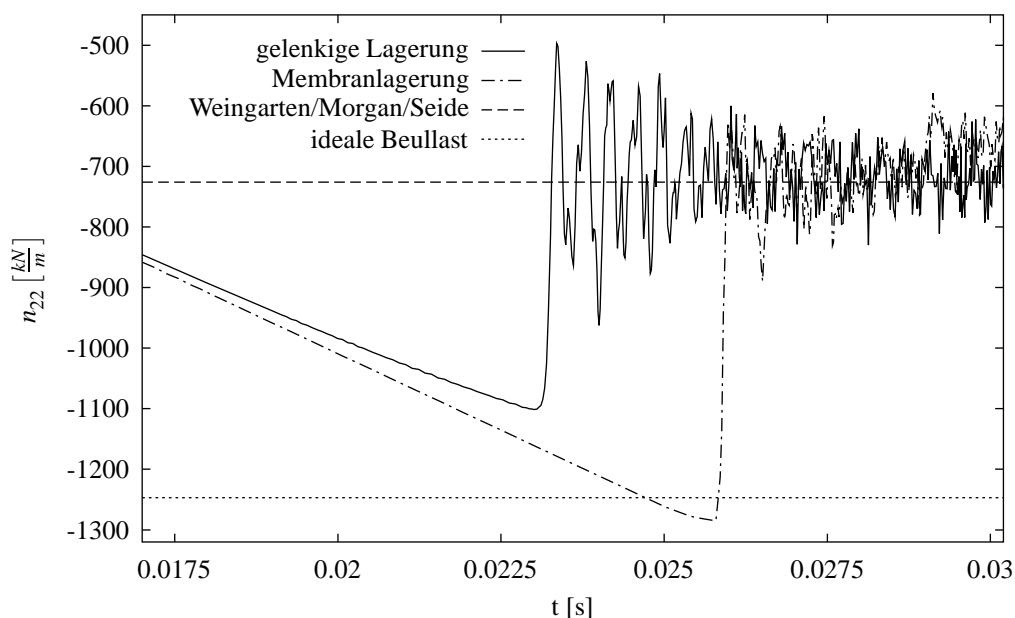
berechnen. Allerdings werden in Versuchen wesentlich niedrigere Werte ermittelt. Hier spielt vor allem die Imperfektionssensitivität der Schale eine wichtige Rolle. Nach Weingarten, Morgan und Seide\* [WMS65] kann die Traglast mit

$$p_{WMS} = p_{krit} \left(1 - 0.902 \left(1 - e^{-\frac{1}{16} \sqrt{\frac{R}{t}}}\right)\right) \quad (4.20)$$

abgeschätzt werden. Für das verwendete Beispiel ergibt sich die ideale Beullast  $p_{krit}$  des perfekten Zylinders nach Gleichung (4.19) zu  $1247 \frac{kN}{m}$  und mit der Abschätzung nach Gleichung (4.20) erhält man  $p_{WMS}$  zu  $726 \frac{kN}{m}$ .

Das System wird mit den in Abschnitt 2.2.3 erläuterten Elementen räumlich diskretisiert. Für die dynamische Berechnung wird das *Generalized- $\alpha$*  Verfahren mit der vorgestellten adaptiven Schrittweitensteuerung verwendet. Zu Beginn der Simulation wird das System kurzzeitig mit Hilfe einer Impulslast gestört, damit eine geringe Imperfektion im Zylinder vorhanden ist. Anschließend erfolgt eine quasi-weggesteuerte axialsymmetrische Belastung wie in Abbildung 4.21 skizziert.

In Abbildung 4.22 sind die Werte der Gleichungen (4.19) und (4.20) sowie die Ergebnisse von zwei Zeitverlaufsberechnungen für Zylinder unterschiedlicher Lagerungsbedingungen dargestellt. Die Ergebnisse zeigen, daß die ideale Beullast bei gelenkiger Lagerung nicht



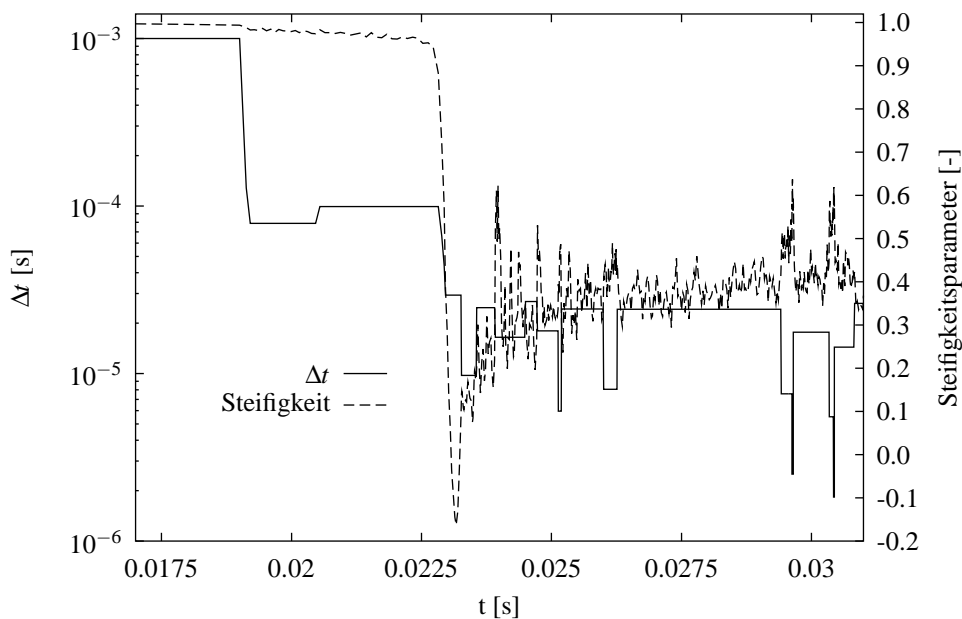
**Abb. 4.22:** Mittlere Meridiankraft in der Zylinderschale

erreicht wird, da das System bereits auf einem niedrigeren Niveau plötzlich in einen hochgradig dynamischen Zustand übergeht. Bei der Membranlagerung wird sogar ein geringfügig

\*Auswertung experimenteller Vergleichsdaten

höheres Niveau als die ideale Beullast erreicht. Dies ist auf die stabilisierende Wirkung der Trägheitskräfte zurückzuführen. Bei beiden Zeitverlaufskurven fällt auf, daß die mittlere Meridiankraft im Zylinder schließlich um die von *Weingarten/Morgan/Seide* ermittelte Beullast schwingt. Allerdings ist es fraglich, ob dies auch tatsächlich der niedrigsten Beullast entspricht [Dem97]. Diese Fragestellung wird hier jedoch nicht weiter verfolgt.

Die numerische Simulation erfordert neben einem robusten Zeitintegrationsverfahren vor allem im Bereich des Beulvorgangs sehr kleine Schrittweiten, damit dieser Vorgang genau erfaßt werden kann. Deshalb ist eine adaptive Schrittweitensteuerung von großem Vorteil, da der Vorbeulbereich mit wesentlich größeren Integrationslängen zu erfassen ist. In Abbildung 4.23 ist dieser Zusammenhang gut zu erkennen. Bei dieser Art von Simulation ist wiederum



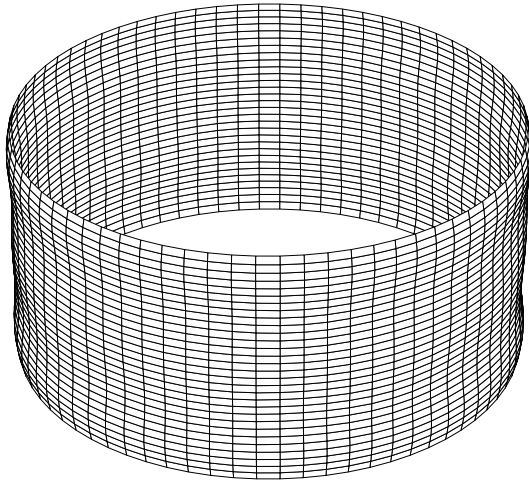
**Abb. 4.23:** Schrittweiten und Steifigkeitsparameter (gelenkige Lagerung)

das Steifigkeitskriterium sehr hilfreich, da der plötzliche Steifigkeitsverlust zu sehr kurzen Schrittweiten führt, während das Fehlerkriterium noch relativ lange Schritte zuläßt.

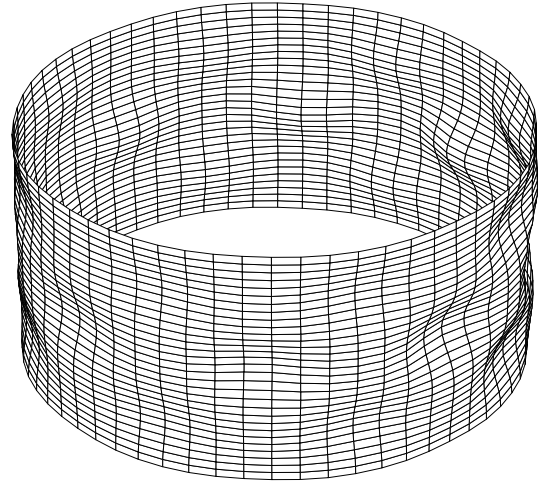
Im weiteren Verlauf der Simulation oszilliert der Steifigkeitsparameter im Bereich zwischen 0.3 und 0.6. Die Schrittweitensteuerung in diesem von hochfrequenten Schwingungen geprägten Bereich übernimmt im wesentlichen der Fehlerindikator sowie die im Abschnitt 4.5 beschriebenen begleitenden Maßnahmen.

Die Abbildungen 4.24, 4.26 und 4.28 zeigen die Verformungen des gelenkig gelagerten Zylinders während des Beulvorgangs. Offensichtlich führt die behinderte Radialverschiebung zu einer zusätzlichen Imperfektion an den Rändern. Somit ist auch die niedrigere Beullast zu erklären.

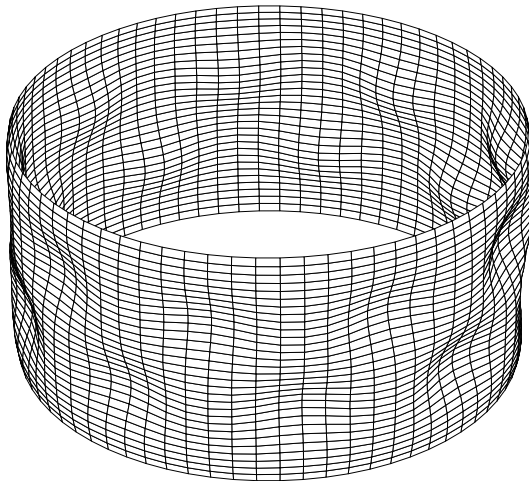
Das Beulmuster des membran gelagerten Zylinders ist zu Beginn des Beulvorgangs ähnlich dem des gelenkig gelagerten Zylinders (Abbildung 4.25 und 4.27). Allerdings geht bei diesem Zylinder das Beulmuster sehr schnell in einen Zustand mit weniger Wellen in Umfangs- und Meridianrichtung über.



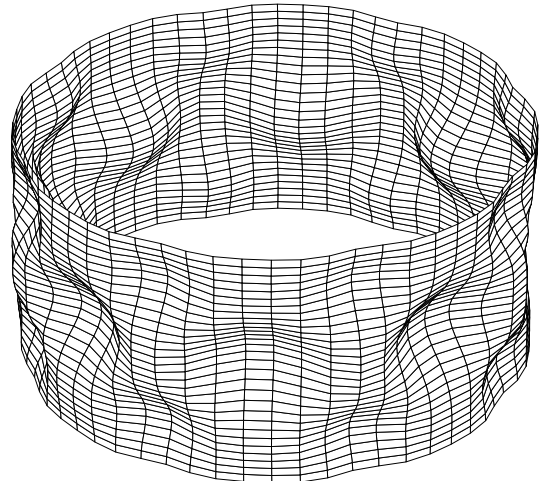
**Abb. 4.24:** gel. Lagerung  $t = 0.02299 s$



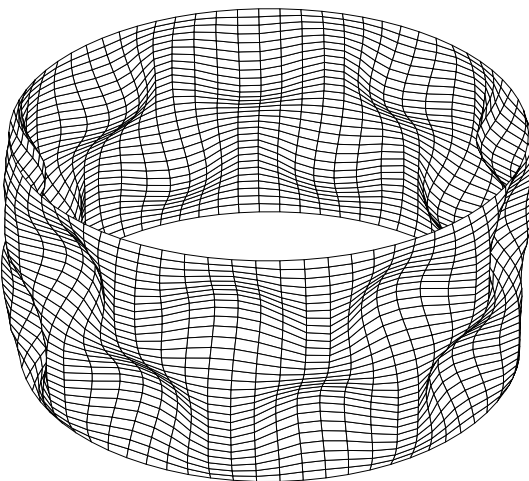
**Abb. 4.25:** Membranlag.  $t = 0.02588 s$



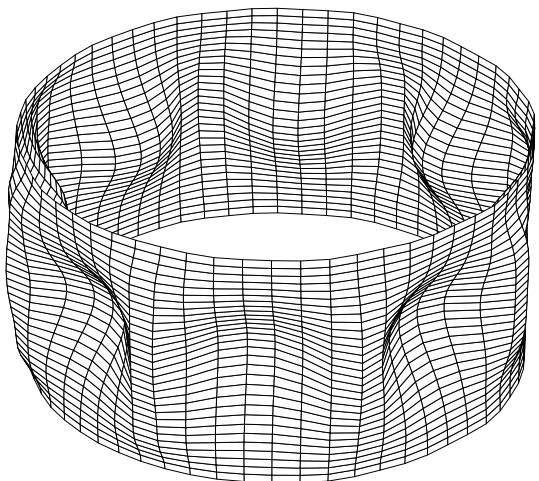
**Abb. 4.26:** gel. Lagerung  $t = 0.02323 s$



**Abb. 4.27:** Membranlag.  $t = 0.02600 s$



**Abb. 4.28:** gel. Lagerung  $t = 0.03269 s$



**Abb. 4.29:** Membranlag.  $t = 0.02614 s$



# Kapitel 5

## Parallele Finite-Element-Berechnung

### 5.1 Allgemeines

Die *Finite-Element-Methode* ist eng mit der Entwicklung leistungsfähiger Rechenanlagen verbunden. Seit den Anfängen der Computerberechnung sind bis heute zwei zentrale Aufgaben geblieben: Die Minimierung der Rechenzeit sowie die Maximierung des verarbeitbaren Problemumfangs. Ähnliche Anforderungen an numerische Simulationen ergeben sich auch in anderen Wissenschaftszweigen:

- In der Klimaforschung und Wettervorhersage arbeitet man zunehmend mit dreidimensionalen Modellen, bei denen neben der Atmosphäre z.B. auch Ozeane miteinbezogen werden. Die Genauigkeit und Zuverlässigkeit dieser Simulationen hängt neben der Kalibrierung des Modells durch gemessene Klimadaten vor allem von der räumlichen und zeitlichen Diskretisierung ab. Der enorme Rechenaufwand wird heute überwiegend durch parallele *Supercomputer* geleistet [Kli97].
- Viele Wissenschaftler arbeiten in Spezialbereichen der theoretischen und angewandten Physik mit numerischen Berechnungen um z.B. den Aufbau der Materie, die Eigenschaften neuer Atomreaktoren oder Abläufe in Sternen (Astrophysik) mit Hilfe von besonders leistungsfähigen Computern zu simulieren [Jül99].
- Untersuchungen zum Strömungsverhalten von Gasen und Flüssigkeiten werden heute meist mit Hilfe der *numerischen Strömungsmechanik (Computational Fluid Dynamics - CFD)* durchgeführt. Aufgrund der sehr hohen Anzahl an Freiheitsgraden und der aufwendigen mathematischen Formulierung ist der Rechenaufwand oft nur mit Hochleistungsrechenanlagen zu bewältigen [LMR<sup>+</sup>96, WBD96].

Analog dem nahezu exponentiellen Wachstum der zur Verfügung stehenden Rechnerkapazitäten wachsen also auch die Anforderungen der numerischen Simulationen. Die parallele Nutzung der aktuell zur Verfügung stehenden Technik ist deshalb kein "Mode-Trend", sondern wird sich dauerhaft als Hilfswissenschaft etablieren, da dies zu einer enormen Leistungssteigerung führen kann.

Insbesondere im Bereich der Informatik werden hierzu Algorithmen entwickelt, mit denen die parallele Verwendung von Prozessoren auf Betriebssystemebene gewährleistet wird. Zudem werden vernetzte Computer mit komfortablen Programmbibliotheken für eine parallele Anwendung erschlossen [G<sup>+</sup>94, Har96].

## 5.2 Grundlagen der Parallelen Berechnung

Die maximale Rechenleistung eines einzelnen Prozessors hängt von vielen Parametern ab. Neben der eigentlichen Prozessorarchitektur (*CISC*, *RISC*, *VLIW*, usw. [Kar93]) geht vor allem die Taktfrequenz der verschiedenen Arbeitsschritte sowie die Größe des zur Verfügung stehenden Rechenspeichers in die Beurteilung der Leistungsfähigkeit ein. Die Geschwindigkeit eines einzelnen Prozessors wird im wesentlichen von der Signallaufzeit zwischen verschiedenen elektronischen Bauteilen begrenzt. Eine Erhöhung der Taktfrequenz kann damit nur erfolgen, wenn die technisch realisierbare Stufe der Mikrotechnik dies zuläßt.

Es ist deshalb naheliegend, die zur Verfügung stehende Technik parallel nebeneinander einzusetzen und die zu bearbeitende Aufgabe in entsprechende Arbeitsschritte aufzuteilen. Aus der Art bzw. dem Umfang dieser Aufteilung lassen sich verschiedene Rechnersysteme bzw. -Architekturen definieren [BDF<sup>+</sup>93].

### 5.2.1 Rechnersysteme – Architekturen

Eine allgemein genutzte Beschreibung von Rechnern beruht auf dem *von-Neumann-Rechner*\*, der aus einem Leitwerk für die Steuerung, einem Rechenwerk für die Ausführung der Operationen, einem Speicherwerk für die Speicherung von Programmen und Daten, einem Ein- und Ausgabewerk für die Ansteuerung von Peripheriegeräten sowie einem gemeinsamen Datenbus für die Übertragung von Daten und Befehlen besteht. Darauf aufbauend läßt sich nach Flynn [Mär94] eine weit verbreitete Rechnerklassifikation vornehmen:

- Der **SISD-Rechner** (*Single Instruction Stream, Single Data Stream*) stellt den *von-Neumann-Rechner* dar. Es wird also je Zeiteinheit nur eine Anweisung auf eine Dateneinheit ausgeführt. Bekannte Vertreter dieser Gruppe sind Computersysteme mit Mikroprozessoren (*Personal Computer, PC*). Aber auch *Vektor-* und *Pipeliner*rechner werden hier eingruppiert.
- Der **SIMD-Rechner** (*Single Instruction Stream, Multiple Data Stream*) verfügt über mehrere Rechenwerke, die von einem Leitwerk gesteuert werden. Jedem Rechenwerk ist ein Speicherbereich zugeordnet. Ein Vertreter dieser Gruppe ist der *Feldrechner*, eine selten genutzte Bauart. Manche Autoren ordnen auch *Vektorrechner* in diese Klasse ein [Mär94].
- Der **MISD-Rechner** (*Multiple Instruction Stream, Single Data Stream*) besitzt mehrere Leitwerke, die Befehle auf einem Datenstrom ausführen, der von Rechenwerk zu Rechenwerk geleitet wird. Eine praktische Umsetzung dieser Kategorie ist schwer vorstellbar.
- Der **MIMD-Rechner** (*Multiple Instruction Stream, Multiple Data Stream*) besteht aus mehreren *von-Neumann-Rechnern*, die entweder mit einem gemeinsamen Speicher verbunden sind oder über ein Verbindungsnetzwerk kommunizieren.

---

\*nach John von Neumann, 1903-1957

Weitergehende Klassifikationen bieten das *Erlanger Klassifikationssystem* (*Erlangen Classification System (ECS)*) sowie die Klassifikation nach Shore [Läm97a, Mär94].

In jedem modernen Computersystem werden Daten parallel abgearbeitet. Grundlegende Unterschiede gibt es jedoch in der Komplexität der Parallelverarbeitung bzw. in welchen Ebenen diese Parallelität vorkommt [Hei94, Kar93]:

- **Bit-Ebene:** Die *Wortlänge* kennzeichnet die Größe der Dateneinheiten, die in einem Bearbeitungsschritt abgearbeitet werden können. Derzeit sind dies meist 32 bit (z.B.: *Intel Pentium Prozessoren*©) oder 64 bit (z.B.: *DEC-Alpha Prozessoren*©).
- **Funktionseinheiten-Ebene:** Oft werden in heutigen *CPU's* (*Central Processing Unit*) mehrere Prozessoren parallel für verschiedene Aufgaben\* verwendet. Jeden dieser Prozessoren bezeichnet man als Funktionseinheit. Darüberhinaus können mit Hilfe des *Pipelining* mehrere Befehle gleichzeitig in verschiedenen Bearbeitungsstadien in der CPU gehalten werden.
- **Daten-Ebene:** Werden bestimmte Datenbereiche (z.B.: Vektoren) zur Bearbeitung auf verschiedene CPU's verteilt, so spricht man von einer Datenparallelität. Es ist naheliegend, daß diese Art der Parallelisierung einen speziellen Programmcode erfordert. Hierbei wird festgelegt, welche Teile parallel abgearbeitet werden können. Dies wird entweder vom Programmierer oder von speziellen *Compilern* vorgegeben. Bekannte Vertreter dieser Gruppe sind die *Vektorrechner* (z.B.: *Cray1*©). Darüberhinaus gehören die *SIMD-Rechner* in diese Gruppe.
- **Befehls-Ebene:** Hier können unterschiedliche Befehle auf verschiedene Datenbereiche angewendet werden. Dies setzt voraus, daß mehrere *CPU's* vorhanden sind, die untereinander Daten austauschen können. Die *VLIW-Systeme* (*Very Long Instruction Word*) stellen hierbei eine besondere Form dar, da sie nur über ein gemeinsames Steuerwerk verfügen. Von allgemeinerem Interesse sind deshalb die *MIMD-Parallelrechner*.

Während die Parallelitäten der Bit-Ebene sowie der Funktionseinheiten-Ebene von der vorhandenen Prozessor-Architektur abhängen und vom Programmierer nicht wesentlich beeinflusst werden können, erfordert die Parallelisierung auf Daten-Ebene, insbesondere auf Befehls-Ebene spezielle programmtechnische Umsetzungen. Im Rahmen der vorliegenden Arbeit werden ausschließlich *MIMD-Parallelrechner* behandelt (Parallelisierung auf Befehls-Ebene). Diese lassen sich wiederum vereinfachend in zwei Gruppen einteilen:

- Systeme, bei denen alle Prozessoren auf einen **gemeinsamen Speicher** gleichberechtigt zugreifen, werden als *shared memory* Systeme bezeichnet. Der gemeinsame Speicher dient nicht nur als Programmspeicher, sondern zugleich auch als Kommunikationseinheit. Diese Rechanlagen sind üblicherweise auf wenige Prozessoren beschränkt und erfordern eine spezielle Steuerung zur Speicherzugriffs-Verwaltung. Ein bekannter Vertreter dieser relativ kleinen und sehr teuren Gruppe ist die *Cray Y-MP*© [BDF+93].

\*z.B.: Gleitkommaberechnung, Integer-Arithmetik, Ein-/Ausgabe

- Die weiteren Ausführungen beziehen sich jedoch auf die zweite, wesentlich häufiger genutzte Gruppe. Diese arbeitet mit einem Verbindungsnetzwerk sowie einem **verteilten Speicher** (*distributed memory*). Damit verfügt jede Rechen-Einheit über einen lokalen Speicherbereich. Der Datenaustausch zwischen den Rechereinheiten erfolgt über das Verbindungsnetzwerk, das je nach Rechner-Architektur über eine feste Topologie\* mit einer sehr hohen Leistungsfähigkeit oder über eine LAN-Kopplung (*Local Area Network*) bzw. WAN-Kopplung (*Wide Area Network*) verfügt.

Der Hauptvorteil der Parallelrechner mit verteiltem Speicher liegt derzeit in ihrer relativ kostengünstigen Herstellung, da hierzu überwiegend Komponenten der Großserienfertigung verwendet werden können.

Eine besondere Form dieser Bauweise stellen sog. *Workstation-Cluster* dar [BDF<sup>+</sup>93]. Hierzu werden meist *UNIX-Workstations*<sup>†</sup> oder sehr preiswerte *Personal Computer* verwendet. Mit Hilfe spezieller Software (siehe folgenden Abschnitt 5.2.2) ist es sogar möglich, ein Netzwerk aus heterogenen Rechnern zu einem Parallelrechner zusammenzuschließen [BR99, Thi95]. Das weltweite *Internet* bietet sogar die Möglichkeit, Rechenanlagen an beliebigen Standorten miteinander zu verbinden.

Hier wird jedoch auch ein wesentliches Problem dieser Art von Parallelrechnern deutlich: Die eingeschränkte Bandbreite der Netzwerkverbindung stellt bei Programmen mit einem hohen Datentransfer eine starke Leistungseinschränkung dar. Deshalb beschränkt sich die Anwendung dieser Rechnerkonfiguration auf Algorithmen, die einen relativ geringen Datentransfer erfordern [Har96].

## 5.2.2 Programmiermodelle und Hilfsmittel

Grundsätzlich lassen sich zwei unterschiedliche Programmiermodelle für *MIMD*-Parallelrechner definieren:

- **Datenparallele Modelle (Datenparallelität):** Bei diesen Modellen werden gleiche Programmabschnitte bei unterschiedlichen Datenbereichen parallel ausgeführt. Vor allem Matrizen- und Vektoroperationen lassen sich hiermit über *Compileranweisungen* (z.B.: *High Performance Fortran*) leicht parallelisieren. Der Vorteil der einfachen Programmierung ist jedoch mit einem gravierenden Nachteil verbunden: Bei Maschinen mit verteiltem Speicher ist meist ein sehr hoher Datentransfer erforderlich, um jeden Prozessor mit den notwendigen Eingabe- bzw. Ausgabedaten zu versorgen. Deshalb wird dieses Modell meist bei *shared memory* Systemen eingesetzt.
- **Nachrichtenaustausch (Funktionelle Parallelität):** Die funktionelle Parallelität wird über eine Aufspaltung eines sequentiellen Algorithmus in parallel ausführbare Teilabschnitte erreicht. Die verschiedenen Programmteile kommunizieren über einen Nachrichtenaustausch (*message passing*) miteinander. Dieses Modell kann also nur realisiert

\*nach *Graphentheorie*: 2D-Gitter, Torus, Baum, Hypercube, siehe hierzu auch [Läm97a] mit einer ausführlichen Beschreibung der Netztopologien

<sup>†</sup>Hersteller: z.B. *Hewlett-Packard, Sun, Silicon Graphics, IBM* ©

werden, wenn eindeutige Anweisungen im Programmcode den Datentransfer steuern. Die parallelisierten Programmteile stellen – abgesehen vom Datentransfer und Steueranweisungen – eigenständige Programme dar.

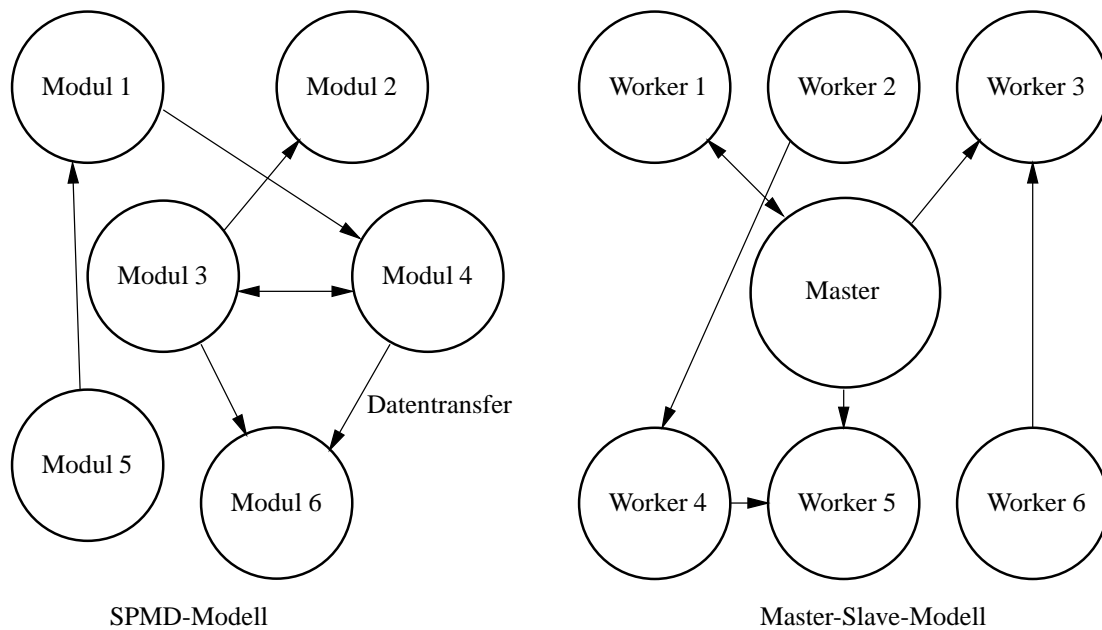
Parallele Programmiermodelle lassen sich auch bezüglich ihrer zeitlichen bzw. räumlichen Parallelität unterscheiden. Wird ein zeitabhängiger Prozeß in unterschiedliche Zeitabschnitte unterteilt die parallel berechnet werden, so handelt es sich um eine zeitparalleles Modell. Es ist naheliegend, daß dies in beschränktem Umfang nur bei linearen Untersuchungen angewandt werden kann, da jedes Zeitintervall Eingangsdaten aus dem vorherigen Intervall benötigt.

Wesentlich besser lassen sich Probleme parallel verarbeiten, die in Teilaufgaben oder Teilbereiche zerlegt werden können und nur an den gemeinsamen Schnittstellen einen Datenaustausch benötigen (räumliche Parallelität). Den einfachsten Sonderfall stellen hierbei Parameterstudien an einem Rechenmodell dar, bei denen die einzelnen Berechnungen unabhängig voneinander ausgeführt werden können. Dann kann jeweils auf einem Rechenknoten (z.B.: Workstation) ein Rechenlauf abgearbeitet werden – eine parallele Programmierung im eigentlichen Sinne erübrigt sich. Sollen dagegen Rechenmodelle während der Simulation miteinander abgeglichen werden (z.B. in der Klimaforschung), kann mit Hilfe der parallelen Programmierung ein Datenaustausch sowie eine zeitliche Synchronisation vorgenommen werden.

Bei einer Parallelisierung im Sinne der vorliegenden Arbeit wird entweder ein Problem in geometrische Teilbereiche zerlegt, die dann jeweils auf einem Rechner bearbeitet werden oder es werden Teile von Programmalgorithmien auf verschiedene Rechner verteilt. Die programmtechnische Umsetzung der Parallelisierung kann wiederum mit verschiedenen Modellen realisiert werden (vgl. Abbildung 5.1):

- Wird nur ein lauffähiges Programm verwendet (ein Programmcode), das beliebig oft gestartet werden kann, so handelt es sich um ein *SPMD*-Modell (*Single Program, Multiple Data*). Die einzelnen Programme können gleichberechtigt nebeneinander arbeiten und sich gegenseitig steuern.
- Alternativ wird ein Modell verwendet, bei dem ein Hauptprogramm (*Master*) die Steuerung beliebig vieler Nebenprogramme (*Worker* oder *Slave*) übernimmt. Meist werden deshalb zwei unterschiedliche Programmcodes für Master und Worker verwendet. Es ist auch möglich, mit einem Programmcode zu arbeiten, wenn dieser so programmiert ist, daß während der Laufzeit die entsprechenden Master- oder Worker-Routinen aktiviert werden [RK98].

Beide Modelle sollten auf unterschiedlichen *MIMD*-Rechnersystemen verwendet werden können. Ist dies der Fall, so spricht man von einer *Portabilität* der Software. Es ist jedoch schwierig, eine optimale parallele Software für unterschiedliche Systeme zu entwickeln, da beispielsweise die Leistungsfähigkeit des Datentransfers in Parallelrechnern mit fester Topologie im Allgemeinen wesentlich höher ist als in Workstation-Clustern. Die Effizienz (vgl. Kapitel 5.2.3) eines Programmsystems ist deshalb auch von der Rechnerarchitektur abhängig.



**Abb. 5.1:** Unterschiedliche Programmiermodelle

### Synchronisation und Datenaustausch

Oft werden mit parallelen Programmen Aufgaben bearbeitet, die schrittweise gelöst werden. Hierzu müssen zwischen den einzelnen Modulen Ausgabe- bzw. Eingabedaten ausgetauscht werden. Dieser Vorgang muß zeitlich synchronisiert werden, damit sich die Daten zum erforderlichen Zeitpunkt am richtigen Ort befinden. Darüberhinaus sollte die Steuerung des parallelen Programmsystems möglichst zentral mit einem Modul erfolgen, damit das System für den Anwender einfach zu verwenden ist (Start, Laufzeitüberwachung, Stop).

Neben Routinen zur Steuerung des Programmsystems (starten und beenden von Prozessen) sind deshalb vor allem Mechanismen erforderlich, die einen Datenaustausch sowie die damit verbundene Synchronisation übernehmen.

Der Nachrichtenaustausch zwischen zwei Prozessen wird als *message passing* bezeichnet. Dabei stellen die Kommunikationsschritte *Senden* und *Empfangen* zwei getrennte Vorgänge dar. Diese Prozesse lassen sich jeweils als *blockierende* und *nichtblockierende* Kommunikation ausführen:

- Beim Vorgang **blockierendes Senden** wird der sendende Prozeß solange angehalten, bis der Empfänger den Erhalt der Nachricht quittiert hat.
- Entsprechend wird der empfangende Prozeß beim **blockierenden Empfangen** solange angehalten, bis die erwartete Nachricht eingetroffen ist.
- **Nichtblockierendes Senden** wird verwendet, wenn der Sender nach dem versenden der Daten sofort weiterarbeiten soll. Falls der Empfänger die Daten nicht gleich aufnehmen kann, müssen geeignete Maßnahmen zur temporären Speicherung vorhanden sein.

- Beim **Nichtblockierenden Empfangen** überprüft der Empfänger, ob Daten für ihn bereitliegen und arbeitet weiter, wenn nichts abgerufen werden kann. Dieser Vorgang kann mehrmals durchgeführt werden.

Eine enge zeitliche Synchronisation bieten also nur die blockierenden Prozesse. Die jeweiligen Sende- und Empfangsarten können je nach Anforderung gemischt werden. Wird sowohl das Senden als auch das Empfangen der Daten als blockierender Prozeß ausgeführt, so spricht man auch von einer *synchronen Kommunikation*.

*Asynchrone Kommunikation* liegt entsprechend vor, wenn entweder das Senden nichtblockierend ausgeführt wird (*asynchrones Senden*) oder der Empfang (*asynchrones Empfangen*). Weitergehende Formen können aus den hier beschriebenen aufgebaut werden (z.B.: Senden an alle anderen Prozesse – *broadcast*).

Damit die jeweiligen Daten an den richtigen Ort gelangen, muß eine genaue Zuordnung erfolgen. Dies wird über einen *Kanal* realisiert. Ein *Kanal* stellt eine unidirektionale Verbindung zwischen zwei Prozessen dar (Sender und Empfänger). Programmtechnisch wird ein Sendevorgang oder eine Gruppe von Sendevorgängen mit einer bestimmten Kanalnummer belegt, die der Empfänger a priori kennt. Der Name des Absenders kann in der Nachricht enthalten sein.

## Programmierhilfen

Herkömmliche Programmiersprachen und Betriebssysteme bieten keine Möglichkeiten, Prozesse auf anderen Rechnern zu starten oder Prozesse kommunizieren zu lassen. Allerdings gibt es seit einigen Jahren zahlreiche Hilfsmittel, mit denen Möglichkeiten zur Prozeßsteuerung sowie zur Kommunikation verfügbar sind. Die wichtigsten Vertreter sind:

- **PARIX** stellt eine Erweiterung des Betriebssystems *UNIX* dar und wird nur auf speziellen Rechnerarchitekturen verwendet. Die parallelen Programme können mit Standardbefehlen programmiert werden (C oder Fortran). Es werden lediglich spezielle Bibliotheken zur Steuerung und Kommunikation zugebunden. Damit sind diese Programme nicht portierbar [BDF<sup>+</sup>93].
- **PVM** (Parallel Virtual Machine) ist ein weit verbreitetes, kostenloses Programmpaket. Es werden Fortran- und C-Bibliotheken für sehr unterschiedliche Plattformen bereitgestellt. Mit PVM können auch heterogene Rechner\* zu einem Parallelrechner zusammengeschlossen werden, sofern diese mit einem Netzwerk verbunden sind [DGMS96]. PVM stellt nicht nur Hilfsmittel für den Datentransfer und die Synchronisation zur Verfügung, es erlaubt auch eine dynamische Prozeßsteuerung. Darüberhinaus gibt es zahlreiche Programme und Bibliotheken, die PVM verwenden oder mit denen die parallele Programmierung erleichtert wird<sup>†</sup> [BS96, G<sup>+</sup>94].

\*z.B.: *UNIX-Workstations* verschiedener Hersteller, *Linux-PC's* und unterschiedliche Parallelrechner

<sup>†</sup>z.B.: *XPVM* zur Visualisierung des Programmablaufs



- **MPI** (Message-Passing-Interface) stellt eine standardisierte Form der parallelen Programmierung dar. Es wird versucht, auf homogenen Rechnersystemen mit Hilfe hardware-spezifischer Bibliotheken eine optimierte Kommunikation und Steuerung zu implementieren [Hem96, MPI95, MPI97].

PVM eignet sich besonders für den Einsatz auf heterogenen Rechnern, deshalb wurde es für die im Rahmen dieser Arbeit erstellten Programme verwendet.

### 5.2.3 Effektivität der Parallelisierung

Eine effiziente Parallelisierung setzt bestimmte Eigenschaften des sequentiellen Algorithmus bzw. der zu bearbeitenden Aufgabenstellung voraus. Grundsätzlich muß der zusätzliche Rechenaufwand sowie der Aufwand für die Kommunikation des parallelen Programmes (*Overhead*) möglichst klein gehalten werden. Der Overhead setzt sich aus drei Anteilen zusammen:

- Synchronisation der verschiedenen Prozesse,
- zusätzlicher Aufwand zur Datenaufbereitung, Aufgabenverteilung und dynamischen Lastverteilung
- und der Kommunikation zwischen den Prozessen.

Die quantitative Beschreibung der Leistungsfähigkeit einer parallelen Applikation erfordert eine genaue Definition verschiedener Zusammenhänge, die im folgenden dargestellt werden.

#### Granularität

Der gesamte Algorithmus sollte eine genügende Granularität besitzen, d.h. er muß in ausreichend große Teilalgorithmen (Teilprozesse) aufgeteilt werden können, in denen die zu berechnenden Daten nicht von anderen Teilalgorithmen gleichzeitig verwendet oder angefordert werden. Die Abschätzung der Granularität kann über den *Volume-to-Surface-Quotienten* erfolgen:

$$VSQ = \frac{\text{Rechenoperationen Teilprozeß}}{\text{Anzahl Eingangs- und Ergebnisdaten}} \quad (5.1)$$

Bei einem großen *VSQ* spricht man von einem grobgranularen Algorithmus, ein kleiner *VSQ* bezieht sich auf einen feingranularen Algorithmus. Der Datentransfer stellt bei Systemen mit verteiltem Speicher immer einen "Flaschenhals" dar, deshalb ist ein grobgranularer Algorithmus vorteilhaft.



### Skalierbarkeit

Mit Skalierbarkeit wird die möglichst einfache Anpassung eines parallelen Programmsystems an eine unterschiedliche Anzahl von Prozessoren bezeichnet. Es ist naheliegend, daß die Skalierbarkeit nicht nur vom verwendeten Algorithmus, sondern auch von der aktuell behandelten Problemgröße abhängt. Die Skalierbarkeit von datenparallelen Modellen ist meist besser als die der funktionell-parallelen Modelle.

### Leistungsbewertung

Die Steigerung der Leistung eines parallelen Programmsystems gegenüber dem sequentiellen Algorithmus kann mit dem *Speedup*  $S(p)$  beurteilt werden:

$$S(p) = \frac{T_1}{T_p}, \quad (5.2)$$

wobei mit  $T_1$  die Rechenzeit sequentiellen Programms bezeichnet wird und mit  $T_p$  die Rechenzeit des parallelen Programms bei Verwendung von  $p$  Prozessoren.

Der Speedup ist also eine Funktion von  $p$ , der Werte zwischen 1 (keine Geschwindigkeitssteigerung) und  $p$  (idealer Speedup) annehmen kann. Werte kleiner als 1 deuten auf einen ungeeigneten Algorithmus hin. Ist  $S(p)$  größer als  $p$ , so spricht man von einem überlinearen Speedup. Die Ursache hierfür liegt darin begründet, daß die verwendeten Rechner kleinere Aufgaben überproportional schneller abarbeiten können als sehr große Aufgaben. Dieser Effekt ist in der unterschiedlichen Geschwindigkeit verschiedener Speicherkomponenten zu suchen (*Cache, RAM und Swap*).

Bezieht man den Speedup auf die Anzahl der verwendeten Prozessoren, so erhält man die Effizienz:

$$E(p) = \frac{S(p)}{p} \quad (5.3)$$

Die Effizienz nimmt Werte zwischen  $\frac{1}{p}$  und 1 an und wird meist in Prozent angegeben.

Die Gleichungen (5.2) und (5.3) gelten nur für einen homogenen, unbelasteten Parallelrechner. Werden Prozessoren verschiedener Geschwindigkeit verwendet, so muß  $T_1$  durch die mittlere Rechenzeit  $T_m$  des sequentiellen Programms auf allen Prozessoren ersetzt werden. Werden externe Lasten (z.B. Systemprozesse oder Prozesse anderer Nutzer) mit einem zufälligen Lastbild einbezogen, so kann eine *stochastische Effizienz* definiert werden (siehe hierzu [Sch94]).

### Amdahl's Gesetz

Unter der Annahme, daß sich jedes Programm aus einem parallelisierbaren Anteil und einem nicht parallelisierbaren Anteil zusammensetzt, ergibt sich nach *Amdahl* [Amd67] folgende

Formel für die Rechenzeit auf einem Rechner:

$$T_1 = T_{seq} + T_{par} \quad (5.4)$$

Werden  $p$  Prozessoren verwendet, ergibt sich die Rechenzeit zu:

$$T_p = T_{seq} + \frac{T_{par}}{p} \quad (5.5)$$

Verwendet man relative Rechenzeiten ( $t_{seq}$  und  $t_{par}$ , mit  $t_{seq} + t_{par} = 100\%$ ), so erhält man den maximal erreichbaren Speedup:

$$S_{max} = \lim_{p \rightarrow \infty} \frac{1}{t_{seq} + \frac{1-t_{seq}}{p}} = \frac{1}{t_{seq}} \quad (5.6)$$

Abhängig von der Größe des nicht parallelisierbaren Rechenaufwandes  $t_{seq}$  ist also der Speedup begrenzt. So wird bei  $t_{seq} = 5\%$  ein maximaler Speedup von 20 erreicht. Eine Rechanlage mit weit mehr als 20 Prozessoren würde für diesen Fall also keine bedeutsame Erhöhung des Speedup's bringen. Somit wären *massiv parallele Systeme* mit einigen hundert Prozessoren sinnlos, da jeder Programmcode sequentielle Anteile enthält. Allerdings nimmt *Amdahl* an, daß das Verhältnis von  $t_{seq}$  und  $t_{par}$  für alle Problemgrößen konstant ist. Deshalb wird in diesem Zusammenhang auch von einem *fixed size Speedup* gesprochen.

### Gustafson's Gesetz

Unter der Annahme, daß der parallelisierbare Anteil mit zunehmender Problemgröße  $n$  schneller als der sequentielle Anteil wächst, kann nach *Gustafson* [Gus88] ein *scaled Speedup* definiert werden:

$$S_{sc} = \frac{t_{seq}(n) + t_{par}(n, 1)}{t_{seq}(n) + t_{par}(n, p)} \quad (5.7)$$

Bei vielen parallelisierbaren Problemen kann vereinfachend angenommen werden, daß

$$\begin{aligned} t_{par}(n, p) &\approx \frac{n^{pp}}{p}, \text{ mit } pp \geq 2 \text{ und} \\ t_{seq}(n) &\approx n^{ps}, \text{ mit } 0 \leq ps < 2 \end{aligned} \quad (5.8)$$

gilt\*. Dies kann meist auch für die Finite-Element-Methode angenommen werden [Boe95, BDF<sup>+</sup>93, Mey97]. *Lämmer* nimmt in [Läm97a] an, daß  $pp = 2$  und  $ps = 1$  gesetzt werden kann. Auch wenn dies für  $ps$  zu optimistisch ist, so kann mit diesen Werten mit Hilfe einer Grenzwertbetrachtung gezeigt werden, daß für zunehmende Problemgrößen  $n$  ein beliebig großer Speedup erzielt werden kann:

$$\begin{aligned} S_{scmax} &= \lim_{n \rightarrow \infty} \frac{n + n^2}{n + \frac{n^2}{p}} \\ &= \lim_{n \rightarrow \infty} \frac{pn + pn^2}{pn + n^2} \\ &= p \end{aligned} \quad (5.9)$$

\*für große  $n$  sind konstante Anteile bei  $t_{seq}$  und  $t_{par}$  vernachlässigbar

Die Grenzwertbetrachtung ( $n \rightarrow \infty$ ) ist mit realen Programmen nicht erreichbar, da nicht davon ausgegangen werden kann, daß sich der parallelisierte Programmteil beliebig oft parallelisieren läßt\*. Vor allem kann jede Rechenanlage nur eine endlich große Aufgabe bearbeiten. Jedes Problem besitzt deshalb einen maximalen Parallelitätsgrad  $p_{max}$  [Hei94].

Einen erheblichen Einfluß auf die tatsächliche Leistungsfähigkeit eines parallelisierten Programms hat auch die Datenkommunikation zwischen den Prozessoren. Die Dauer einer Datenübertragung hängt nicht nur von der Datenmenge, sondern auch von einer annähernd konstanten Initialisierungsphase ab. Gerade bei sehr vielen Prozessoren und einem vergleichsweise langsamen Kommunikationsnetz ist auch dadurch der erreichbare Speedup begrenzt.

Trotzdem ist ein parallelisierter Algorithmus besonders bei der Berechnung von großen Systemen einem sequentiellen Ablauf überlegen. Der problemspezifische maximale Speedup kann nur erreicht werden, wenn alle Prozessoren gleichmäßig schnell ihre jeweils zugeordnete Aufgabe erledigen.

### Load balancing

Unterschiedliche Rechenleistungen bzw. Geschwindigkeiten können viele Ursachen haben:

- Unterschiedliche Prozessoren (Architektur, Taktfrequenz)
- Unterschiedliche Leistungsfähigkeit von wichtigen Komponenten der Rechenknoten (Rechenspeicher, Festplatte)
- Unterschiedliche Netzwerkanbindung (Transferleistung, Signallaufzeit)
- Zusätzliche Belastung einzelner Prozessoren durch Systemprozesse und Prozesse anderer Nutzer

Mit Hilfe eines Lastausgleichs (*load balancing*) können unterschiedliche Rechenleistungen im Parallelrechner mit dem Ziel des optimalen Speedup's berücksichtigt werden. Eine Lastverteilung (*load sharing*) stellt lediglich sicher, daß alle Prozessoren des verwendeten Parallelrechners eine Aufgabe bearbeiten [Hei94]. Die Terminologie ist in der Literatur uneinheitlich; so verwendet z.B. *Schneckenburger* [Sch94] den Begriff *adaptive Lastverteilung* statt *load balancing*.

Der Lastausgleich kann durch verschiedene Maßnahmen erfolgen:

- Ein **Statischer Lastausgleich** kann angewendet werden, wenn der Parallelrechner über eine homogene Struktur verfügt und nicht von anderen Prozessen während der Laufzeit beeinflusst wird. Die Zuteilung der Aufgaben erfolgt bei Finite-Element-Berechnungen entweder über geometrisch orientierte Verfahren (Netzpartitionierung) oder an abstrakten Gebilden (z.B. an der Gesamtsteifigkeitsmatrix).

---

\*die Skalierbarkeit ist also begrenzt

Bei adaptiven Netzanpassungen oder bei nichtlinearen Berechnungen kann während der Berechnung ein Ungleichgewicht zwischen den ursprünglich gewählten Bereichen eintreten, wodurch ein erneuter Lastausgleich notwendig werden kann [LM97b].

- **Implizite Methoden** sind in manchen Algorithmen bereits a priori enthalten. Diese lassen sich z.B. in zufallsgesteuerten Suchalgorithmen finden [Sch94]. Da dies jedoch eine seltene Ausnahme darstellt, wird diese Vorgehensweise hier nicht weiter verfolgt.
- Das **Task Farming** (auch *Pool of Tasks*) kann angewendet werden, wenn viele unabhängige Teilaufgaben bearbeitet werden. Jedem Prozessor wird eine neue Aufgabe zugeteilt, falls er seine vorhergehende Aufgabe fertig bearbeitet hat. Dies geschieht solange, bis alle Aufgaben abgearbeitet sind. Diese einfach zu steuernde Methode kann nur bei einer genügend großen Menge von Teilaufgaben eingesetzt werden – sie muß wesentlich größer als die Anzahl der zur Verfügung stehenden Prozessoren sein. Beim Einsatz des Task Farming im Bereich der Finite-Element-Methode entsteht ein enormer Datentransfer, weshalb diese Methode hierzu kaum verwendet wird [WR99].
- Der **Datenaustausch** zwischen einzelnen Prozessen stellt gerade für die Finite-Element-Methode eine weit verbreitete Möglichkeit dar, mit einem relativ geringen Datentransfer eine wirkungsvolle adaptive Lastverteilung zu implementieren. Mit Hilfe eines statischen Lastausgleichs wird jedem Prozessor zunächst ein bestimmtes Gebiet (Datenbereich) zugewiesen, das während der Berechnung jeweils vergrößert oder verkleinert wird, je nachdem, ob ein Prozessor langsamer oder schneller als die übrigen Prozessoren die zugeordnete Aufgabe bearbeitet.

Die Berechnung der Teilgebiete kann ohnehin meist nicht unabhängig voneinander erfolgen, weshalb ein Datenaustausch zwischen benachbarten Gebieten obligatorisch ist. Die Minimierung des gesamten Datentransfers erfordert zusammenhängende und gleichmäßig berandete Gebiete, was einem möglichst flexiblen *load balancing* entgegensteht (siehe Abschnitt 5.3.1 und [Läm97a, Mey97]).

Darüberhinaus gibt es die Möglichkeit, daß ein spezielles Lastverteilungssystem ganze Prozesse einschließlich ihrer zugeordneten Daten von einem überlasteten Prozessor auf einen weniger belasteten oder generell leistungsfähigeren Prozessor verschiebt [Sch94, Sti93]. Dieses Vorgehen erfordert einen sehr großen Aufwand und ist nur bei wenigen Anwendungen sinnvoll. Auch die automatische Zuteilung von Prozessen auf geeignete Rechner (*load sharing*) soll im Rahmen der vorliegenden Arbeit nicht betrachtet werden (siehe hierzu z.B. das System *Condor* in [PL94]).

### 5.3 Parallelisierung der dynamischen Berechnung

Wie bereits in Abschnitt 5.2.2 erwähnt wurde, ist eine Aufteilung des Anfangswertproblems in mehrere Abschnitte nicht möglich, da jeder Abschnitt die Ausgangsdaten des Vorgängers als Eingangsdaten benötigt. Somit bietet sich für eine parallele Berechnung nur die eigentliche Finite-Element-Berechnung an [CF88].

Die Kopplung von Freiheitsgraden entsteht lediglich aus der geometrischen Beziehung der jeweiligen Elemente. Es können deshalb die üblichen Methoden der parallelen Finite-Element-Berechnung verwendet werden [HF98, RS99].

Bei der parallelen Programmierung ist vorab zu entscheiden, auf welchem Rechnersystem das parallele Programm verwendet wird:

- Soll ein *massiv paralleles System* mit sehr vielen Prozessoren eingesetzt werden, dann ist es unbedingt notwendig, daß der gesamte Programmcode parallelisiert wird, um einen möglichst hohen Speedup zu erzielen [JT98]. Als Programmiermodell bietet sich meist das *SPMD*-Modell an (vgl. Abschnitt 5.2.2).
- Für den Einsatz auf einem *Workstation Cluster* mit weniger als 10-15 Rechnern kann ein geringer sequentieller Anteil (kleiner als 10%) hingenommen werden. Nach *Amdahl's Gesetz* beträgt der maximale Speedup mit  $t_{seq} \approx 10\%$  und 10 Rechnern ca. 5.3. Bei  $t_{seq} \approx 5\%$  und 10 Rechnern liegt er bereits bei 6.9, also nahe am theoretisch erreichbaren Speedup von 10. Bei einer zunehmenden Problemgröße sollte nach *Gustafson's Gesetz* der sequentielle Anteil langsamer wachsen als der parallelisierte Anteil.

### 5.3.1 Gebietszerlegung

Hierbei werden geometrisch zusammenhängende Teilbereiche der FE-Diskretisierung so aufgeteilt (*partitioniert*), daß jeder Prozessor ein zusammenhängendes Gebiet erhält. Jeder Prozessor kann die Steifigkeitsmatrix für sein Teilgebiet unabhängig von den Nachbargebieten generieren. Die Gebiete werden im Inneren mit einer Substrukturtechnik bearbeitet. Die Verbindung zu den Nachbargebieten wird über einen Datenaustausch bei der iterativen Gleichungslösung gewährleistet. Es ist deshalb notwendig, daß die Länge der Grenzen insgesamt minimiert wird, um einen minimalen Datentransfer zu erhalten.

Die berechneten Ergebnisse liegen verteilt auf verschiedenen Rechnern, weshalb geeignete Maßnahmen zur zentralen Speicherung oder zum verteilten *Post-Processing* getroffen werden müssen.

Zum Themenbereich Gebietszerlegung gibt es viele Verfahren, die in der Literatur ausführlich diskutiert werden (z.B.: [FR94, Kla96, Läm97a, LB97, LM97b, Mey97, LT94, IST97]).

### Parallele Gleichungslösung

Die Lösung von sehr großen, linearen Gleichungssystemen stellt in der *Finite-Element-Berechnung* meist eine der aufwendigsten Operation dar. In der Vergangenheit wurden hierzu viele Verfahren entwickelt und verfeinert (z.B.: *konjugiertes Gradienten*-, *Lanczos*- sowie *Mehrgitterverfahren*) [Boe95, BDF<sup>+</sup>93, CA96, DDNR96, FES98, Läm97a, Mey97, RALS99, RSLA98, Sch96a].

### 5.3.2 Parallelisierung auf Elementebene

In manchen Bereichen spielt der numerische Aufwand zur Gleichungslösung im Vergleich zu anderen Programmteilen nur eine untergeordnete Rolle:

- Bei Verwendung einer expliziten Zeitintegration für physikalisch stark nichtlineare Probleme wird nahezu die gesamte Rechenzeit für die Ermittlung der Dehnungen, Spannungen und inneren Kräfte der Elemente benötigt. Es ist deshalb sinnvoll, nur diesen Programmteil zu parallelisieren [DNO<sup>+</sup>97].
- Die adaptive Berechnung mit Hilfe der p-Version der Finite-Element-Methode führt zu hohen Polynomgraden der Ansatzfunktionen. Der numerische Aufwand zur Erstellung der Elementmatrizen ist deshalb im Vergleich zur Gleichungslösung wesentlich größer. Rank/Rücker et al verwenden die Parallelisierung deshalb nur zur Elementberechnung (Elementmatrizen und Nachlaufrechnung), während die Gleichungslösung sequentiell erfolgt [RK98, RKR98, RRDB99].
- Semi-analytische Finite-Elemente (z.B. Ring-Elemente mit Fourier-Reihenansatz in Umfangsrichtung nach Wunderlich et al [Wun67, WCO85, WOSL87]) benötigen bei physikalisch und geometrisch nichtlinearen Berechnung ebenfalls nahezu die gesamte Rechenzeit auf Elementebene. Die Gleichungslösung nimmt hier weniger als 10% der Gesamtrechenzeit ein [Rap98, WR99].

Die Parallelisierung auf Elementebene ist also nur für spezielle Vorgehensweisen sinnvoll. Unter bestimmten Voraussetzungen besitzt sie jedoch gerade für die Parallelisierung semi-analytischer Ring-Elemente Vorteile gegenüber der Gebietszerlegung, wie die folgende Beschreibung zeigt.

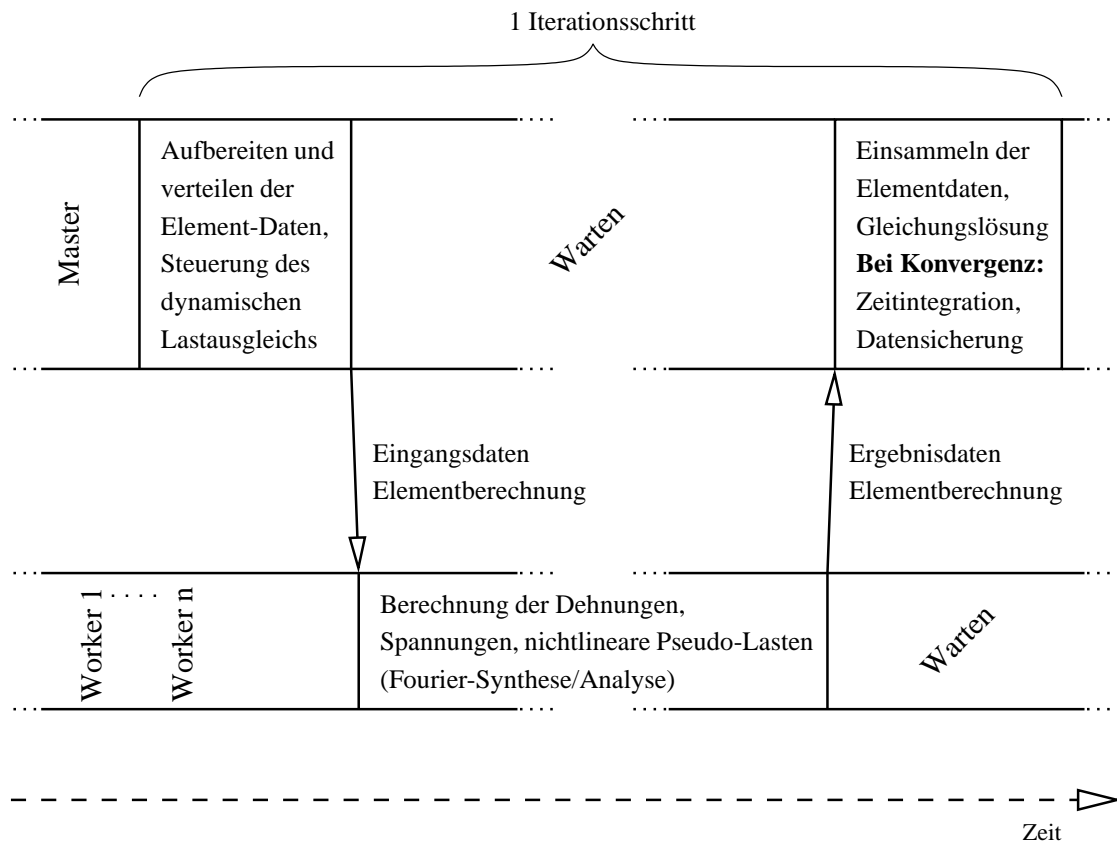
#### Programmiermodell

Als Programmiermodell bietet sich das *Master-Worker* Modell an (auch als *Master-Slave* Modell bezeichnet), bei dem die parallele Elementberechnung mit den Workern, die sequentielle Berechnung mit dem Master-Modul übernommen wird. In Abbildung 5.2 ist die zeitliche Zuordnung sowie der Datentransfer für einen Iterationsschritt dargestellt.

Zu Beginn eines Iterationsschrittes bereitet der Master die notwendigen Eingangsdaten auf und sendet sie an die Worker. Bei Bedarf steuert er auch den dynamischen Lastausgleich (siehe Abschnitt 5.3.3).

Die Worker haben entsprechend der speziellen Finite-Element-Formulierung (siehe Abschnitt 2.2.3, Seite 10) folgende Aufgaben abzuarbeiten:

- Bestimmung des Element-Verschiebungsvektors in jeder Fourier-Harmonischen
- Ermittlung der Verzerrungen an diskreten Stellen über den Umfang



**Abb. 5.2:** Schematischer Programmablauf für einen Iterationsschritt

- Berechnung des neuen Spannungszustandes  $\Delta\sigma$  und der nichtlinearen Anteile  $\Delta\sigma_{nl}$  aus den inkrementellen Dehnungen
- Analyse der Fourier-Koeffizienten der nichtlinearen Spannungsanteile  $\Delta\sigma_{nl}$
- Ermittlung des äquivalenten Knotenlastvektors der nichtlinearen Spannungsanteile in globalen Koordinaten  $\Delta\mathbf{P}_{nl}$

Diese numerisch sehr aufwendigen Berechnungen müssen für jedes einzelne Element – also unabhängig von anderen Elementen – vorgenommen werden.

Am Ende des Iterationsschrittes sammelt der Master die Daten der Worker ein und löst das Gleichungssystem. Nach dem letzten Iterationsschritt (Konvergenz) speichert er die aktuellen Daten für den Postprozessor und führt die Zeitintegration durch.

Die Zuordnung der Elemente auf die Worker (*mapping*) kann mit zwei unterschiedlichen Methoden erfolgen:

- Der Master teilt den Workern jeweils beliebige Elemente zu. Es müssen also sowohl die jeweiligen Elementdaten (z.B. Spannungen und Dehnungen) des Grundzustandes bzw. des vorhergehenden Iterationsschrittes übertragen werden, als auch die zugeordneten Knotenweggrößen sowie deren Ableitungen. Diese Vorgehensweise bietet den enormen Vorteil, daß zum dynamischen Lastausgleich das *Task Farming* verwendet werden



kann, wodurch auch bei stark schwankenden externen Rechenlasten ein hervorragender Lastausgleich erreicht wird. Nachteilig wirkt sich vor allem der enorme Datentransfer aus, da sowohl die Menge an Daten als auch die Anzahl an notwendigen Transfers sehr groß wird (feine Granularität). Probeläufe beim Entwickeln der hier vorgestellten Methode haben gezeigt, daß dieser Datentransfer nur einen völlig unzureichenden Speedup zuläßt.

- Jeder Worker erhält zu Beginn der Simulation die gleiche Anzahl oder eine entsprechend der jeweils maximal möglichen Prozessorleistung berechnete Anzahl von Elementen. Die Elementdaten bleiben dem jeweiligen Worker fest zugeordnet, der Master oder andere Worker haben keinen Zugriff auf diese Daten. Der Datentransfer zwischen Master und Worker umfaßt deshalb nur Knotenverformungen (und deren Ableitungen im ersten Iterationsschritt). Der Worker sendet dem Master den berechneten Anteil des Pseudo-Lastvektors. Es werden also relativ wenige und vor allem kleine Datenblöcke ausgetauscht (grobe Granularität). Lediglich am Ende eines Iterationsschrittes (Konvergenz) werden die für den Postprozessor benötigten Daten zum Master gesandt und abgespeichert.

Die Parallelisierung auf Elementebene bietet vor allem bei der Berechnung gekoppelter Strukturen Vorteile: Es können verschiedene Elementarten relativ einfach auf einem Worker bearbeitet werden, da sie völlig unabhängig voneinander sind.

Im Gegensatz hierzu entstehen bei der Gebietszerlegung bei Verwendung unterschiedlicher Elemente schwer partitionierbare Gebiete, die einem hohen Speedup entgegenstehen und damit die parallele Effizienz beeinträchtigen.

### Parallele Programmierung

Die programmtechnische Umsetzung kann mit den in Abschnitt 5.2.2 beschriebenen Programmbibliotheken erfolgen. Mit einem moderaten Aufwand erhält man hiermit ein sehr gut portierbares und gut zu bedienendes System.

Aus Kosten- und Verfügbarkeitsgründen werden derzeit oft *Workstation-Cluster* für die Parallelisierung vorhandener FEM-Software verwendet [HA96, LMR<sup>+</sup>96, Thi95, WBD96]. Abgesehen von den Problemen des *dynamischen Lastausgleichs*, der relativ geringen Netzwerkleistung sowie der eingeschränkten Skalierbarkeit gelten die hiermit gewonnenen Erkenntnisse auch für andere *MIMD*-Parallelrechner mit verteiltem Speicher [Rap98].

Da bei der Programmierung der im Rahmen dieser Arbeit erstellten Software die Programm-bibliothek *PVM* (*Parallel Virtual Machine*) verwendet wurde, wird hier die visuelle Darstellung des Programmablaufes mit Hilfe des in [BS96] beschriebenen Systems *XPVM* gezeigt.

Alle Simulationen wurden auf einem *Workstation-Cluster* aus verschiedenen Rechnern der Firma *Hewlett-Packard* ausgeführt. Die Rechner sind über ein *LAN-Ethernet* mit einer unterschiedlichen Leistung (10 und 100 MBit/s) und einem *Switch* sternförmig verbunden. Die Portierung auf ein anderes System ist problemlos, sofern die jeweilige Rechnerarchitektur



von *PVM* unterstützt wird. Dies ist bei nahezu allen derzeit verfügbaren Systemen der Fall. Deshalb kann das Programm sogar auf einem heterogenen Rechnernetz verwendet werden.

Abbildung 5.3 zeigt einen typischen Programmablauf für eine Berechnung wie sie in Kapitel 6 dargestellt wird. Im oberen Bereich sind neben den Bedienungselementen der Visualisierungssoftware die verwendeten Rechner mit dem Verbindungsnetzwerk dargestellt.

Die untere Hälfte zeigt die räumliche Zuordnung der Daten sowie den zeitlichen Programmablauf (gezeigtes Zeitfenster: ca. 6 Sekunden). Auf dem Rechner "zuse" wird sowohl das Master-Modul als auch ein Worker abgearbeitet. Durch die zeitliche Trennung während der Simulation ist keine negative gegenseitige Beeinflussung der beiden Module gegeben. Dies setzt allerdings voraus, daß die Systemressourcen Rechenpeicher und temporärer Plattenpeicher ausreichend vorhanden sind.

Die Übereinstimmung zwischen dem theoretischen Programmablauf (Abbildung 5.2) und dem tatsächlichen Ablauf ist sehr gut zu erkennen: Links erhalten die Worker\* die Eingangsdaten vom Master. Der Datentransfer ("Message") erfordert sehr wenig Zeit, was am nahezu senkrechten Verlauf der Zuordnungs-Graphen erkennbar ist. Anschließend bearbeiten die Worker die erste Elementart (hier: 3-D Kontinuums-Elemente). Der Master bereitet währenddessen die Daten für die zweite Elementart auf (hier: Schalen-Elemente) und sendet die Daten wiederum an die Worker†. Die Worker senden die Ergebnisse zurück an den Master und warten auf die Eingangsdaten des nächsten Iterationsschrittes.

Der zweite hier gezeigte Iterationsschritt ist der letzte in diesem Zeitschritt, da alle Worker am Ende die Elementdaten für das *Post-Processing* an den Master senden (zu erkennen an den vielen "Message" Graphen).

In Abbildung 5.3 ist auch die bereits erwähnte Problematik des dynamischen Lastausgleichs zu erkennen: Während die Worker den ersten Iterationsschritt nahezu gleichzeitig beenden, geschieht dies im zweiten Schritt sehr unterschiedlich.

Die kumulierte Rechnerauslastung (siehe Abbildung 5.4) verdeutlicht diese Problematik nochmals. Der linke Block stellt wieder den ersten Iterationsschritt dar und ist beinahe senkrecht berandet. Der zweite Block besitzt entsprechend der unterschiedlichen Rechenzeiten der Worker eine stufenförmige Berandung. Die Ursache hierfür ist eine ungleich verteilte Rechenlast. Bei der Darstellung in Abbildung 5.4 ist zu beachten, daß der Parallelrechner aus sechs Prozessoren besteht, auf denen sieben Prozesse (*Tasks*) bearbeitet werden. Die nahezu durchgängige "Wartephase" im oberen Bereich ist deshalb lediglich auf das Warten eines Prozesses, nicht jedoch auf das Warten eines Prozessors zurückzuführen.

Für einen optimalen Speedup ist eine möglichst gleichmäßige Auslastung aller Rechner notwendig. Im folgenden Abschnitt wird diese Aufgabe diskutiert.

---

\*hier als "slave" bezeichnet

†jeweils *blockierendes Empfangen* als *asynchrone Kommunikation*

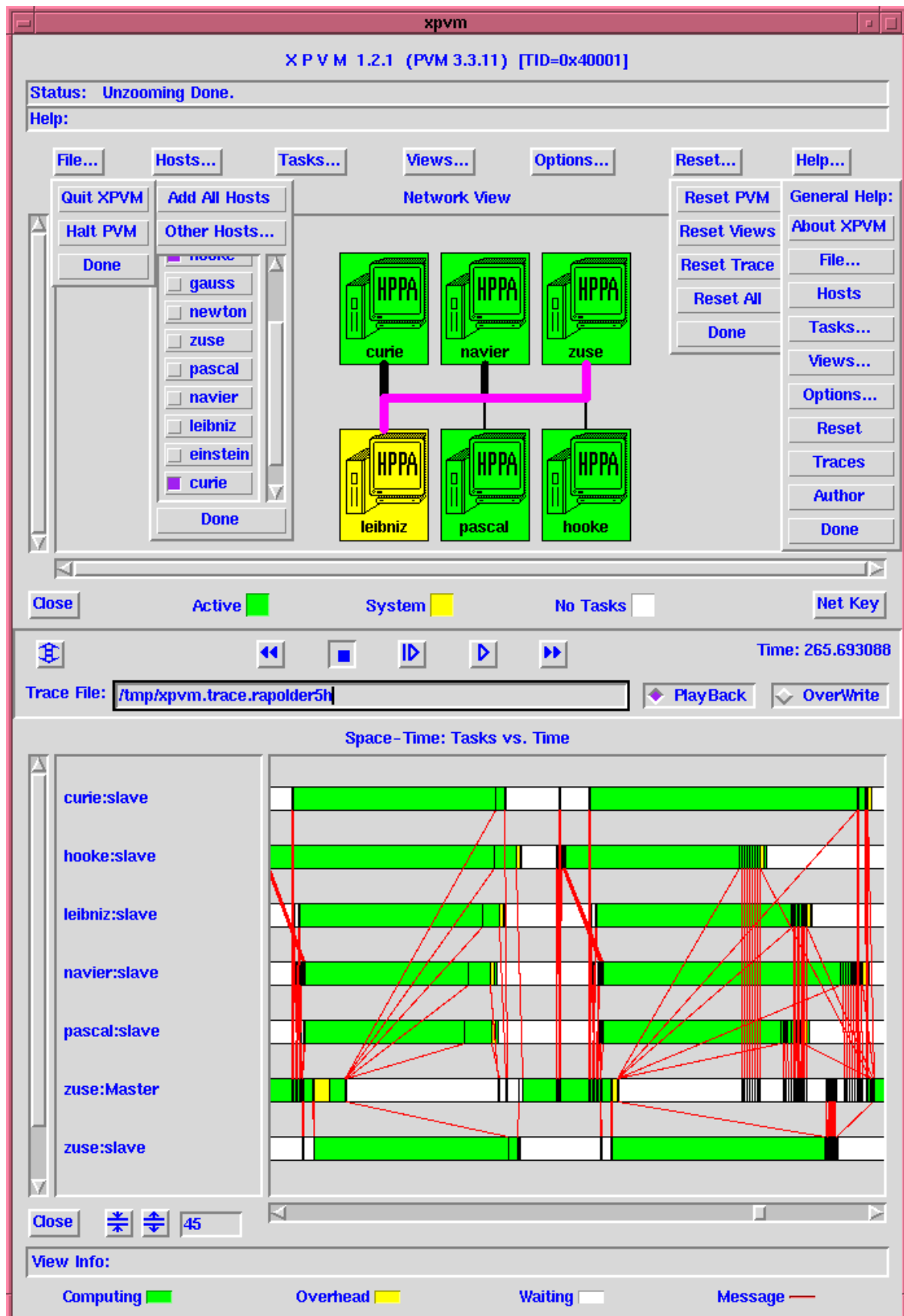


Abb. 5.3: Visualisierung des parallelen Programmablaufs: XPVM



Abb. 5.4: XPVM: Kumulierte Rechnerauslastung

### 5.3.3 Dynamischer Lastausgleich

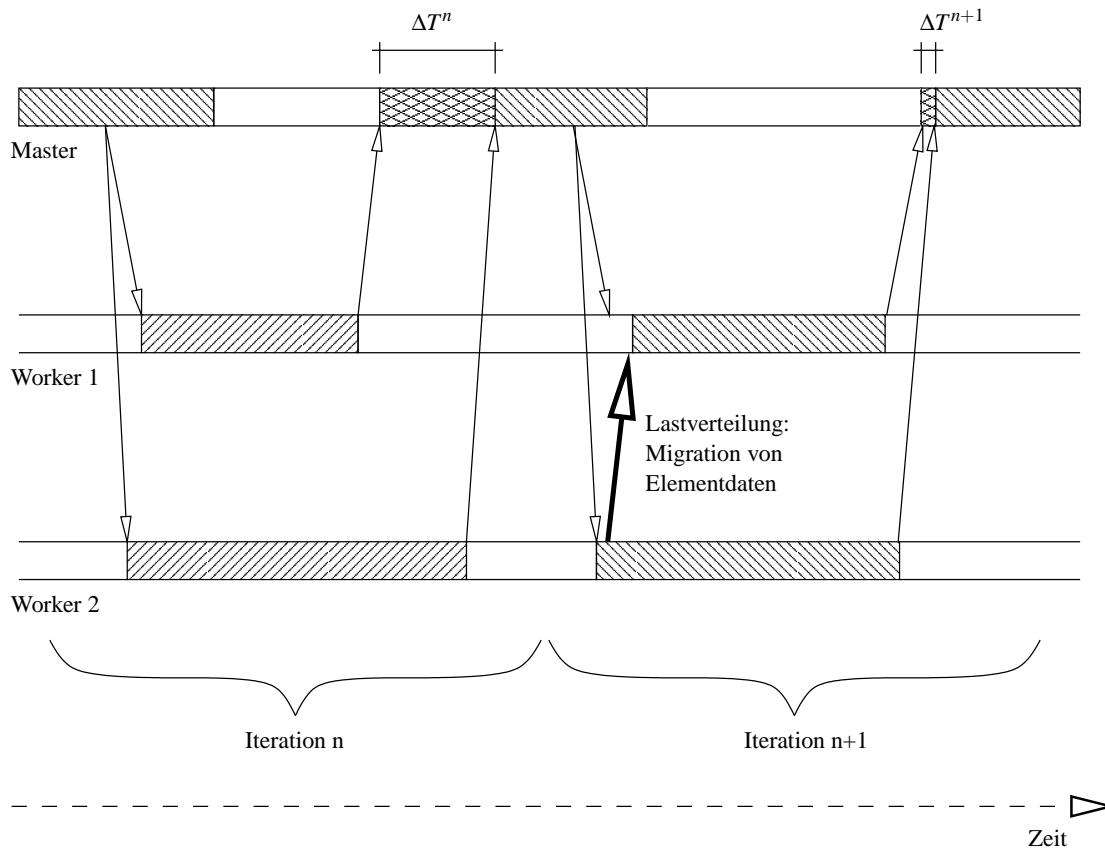
Die ungleiche Verteilung der Rechenlast kann sich sehr ungünstig auf die Effizienz des parallelen Programms auswirken (siehe Abschnitt 5.3.4). Die Ursachen für eine unausgewogene Rechenlast sind hier vor allem:

- Während der Simulation kann bei manchen Elementen ein wesentlich höherer Rechenaufwand auftreten als bei anderen. Dies ist eine Folge der physikalischen und geometrischen Nichtlinearitäten, die teilweise eine aufwendige Iteration auf Elementebene erfordern.
- Der Einsatz der parallelen Software auf heterogenen Rechner-Clustern führt zu den bereits in Abschnitt 5.2.3 dargestellten Problemen (unterschiedliche Rechner, Beeinflussung durch andere Nutzer usw.)

Aufgrund der festen Zuordnung der Elemente auf die Worker ist die einfachste Möglichkeit des Lastausgleichs mit Hilfe des *Task Farming* nicht möglich. Es wird deshalb die bereits in [Rap98, WR99] dargestellte Methode verwendet, wobei einige Modifikationen vorgenommen werden.

### Lokalisierung der ungleichen Lastverteilung

Zunächst muß ein geeigneter Parameter gefunden werden, mit dem die ungleiche Lastverteilung quantitativ bewertet werden kann. Ein einfach zu ermittelnder und trotzdem aussagefähiger



**Abb. 5.5:** Dynamischer Lastausgleich – Migration von Elementen

higer Wert ist die Zeitdifferenz zwischen dem jeweiligen Eintreffen der Antwort der Worker beim Master  $\Delta T^n$  (vgl. Abbildung 5.5). Dieser Wert bezieht sich nur auf den letzten Iterationsschritt und kann durch verschiedene Ereignisse beeinflusst werden:

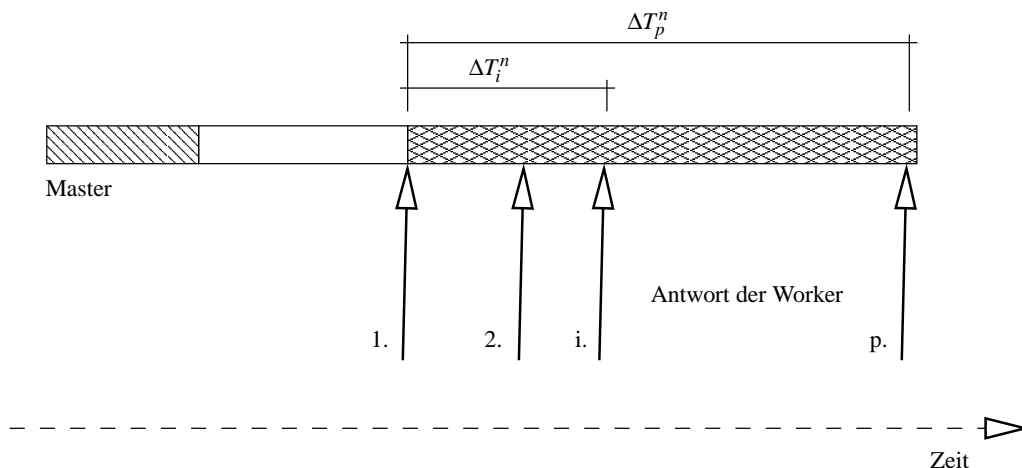
- Infolge der nichtlinearen FE-Berechnung können in manchen Bereichen aufwendige Iterationen auf Elementebene stattfinden. Diese sind meist auch im nächsten Iterationsschritt zu erwarten.
- Auf einem oder mehreren Rechnern sind kurzzeitig externe Lasten aufgetreten, die Systemressourcen verbraucht haben (CPU, Speicher usw.) und damit den jeweiligen Worker beeinträchtigt haben. Sofern dieses Ereignis mehrmals auftritt, sollte es für die nächste Iteration berücksichtigt werden, ansonsten nicht.
- Auf einem Rechner kann ein längerer Prozeß beendet worden sein. Zukünftig können also die freigewordenen Ressourcen verwendet werden.
- Auf einem Rechner kann ein langandauernder Prozeß gestartet worden sein. Dieser Rechner wird auch zukünftig nicht sehr leistungsfähig sein und sollte deshalb entlastet werden.

Aus dem Verhalten des Systems im letzten Iterationsschritt kann also nicht zuverlässig das zukünftige Verhalten abgeschätzt werden. Hierzu sind verbesserte Methoden nützlich:

- Mit Hilfe *neuronaler Netze* kann das Verhalten eines kompletten *Workstation Clusters* über lange Zeiträume hinweg gelernt werden. Hierbei wird nicht nur das technische Verhalten der Rechner, sondern auch das Verhalten der verschiedenen Nutzer (z.B. Art und Dauer der Rechnerauslastung) erfaßt. Dieses sehr aufwendige Vorgehen kann nur mit Hilfe spezieller Software umgesetzt werden [SLP94].
- Nach [Sch94] kann die externe Ressourcen-Nutzung über ein stochastisches Lastmodell erfaßt werden. Hieraus lassen sich relativ aufwendige Systeme entwickeln, die sich je nach Art des Rechnernetzes automatisch optimieren (z.B. das System *ALDY - Application Load Distribution sYstem* [Sch95]).

Es können aber auch relativ einfache Methoden verwendet werden. Im Folgenden wird eine Möglichkeit vorgestellt, die trotz ihrer einfachen Implementierbarkeit sehr gute Resultate liefert.

Zunächst wird angenommen, daß jedem verwendeten Prozessor genau ein Worker zugeordnet ist. Es existieren also  $p$  Prozessoren und  $p$  Worker. Das Eintreffen der Antworten der Worker



**Abb. 5.6:** Stochastisch verteilte Antwort der Worker

wird durch die bereits erwähnten Faktoren beeinflusst und erfolgt deshalb zufällig (Abbildung 5.6). Ziel ist die Minimierung der Differenz der unterschiedlichen Antwortzeiten:

$$\Delta T_i^n \rightarrow 0 \quad \forall \quad i \in p \quad (5.10)$$

Hierbei ist zu beachten, daß  $\Delta T_i^n$  als absoluter Wert gemessen wird. Es muß deshalb eine Bezugsgröße eingeführt werden. Naheliegend ist die aktuelle mittlere Rechenzeit  $T_{mittel}^n$  der Worker für einen Iterationsschritt. Als gut ausgeglichen kann nun ein Parallelrechner definiert werden, für den gilt:

$$\frac{\Delta T_p^n}{T_{mittel}^n} \leq 5\% \quad (5.11)$$

Darüberhinaus wird ein Vektor  $\mathbf{H}$  definiert, mit dem das Systemverhalten der letzten Iterationsschritte miteinbezogen wird:

$$H_i^n = \frac{H_i^{n-1} w \frac{\Delta T_i^n}{\Delta T_p^n}}{w + 1}, \quad 0 \leq H_i^n \leq 1 \quad (5.12)$$

Der Parameter  $w$  steuert die Wichtung des bisherigen Verhaltens im Verhältnis zum aktuellen Verhalten des Parallelrechners. Für kleinere Netzwerke mit wenigen externen Nutzern kann ein konstanter Wert (z.B.:  $1 \leq w \leq 5$ ) verwendet werden.

Bei größeren Anlagen mit einer großen Anzahl von Nutzern kann  $w$  optimiert werden. Hierzu wird  $w$  langsam in den genannten Grenzen variiert. Sobald Gleichung (5.11) über einen längeren Zeitraum ein Minimum annimmt, ist der optimale Wert für  $w$  gefunden worden.

Dieses Vorgehen eignet sich auch für portierbare Systeme, da für jeden Parallelrechner ein spezielles  $w$  gefunden werden kann. Bei stark schwankenden, kurzzeitigen Lasten ergibt sich ein relativ kleines  $w$ , bei einem Rechner mit lang andauernden Lasten bzw. einem unbelasteten Parallelrechner wird  $w$  groß. Es handelt sich hierbei also um die Vorstufe eines lernenden Systems.

### Zuordnung der Elemente

Die Zuordnung der Elemente auf die Worker (*Mapping*) erfolgt zu Beginn der Berechnung entweder gleichmäßig oder entsprechend der maximalen Leistungsfähigkeit der jeweiligen Prozessoren, da der aktuelle Zustand des Parallelrechners unbekannt ist. Alternativ kann mit Hilfe einer kurzen Testsimulation das aktuelle Verhalten vorab ermittelt werden.

Während des Programmablaufs muß anhand der ungleichen Lastverteilung ermittelt werden, welcher Worker wieviele Elemente von welchem Worker erhalten soll. Hierzu wird in [CCEA95] eine Methode vorgeschlagen, bei der mit Hilfe von Kostenfunktionen vorab für jedes Rechnernetz individuell ermittelt wird, welcher Aufwand für das Berechnen der Teilaufgaben, für das Versenden der Ein- und Ausgabedaten sowie für die Migration von Elementen erforderlich ist. Anhand dieser Kostenfunktionen kann beim eigentlichen Programmablauf ermittelt werden, ob sich eine Migration lohnt und wieviele Elemente transferiert werden müssen.

Diese Vorgehensweise erfordert allerdings einen beträchtlichen Aufwand zur Ermittlung der Kostenfunktionen. Außerdem müssen diese bei einem heterogenen Rechner-Cluster für jeden einzelnen Rechner ermittelt werden.

Eine wesentliche Vereinfachung ergibt sich, wenn statt der Kostenfunktionen allgemeine Annahmen getroffen werden:

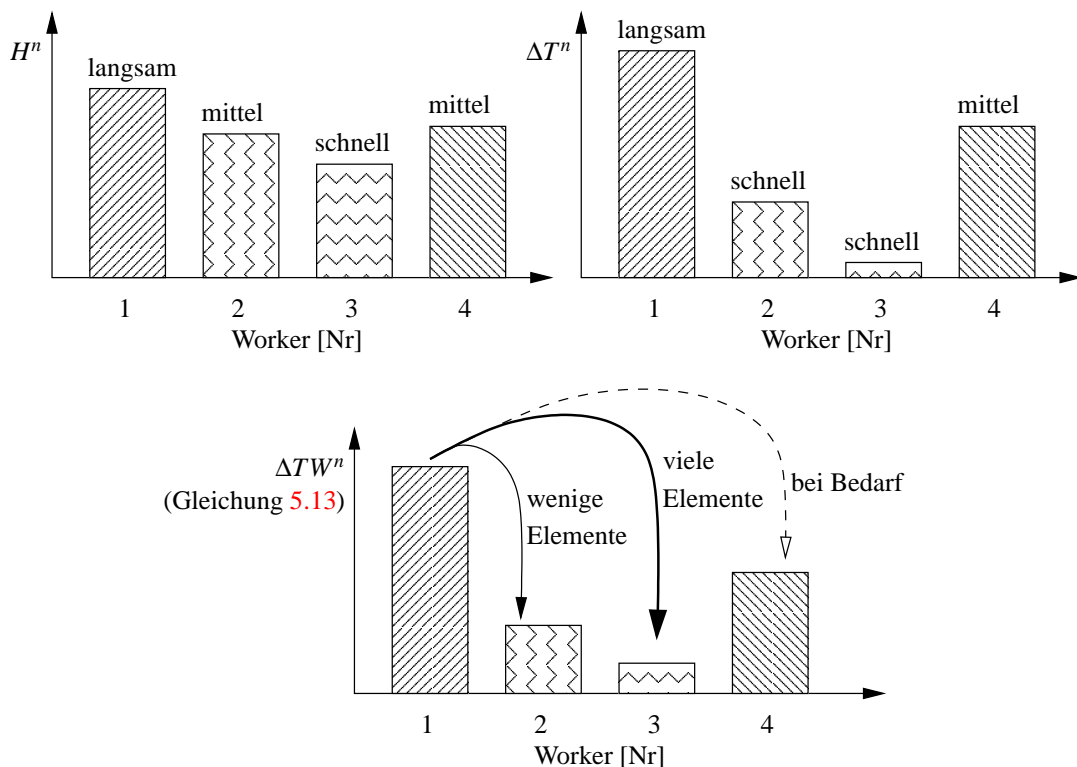
- Der Transfer der Eingangs- und Ausgangsdaten erfordert im Vergleich zur eigentlichen Berechnung sehr wenig Zeit. Es wird deshalb angenommen, daß die Datentransferzeit etwa 1% der Rechenzeit  $T_{par}$  beträgt (vgl. Abbildung 5.3). Da sie bei allen Prozessoren etwa gleich groß ist, kann sie vernachlässigt werden.

- Die jeweilige Zeit der Elementberechnung auf einem Worker wird für einen Iterationsschritt als konstant angenommen. Die bereits erwähnten ungleichen Rechenzeiten infolge der Nichtlinearitäten bleiben also hier unberücksichtigt, sie werden aber indirekt bei der Ermittlung der Gesamtrechenzeit des jeweiligen Workers berücksichtigt.
- Die Migration von Elementen erfordert einen relativ hohen Aufwand (auslesen und versenden vieler Daten, abspeichern der Daten beim Empfänger). Sie darf deshalb nur bei einer merklich ungleich verteilten Rechenlast vorgenommen werden. Gleichung (5.11) dient hierbei als Indikator.

Wird anhand Gleichung (5.11) ein ungenügender Lastausgleich festgestellt, muß zunächst mit Hilfe von  $\mathbf{H}$  (Gleichung 5.12) und  $\Delta T_i^n$  (Gleichung 5.10) die aktuelle Leistungsfähigkeit der jeweiligen Worker ermittelt werden. Hierzu wird  $\Delta T_i^n$  mit  $\mathbf{H}$  gewichtet:

$$\Delta TW_i^n = \Delta T_i^n \cdot H_i^n \quad (5.13)$$

Aus der Anzahl der zuletzt auf dem langsamsten Worker ( $\max(\Delta TW_i^n)$ ) bearbeiteten Elemente und der Zeitdifferenz der Antworten des langsamsten und der mittleren Rechner im letzten Iterationsschritt kann die Anzahl der Elemente für die Migration ermittelt werden. Analog wird geprüft, wieviele Elemente der schnellste Worker ( $\min(\Delta TW_i^n)$ ) aufnehmen kann, ohne dabei selbst zu langsam zu werden (vgl. Abbildung 5.7). Ist die Anzahl der abzugebenden



**Abb. 5.7:** Ermittlung der Anzahl der Elemente für die Migration

Elemente größer, werden sukzessive die nächsten, noch relativ schnellen Worker verwendet.

Die Einbeziehung des Vektors  $\mathbf{H}$  verhindert, daß bei einer stark schwankenden Systemlast kurzzeitig frei werdende Prozessoren überlastet werden und nur sporadisch stark belastete Prozessoren zu viel Rechenlast abgeben.

Nachdem die Anzahl der Elemente bestimmt ist, muß nun ermittelt werden, welche Elemente zu welchem Worker gelangen sollen. Bei der Gebietszerlegung spielt die Optimierung der Gebiete hinsichtlich der Minimierung der Grenzlänge eine entscheidende Rolle. Bei dem hier gezeigten Verfahren (Parallelisierung auf Elementebene) wirkt sich eine disperse Aufteilung der Gebiete auch nachteilig aus, da hierdurch knotenbezogene Daten mehrfach transferiert werden müssen.

Während bei der Gebietszerlegung relativ aufwendige Algorithmen verwendet werden müssen [Kla96, LB97], kann hier mit einem einfachen Zuordnungsverfahren das Mapping der Elemente erreicht werden. Ein flexibler und vor allem schnell wirkender Lastausgleich erfordert jedoch, daß von jedem Worker zu jedem Worker Elemente transferiert werden können. Eine unregelmäßige Zuordnung würde bereits nach wenigen Migrationen zu einer völlig dispersen Datenstruktur führen.

Zu Beginn der Berechnung werden die Elemente auf die Worker jeweils mit aufsteigender Numerierung verteilt (Abbildung 5.8 oben). Worker 1 erhält also die ersten Elemente (1...), Worker  $p$  die letzten Elemente (...e). Die Zuordnung von Elementen bei der Migration erfolgt nach einem unkomplizierten Schema:

Ist der langsame Worker in der Reihenfolge vor dem schnellen Worker, so gibt er die Elemente mit der jeweils höchsten Nummer ab (vgl. Abbildung 5.8, 1. und 2. Lastausgleich). Ist die Reihenfolge der Worker umgekehrt, dann werden die Elemente mit der jeweils niedrigsten Nummer abgegeben (vgl. Abbildung 5.8, 3. Lastausgleich).

Hierdurch entstehen geringfügig überlappende Bereiche, wodurch die zusätzliche Menge der Ein- und Ausgangsdaten auf ca. 30 % anwächst. Jedoch bewirkt dieser Zuwachs nur eine minimale Verlängerung der Datentransferzeit. Bei einer völlig dispersen Datenstruktur wäre dagegen eine Vervielfachung der Datenmenge hinzunehmen, wodurch sich die Transferzeit erheblich verlängern würde.

Ein jederzeit optimaler Lastausgleich kann jedoch auch mit diesen Methoden nicht erfolgen, da das genaue Verhalten des Parallelrechners nicht zuverlässig für den Zeitraum des nächsten Iterationsschrittes abgeschätzt werden kann.

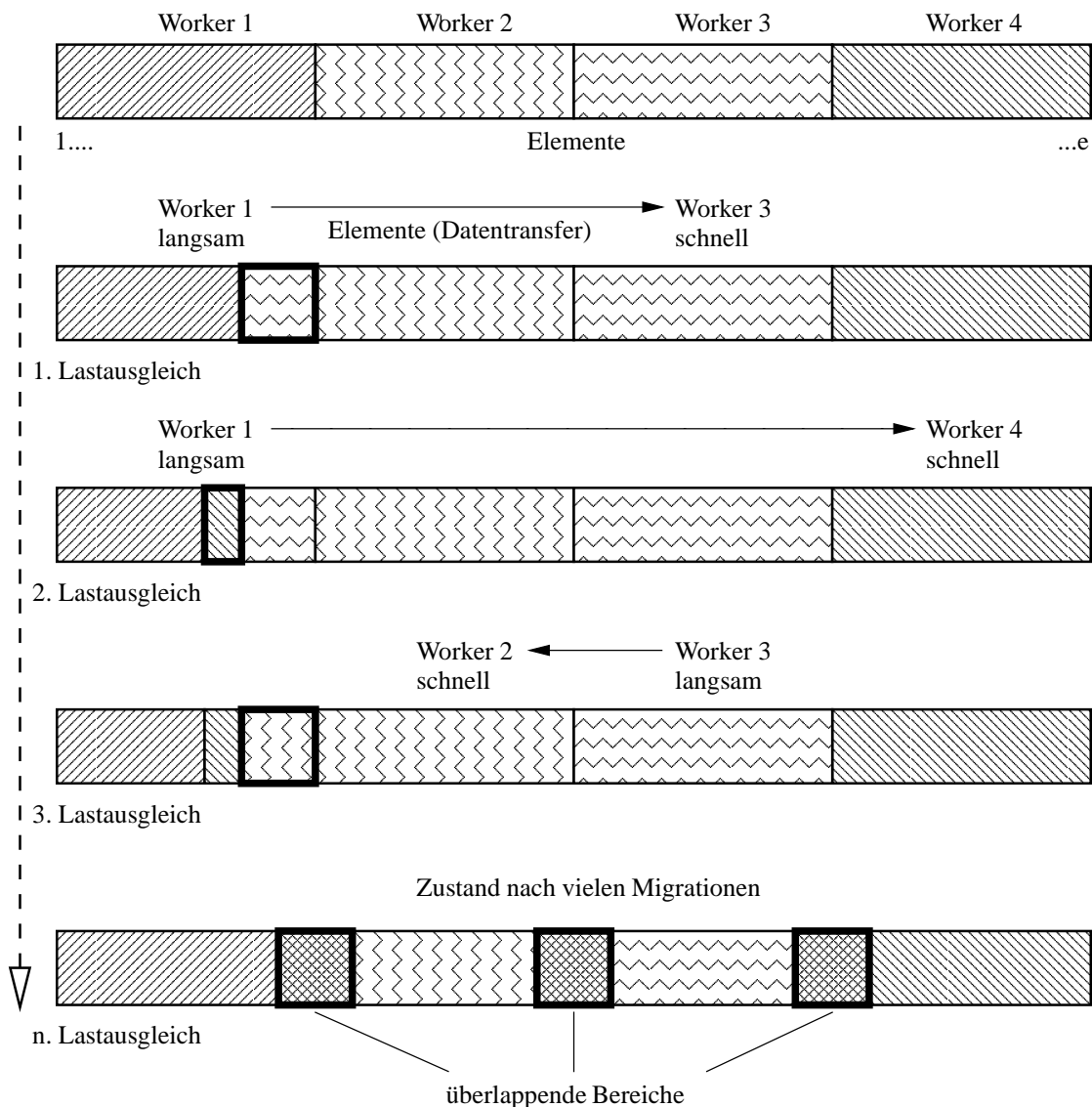
Abbildung 5.3 auf Seite 82 zeigt einen typischen Fehler: Der Worker auf dem Rechner "hooke" gibt zu Beginn des 2. Iterationsschrittes Elemente an den Rechner "navier" ab, da "hooke" bisher überlastet war. Allerdings wird "hooke" auch extern entlastet, wodurch er den 2. Iterationsschritt als erster beendet. Der Lastausgleich hat also in diesem Fall das Verhalten falsch prognostiziert.

Im gesamten Mittel kann jedoch mit Hilfe des dynamischen Lastausgleichs eine deutliche Steigerung der Leistungsfähigkeit des Parallelrechners erzielt werden, wie die folgenden Simulationsergebnisse zeigen.

### 5.3.4 Parallele Simulation – Ergebnisse

Mit numerischen Simulationen von großen FE-Systemen (vgl. Kapitel 6) wird das Verhalten sowie die Leistungsfähigkeit der parallelen Software dargestellt.





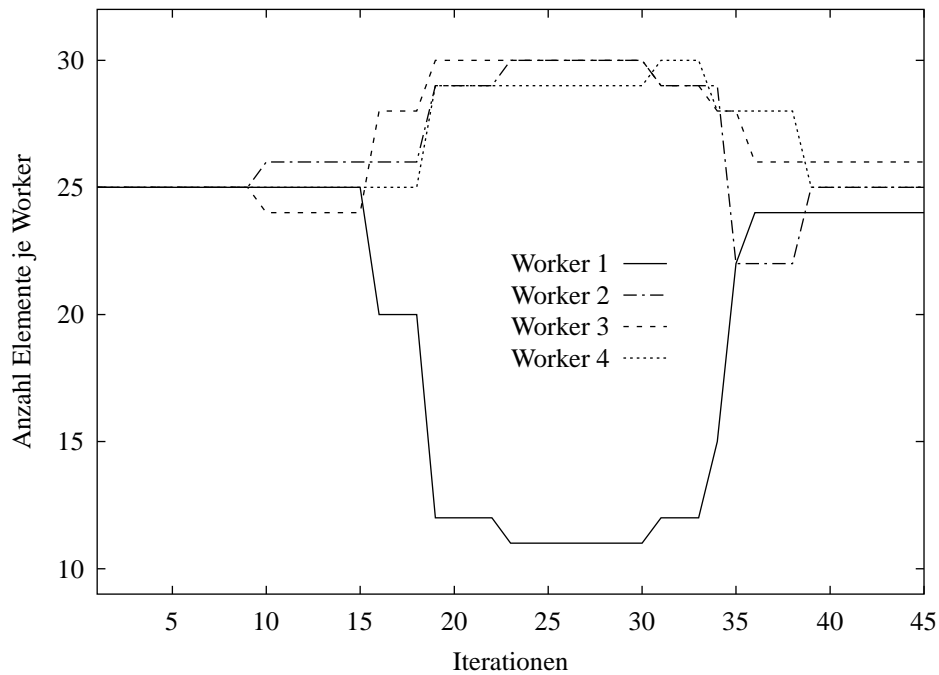
**Abb. 5.8:** Mapping der Elemente (vereinfachtes Schema)

### Dynamischer Lastausgleich

Die Funktion des Lastausgleichs kann zunächst am unbelasteten Parallelrechner getestet werden. Bei einem homogen aufgebauten Rechnersystem müssen alle Prozessoren annähernd gleich viele Elemente berechnen.

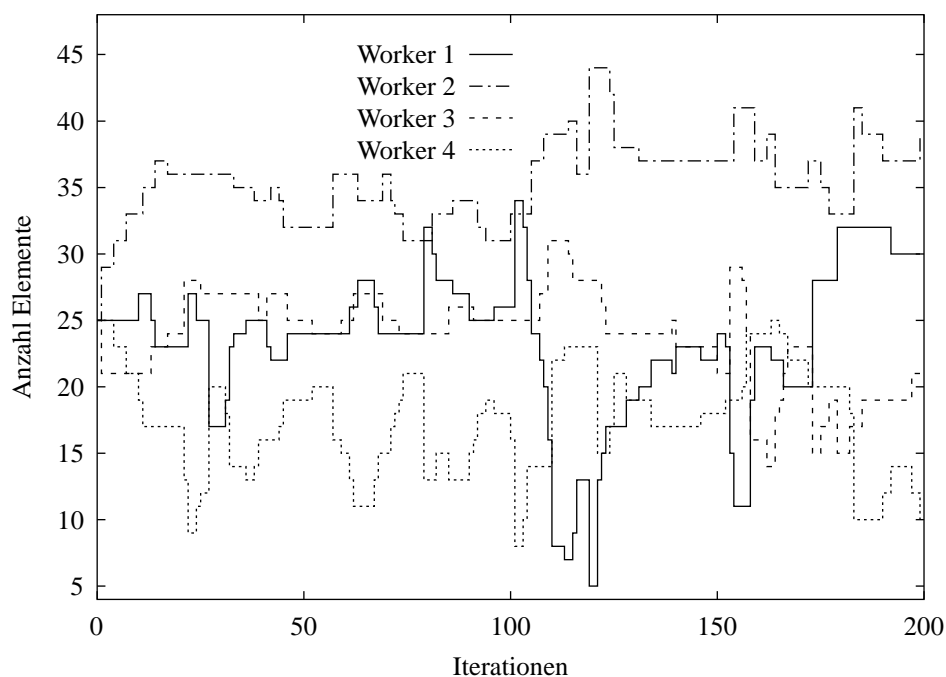
Wird nun ein Prozessor mit einem externen Prozeß gleicher Priorität belastet, ist der Worker auf diesem Rechner nur noch etwa halb so leistungsfähig wie vorher. Etwa die Hälfte der auf diesem Prozessor berechneten Elemente müssen deshalb auf die anderen Prozessoren verteilt werden (Abbildung 5.9). Dieser Vorgang benötigt einige Iterationsschritte, da das System eine gewisse Trägheit besitzt. Nach Beendigung der externen Last teilt das System dem nun relativ schnellen Prozessor sukzessive Elemente von den anderen, langsameren Prozessoren zu. Auch hier läßt sich der optimale Lastausgleich nach einigen Schritten finden.

Bei einem Workstation Cluster, der von verschiedenen Personen mit sehr unterschiedlichen



**Abb. 5.9:** Lastausgleich bei kurzzeitiger (ca. 1 min), externer Systemlast auf einem Rechner (Worker 1)

Rechenaufgaben belastet wird (z.B.: Textverarbeitung, Programmentwicklung mit kurzen und langen Testläufen), reagiert der dynamische Lastausgleich mit einem stark schwankenden Antwortprofil (vgl. Abbildung 5.10). Gut zu erkennen ist hier die sehr unterschiedliche



**Abb. 5.10:** Lastausgleich bei einem heterogenen Workstation Cluster mit unterschiedlichen externen Lasten (Zeitraum: ca. 10 min)

Zuteilung von Elementen: Worker 2 erhält teilweise mehr als 40 Elemente, während Worker 1 nur ca. 5 Elemente bearbeitet. Es ist naheliegend, daß hier Worker 1 bei einer statischen Lastverteilung (mit jeweils 25 Elementen) die Berechnung stark behindern würde, da er etwa fünfmal so viele Elemente bearbeiten müßte.

Aus der Elementverteilung ist auch die unterschiedliche Dauer externer Lasten erkennbar. So wird Worker 1 bei ca. 30 Iterationen kurz, bei 110 Iterationen relativ lange belastet. Aus dieser Abbildung ist nicht genau ablesbar, welcher Worker entlastet und welcher belastet wird, da sich jede Lastverteilung auf mehrere Rechner auswirken kann. Es ist jedoch deutlich erkennbar, daß Worker 1 durchschnittlich langsamer als Worker 2 ist. Die Ursache liegt in diesem Fall in der unterschiedlichen maximalen Prozessorleistung begründet.

### Speedup

Zur Beurteilung der Leistungsfähigkeit der parallelen Software werden Simulationen auf dem in Seite 80 beschriebenen Workstation-Cluster durchgeführt. Nach zahlreichen Simulationen kann bei dem hier verwendeten Verfahren (Parallelisierung auf Elementebene, sequentielle Gleichungslösung) festgestellt werden, daß der erreichbare Speedup nahezu unabhängig von der Problemgröße ist.

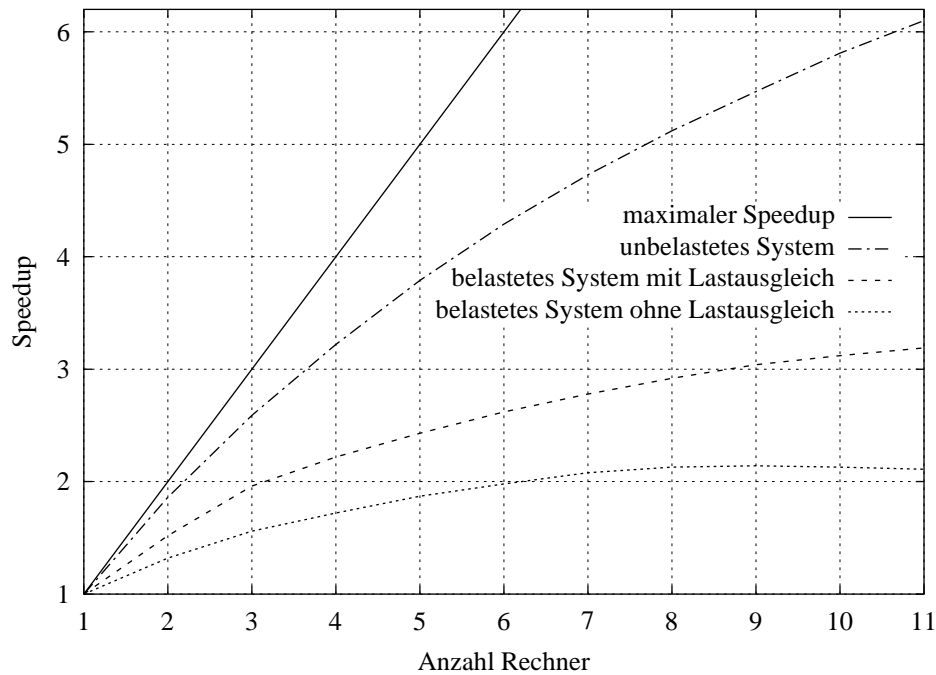
Das bedeutet, daß der scaled Speedup nach *Gustafson* (Gleichung 5.7, Seite 74) etwa gleich dem Speedup nach *Amdahl* (Gleichung 5.6, Seite 74) ist. Diese zunächst enttäuschende Feststellung ist für die hier verwendeten Workstation-Cluster unproblematisch, da *Gustafson's* scaled Speedup vor allem für massiv parallele Systeme mit sehr vielen Prozessoren interessant ist (Gleichung 5.9, Seite 74).

Die Ursache für diesen Zusammenhang liegt darin begründet, daß nicht nur der Aufwand der Elementberechnung ungefähr quadratisch zunimmt, sondern auch der Aufwand für die Gleichungslösung. Eine Ausnahme bilden nur sehr kleine Systeme, bei denen der Initialisierungsaufwand zur Kommunikation im Vergleich zum Rechenaufwand sehr hoch ist. Im Folgenden wird deshalb nur der fixed size Speedup eines großen FE-Systems betrachtet.

Da etwa 5 % des Programmcodes nicht parallelisiert sind und der zusätzliche Datentransfer berücksichtigt werden muß, kann nach *Amdahl's* Gesetz ein maximaler Speedup von ca. 10 erwartet werden. Tatsächlich wird mit 10 unbelasteten Rechnern ein Speedup von ca. 6 erreicht (Abbildung 5.11).

Die hier ermittelten Speedup's sind denen von *Rank/Rücker* et al ermittelten Werte sehr ähnlich (vgl. Systeme hoher Polynomgrade in [RKR98, RRDB99]). Aufgrund der analogen Vorgehensweise ist dies auch zu erwarten.

Bei einem System mit externen Lasten ist zu beachten, daß auch bei einem optimalen Lastausgleich nicht der maximale Speedup erreicht werden kann, da der Parallelrechner im Mittel weniger leistungsfähig ist. Sind alle Prozessoren durchschnittlich mit einem zusätzlichen Prozeß belastet, so halbiert sich der Speedup im Vergleich zum sequentiellen Programm auf einem unbelasteten Rechner. Beim "belasteten System" in Abbildung 5.11 ist im Mittel jeder Worker mit einem externen Prozeß belastet, der Master ist unbelastet.



**Abb. 5.11:** Speedup bei einem homogenen Workstation Cluster

Wird der automatische Lastausgleich nicht verwendet, blockieren extern belastete Prozessoren das gesamte System und der erreichbare Speedup ist äußerst unbefriedigend. Interessant ist hierbei, daß der Speedup nach einem Maximum bei ca. 8 Rechnern mit weiter zunehmender Prozessoranzahl wieder geringfügig sinkt. Die langsamen, hoch belasteten Prozessoren blockieren die Simulation also zunehmend.

Ein besonderes Problem bei einer Master-Worker Konfiguration ergibt sich, wenn der Rechner, auf dem das Master-Modul abgearbeitet wird, von einem externen Prozeß belastet wird. Hierdurch verlängert sich die Bearbeitungszeit für den sequentiellen Anteil, wodurch der erreichbare Speedup überproportional sinkt. Eine minimale Rechenzeit ergibt sich deshalb nur, wenn der Master auf einem unbelasteten Prozessor läuft. Auf einem heterogenen Rechner-Cluster ist es daher sinnvoll, das Master-Modul auf einem leistungsstarken, wenig belasteten Rechner zu starten und die Worker auf die übrigen Rechner zu verteilen.

Mit dieser Vorgehensweise kann also für besondere Finite-Element-Methoden eine beachtliche Reduzierung der Rechenzeit erreicht werden. Das System eignet sich insbesondere für den Einsatz auf preiswerten Workstation-Clustern.

# Kapitel 6

## Anwendungsbeispiel: Flüssigkeitsgefüllte Behälter

In diesem Kapitel wird mit Hilfe von einigen exemplarischen Berechnungen die Leistungsfähigkeit der in dieser Arbeit diskutierten Algorithmen gezeigt. Die methodischen Gesichtspunkte werden durch vergleichende Parameterstudien von unverankerten, flüssigkeitsgefüllten Behältern unter Erdbebeneinwirkung ergänzt. Außerdem wird die synthetische Generierung von Lastmodellen für die numerische Simulation von nichtlinearen Strukturen unter Erdbebeneinwirkung gezeigt.

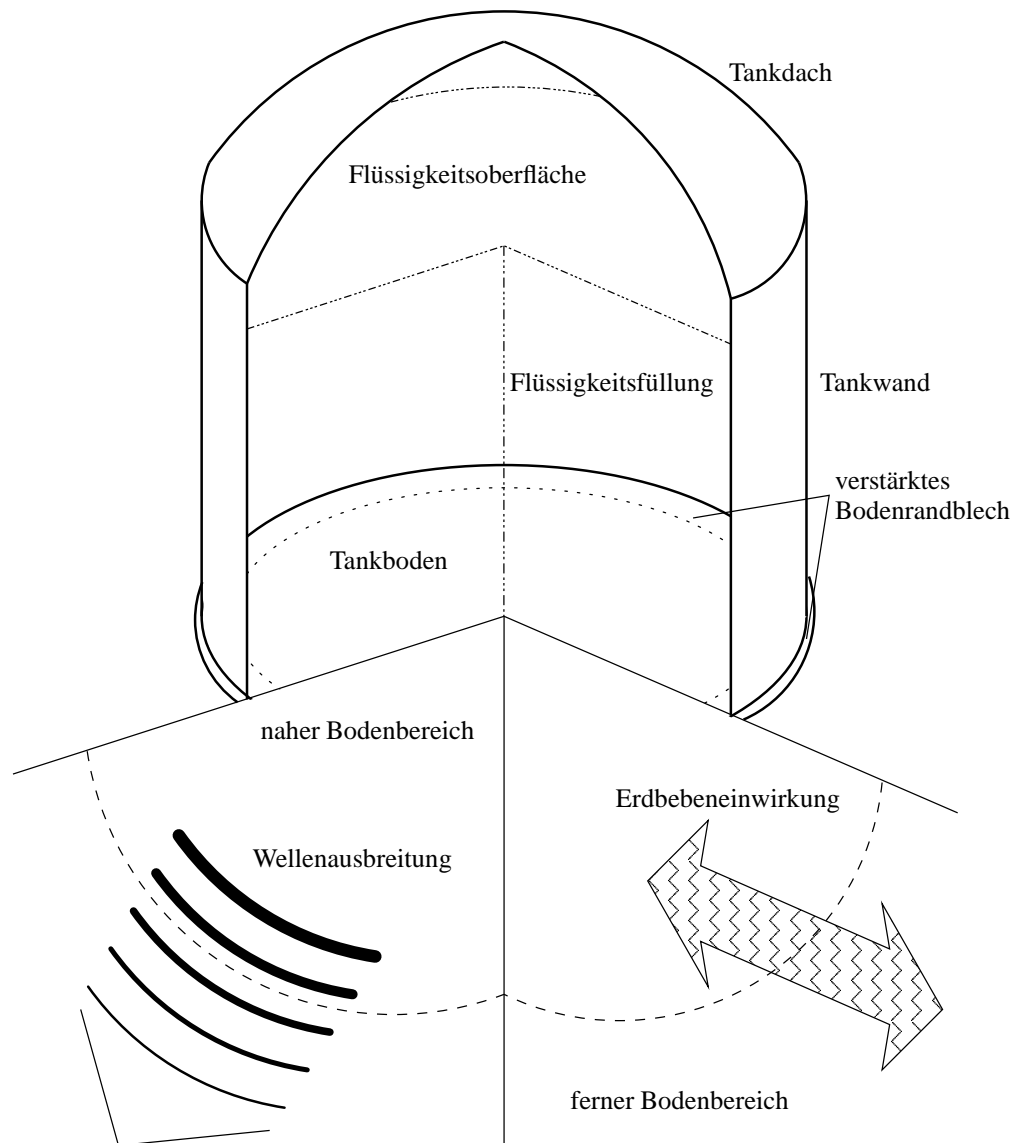
Auch wenn die gewählte Art der Beispiele sehr speziell erscheinen mag, können die dargestellten Methoden und numerischen Effekte auch auf viele andere Bereiche übertragen werden:

- Bei der numerischen Simulation von Staumauern und ihrer Wechselwirkung mit der Gründung sowie dem gespeicherten Wasser unter Erdbebeneinwirkung werden große FE-Berechnungen im Zeitbereich mit unterschiedlichen Elementeigenschaften für die Flüssigkeit, das Bauwerk sowie die Gründung durchgeführt [FB96, WB95, YKV99].
- Die geometrisch und physikalisch nichtlineare Untersuchung von dynamisch beanspruchten Strukturen wird zunehmend an komplexen Strukturen (z.B. Schalenträgerwerke) angewandt [dBM95, Mey98, San96].
- Die detaillierte Analyse von Erdbebenschäden und die zuverlässige Sicherheitsabschätzung komplexer Strukturen erfordert neben der Bereitstellung von natürlichen und künstlich generierten Erdbebenaufzeichnungen eine effiziente dynamische Strukturanalyse [Bac95, BW99, MKCB99, MPA97, PSS90].
- Gerade bei großen FE-Berechnungen bieten sich parallele Konzepte zur Minimierung der Rechenzeiten an [FR94, FES98, Läm97a].

### 6.1 Systembeschreibung Behälter

Die hier untersuchten Behälter dienen zur Speicherung von Flüssigkeiten (Wasser, Getränke und chemische Produkte). Als Werkstoff wird meist Stahl, in manchen Fällen auch Leichtmetall verwendet. Zur Kostenminimierung wird oft – gerade bei Großbehältern der petrochemischen Industrie – auf eine Verankerung der Tankwand verzichtet [Ham80, HS98]. Während

eines Erdbebens kann der Behälter deshalb partiell abheben [MN90]. Dies geschieht auch bei verankerten Tanks, wenn die vorgesehene Verankerung den enormen Kräften einer Erdbebeneinwirkung nicht Stand hält und reißt [FRS90, Tem95].



**Abb. 6.1:** Modell: Flüssigkeitsgefüllter Behälter unter Erdbebeneinwirkung

Abbildung 6.1 zeigt schematisch den komplexen Aufbau der zu untersuchenden Struktur:

- Die zylinderförmige Tankwand besteht aus mehreren Blechringen (*Schüssen*), deren Dicke entsprechend der statischen Belastung aus dem Flüssigkeitsdruck abgestuft ist.
- Das Bodenblech ist zur Aufnahme der Biegestörung aus dem Zylinder am Rand verstärkt ausgebildet. Der übrige Bereich hat lediglich eine Dichtungsfunktion und besteht deshalb aus einem relativ dünnen Blech.
- Das meist kugelförmige Dach besteht aus Längs- und Querträgern sowie einer dünnen Dachhaut. Bei der Berechnung kann eine ideelle Blechdicke angenommen werden.

- Bei Behältern, die leichtflüchtige Stoffe enthalten (z.B. Benzin oder Gase), befindet sich eine *Schwimmdecke* mit einer Membrandichtung im Behälter [DIN79, KLMW98]. Dieses Bauteil hat keine statische Tragwirkung und kann deshalb in der numerischen Simulation vernachlässigt werden.
- Der Bereich direkt unter dem Behälter besteht entweder aus einer verdichteten Bodenschicht oder aus Beton. Bei einer dynamischen Beanspruchung können insbesondere bei einer verdichteten Bodenschicht physikalisch nichtlineare Effekte auftreten, die in der Berechnung berücksichtigt werden sollten.
- Der entfernte Bodenbereich verhält sich annähernd linear elastisch. Bei der dynamischen Beanspruchung des Behälters entstehen Wellen, die in den Halbraum abgestrahlt werden. Dies führt zu einem Energieverlust des untersuchten Systems und muß deshalb in der numerischen Simulation berücksichtigt werden.
- Die Belastung des Tanks resultiert im wesentlichen aus der Flüssigkeitsfüllung, die mit dem hydrostatischen Flüssigkeitsdruck auf den Behälter wirkt. Bei der dynamischen Simulation muß neben der trägen Masse auch das Schwappen der Flüssigkeit einbezogen werden.

### 6.1.1 Schadensfälle

Nach starken Erdbeben konnten in der Vergangenheit zahlreiche Schadensfälle beobachtet werden. Abbildung 6.2 zeigt exemplarisch zwei Fälle aus Kalifornien.

Das linke Foto stammt von einem unverankerten Behälter\*, der beim *San Fernando Erdbeben* am 9. Februar 1971 beschädigt wurde [Ste71]. Gut zu erkennen ist das überstehende Bodenrandblech sowie das Betonringfundament. Durch das Erdbeben ist am unteren Behälterrind eine ausgeprägte Wulstbeule entstanden, die wegen ihrer charakteristischen Form auch als "Elefantenfuß" bezeichnet wird. Im Bild rechts unten ist anhand der abgerissenen Kabel deutlich zu sehen, daß der Behälter während des Bebens erheblich vom Fundament abgehoben hat.

Rechts oben ist in Abbildung 6.2 ein 1979 im *San Bernardino County*† errichteter Wasserbehälter zu sehen, der während eines sehr starken Bebens im Juni 1992 beschädigt wurde. Während des Bebens hat der Tank von der Gründung abgehoben, was an dem herausgerissenen und verbogenen Anschlußrohr zu erkennen ist. Abgesehen von dem beschädigten Rohr ist der Tank dicht geblieben. Allerdings ist auch er durch eine ausgeprägte Wulstbeule am Fuß unbrauchbar geworden. Der Wassertank hatte relativ kleine Abmessungen: Eine Höhe von 7.3 m, einen Radius von 7.2 m und eine Blechdicke von ca. 3.2 mm [EQE92].

---

\* Abmessungen und Inhalt unbekannt

† Südkalifornien, ca. 50 km nördlich von Palm Springs





Abb. 6.2: Schadensfälle unverankerter Flüssigkeitsbehälter nach einem Erdbeben

### 6.1.2 Bemessung - Überblick

Für die statische Bemessung flüssigkeitsgefüllter Behälter ist die Beanspruchung aus Eigengewicht, aus dem Flüssigkeitsdruck sowie aus äußeren Wind- und Schneelasten maßgebend. Die Berechnung kann z.B. nach DIN 4119 Teil 2 [DIN79] erfolgen.

Eine dynamische Beanspruchung kann durch die Druckwelle einer Explosion entstehen oder aus einem Erdbeben resultieren. Seit vielen Jahren wird an verschiedenen Verfahren zur Bemessung flüssigkeitsgefüllter Behälter unter Erdbebeneinwirkung gearbeitet. Meist wird hierbei von einer Verankerung der Tankwand-Boden-Ecke und starrem oder linear-elastischem Baugrundverhalten ausgegangen [MGC82, EC893, SA95].

Neben verschiedenen analytischen Ansätzen [FS88, YV96] spielen zunehmend auch Finite-Element-Modelle eine wichtige Rolle [BP87, DS96, HWC<sup>+</sup>96, Sch96b, Sch90].

Aber auch unverankerte Behälter sind in der Vergangenheit vielfältig untersucht worden [Fis81, FRS90, Man86, Pee88, Sch90, Tem95]. Manche Autoren verwenden stark vereinfachte Modelle (z.B. Balken-Modelle) [MV94, Mal96] oder experimentelle Untersuchungen [HH79, NC82].

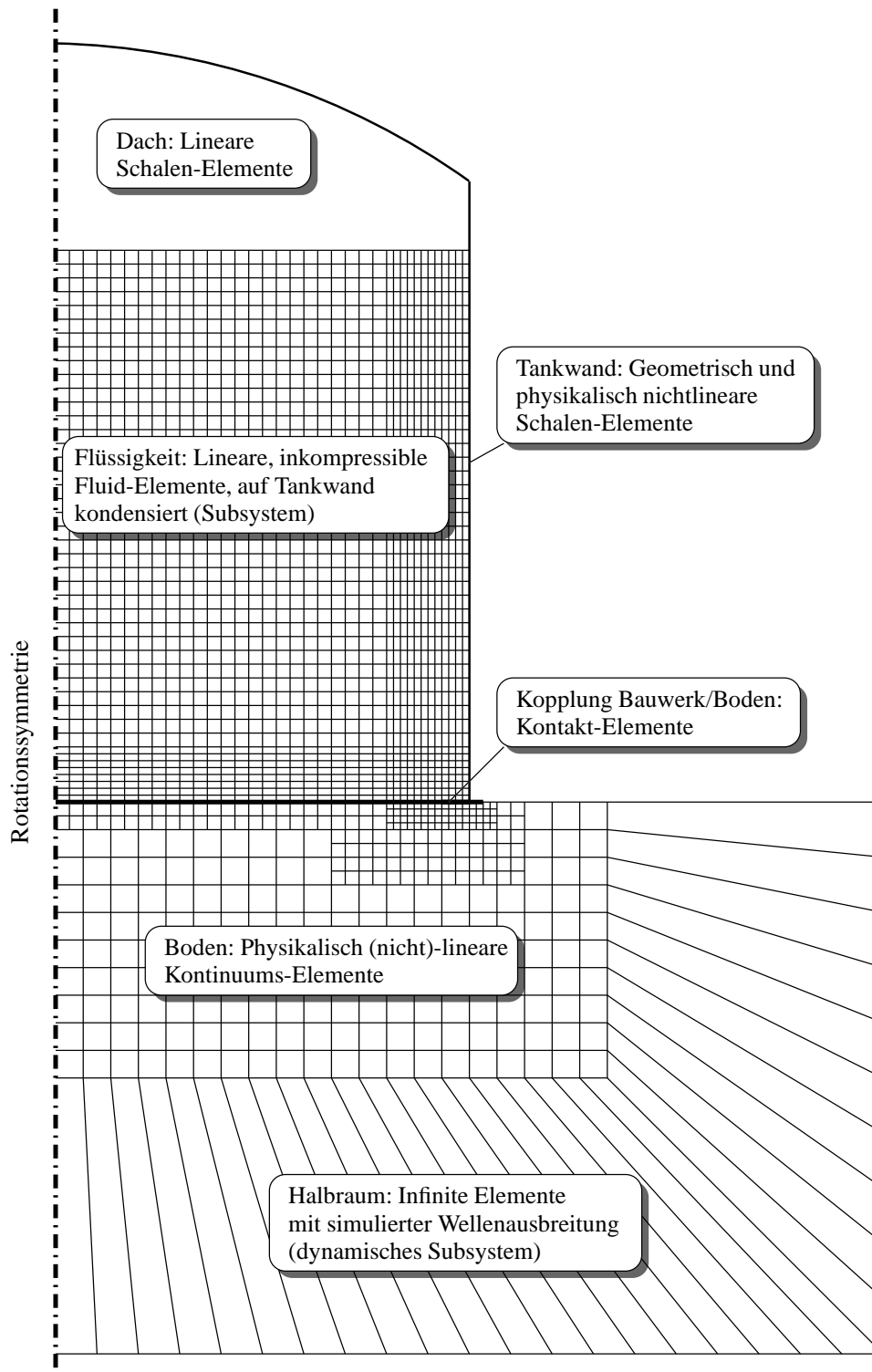


Im Mittelpunkt der nachfolgenden Ausführungen steht jedoch nicht die Bemessung der unverankerten Behälter, sondern das methodische Vorgehen bei der numerischen Simulation. Neben dem speziellen Finite-Element-Modell wird auch die Ermittlung der Erdbebeneinwirkung diskutiert. Die exemplarischen Berechnungsbeispiele dienen zur Veranschaulichung des Tragverhaltens sowie zur Darstellung von besonderen Effekten der numerischen Simulation.

## 6.2 Finite-Element-Modell

Die spezielle Methodik der in dieser Arbeit verwendeten Ring-Elemente wurde bereits in Abschnitt 2.2.3 skizziert. Hier folgt nun die Beschreibung der mechanischen Elementeigenschaften. Diese Eigenschaften sind weitgehend unabhängig von der verwendeten Elementgeometrie und können auch auf andere Diskretisierungen übertragen werden.

- Das **Schalen-Element** basiert auf den Annahmen einer dünnen Schale mit kleinen Verzerrungen, großen Verschiebungen und moderaten Rotationen. Die Querschnitte bleiben eben (senkrecht zur Schalenmittelfläche) und weisen keine Normalspannungen senkrecht zur Schalenmittelfläche auf. Es wird isotropes, elasto-plastisches Materialverhalten angenommen ( $J_2$ -Fließtheorie kombiniert mit kinematisch-isotroper Verfestigung). Statt der Verwendung von Ansatzfunktionen wird in Meridianrichtung über eine numerische Integration des Differentialgleichungssystems 1. Ordnung eine asymptotisch exakte Steifigkeitsmatrix erzeugt [Wun67, WCO85, WOSL87].
- Beim **Fluid-Element** wird angenommen, daß die Flüssigkeit reibungsfrei, inkompressibel und drehungsfrei ist. Die Masse der Flüssigkeit wird auf die Tankwand kondensiert (*added-mass*). Darüberhinaus wird die Bewegung der freien Oberfläche (Schwappen) über einen konvektiven Druckanteil erfaßt und zusätzlich auf die Behälterwand aufgebracht [WST92, WSG89].
- Zwischen Tankboden und Gründung werden **Kontakt-Elemente** verwendet. Die Formulierung beruht auf einem Kontinuums-Element mit einem hyperelastischen Stoffgesetz für das Öffnen und Schließen, kombiniert mit einer auf das Gleiten beschränkten inkrementellen Plastizität [Tem95, WT95, WCST95].
- Der nahe Bodenbereich (Gründung) wird durch **Kontinuums-Elemente** mit variabler Knotenanzahl (4-9) diskretisiert. Das Element kann physikalisch lineares und nichtlineares Verhalten erfassen. Die Besonderheit liegt in der aufwendigen Einbeziehung der Stoffeigenschaften bei Ent- und Wiederbelastung mit Hilfe eines Mehrflächenmodells (kombinierte kinematisch-isotrope Verfestigung mit Memory-Funktion). Zudem kann Scherversagen und Bodenverflüssigung berücksichtigt werden [CPWJ91, WCST95].
- Der entfernte Bodenbereich (Halbraum) des Systems wird durch **Infinite Elemente** erfaßt. Sie basieren auf einer asymptotischen Formulierung und verfügen über spezielle Dämpfungseigenschaften (Dämpfungs-Layer), damit die Wellenausbreitung verschiedener Wellenarten (Druck- und Scherwellen) möglichst realistisch simuliert werden kann und Reflexionen an einem künstlichen Rand verhindert werden [WCST95, WSST91].



**Abb. 6.3:** Finite-Element-Modell eines Behälters unter Ausnutzung der Rotations-symmetrie (semi-analytische Ring-Elemente)

Entsprechend dem physikalischen Modell (Behälter, Flüssigkeit, Boden) wird mit den beschriebenen Elementen ein möglichst genaues mathematisches Modell erzeugt. Unter Ausnutzung der Rotationssymmetrie kann die räumliche Diskretisierung an einer halben, vertikalen Schnittfläche entlang der Rotationsachse erfolgen (Abbildung 6.3).

Die Anzahl der Fourier-Reihenglieder wird entsprechend der gewünschten Genauigkeit und der Art der Beanspruchung festgelegt. Besonders bei lokalen Effekten, wie sie bei unverankerten Behältern auftreten, müssen relativ viele Reihenglieder angesetzt werden (etwa 15).

Der Bereich der Verbindung zwischen Tankboden und Tankwand kann a priori adaptiv verfeinert werden. Hierzu muß auch die Flüssigkeit und der Bodenbereich feiner diskretisiert werden. Die Anzahl der Harmonischen bleibt jedoch über die gesamte Struktur konstant.

Zur Reduzierung des Rechenaufwandes bei der iterativen Ermittlung des Gleichgewichts können lineare Bereiche mit Hilfe der dynamischen Substrukturtechnik behandelt werden. Dieses Vorgehen wird beim entfernten Bodenbereich sowie bei der Flüssigkeit angewandt [WST92].

### 6.2.1 Kopplung unterschiedlicher Elemente

Bei der Kopplung unterschiedlicher Elemente müssen neben der gewünschten physikalischen Wirkung auch numerische Effekte berücksichtigt werden.

Die Kopplung von Fluid- und Schalen-Elementen wurde bereits im vorhergehenden Abschnitt 6.2 skizziert. Sie wird ausführlich in [WSG89, WST92, Tem95] behandelt. Hier wird nun die Kopplung der Schalen-Elemente (Tankboden) mit den Kontakt-Elementen diskutiert.

Diese Kopplung stellt eine Verbindung sehr unterschiedlicher Elemente dar: Das Schalen-Element kann in radialer und vertikaler Richtung aufgrund der speziellen Formulierung mit Übertragungsmatrizen sehr komplexe Zustände darstellen, während mit dem Kontakt- bzw. Bodenelement nur lineare oder quadratische Ansätze gewählt werden können. Darüberhinaus verfügt das Schalen-Element über Rotations- und Translationsfreiheitsgrade (vgl. Abbildung 2.2, Seite 11), die Kontinuums-Elemente nur über Translationsfreiheitsgrade.

In Umfangsrichtung ist infolge der gleichen Anzahl von Fourier-Reihengliedern eine homogene Verbindung vorhanden, weshalb sich eine kinematische Kopplung der Translationen anbietet. In der vertikalen Schnittebene können zwei unterschiedliche Diskretisierungen mit unterschiedlichen Elementansätzen verwendet werden (vgl. Abbildung 6.4 und 6.5).

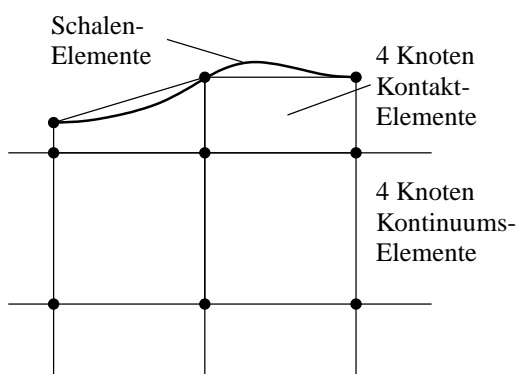


Abb. 6.4: Kopplung mit linearen Kontakt-Elementen

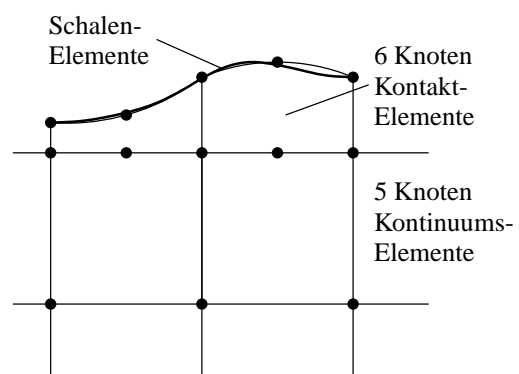


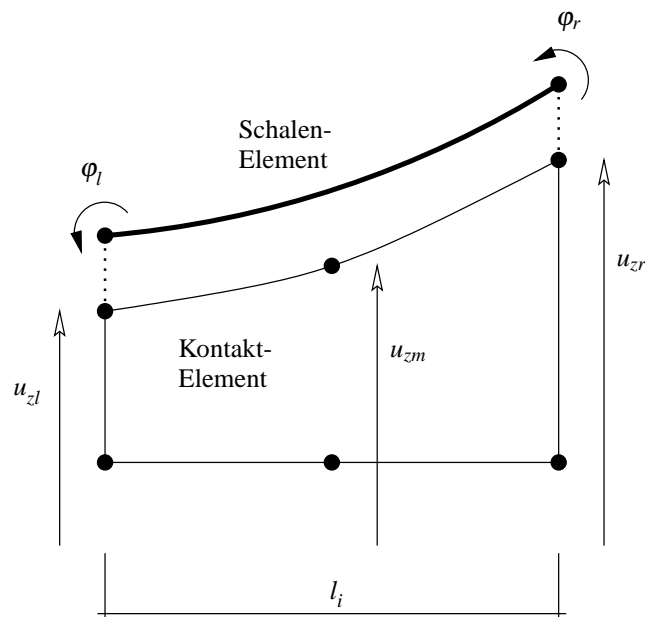
Abb. 6.5: Kopplung mit quadratischen Kontakt-Elementen

Bei Verwendung von Kontinuums-Elementen mit linearen Ansätzen (Abbildung 6.4) können nur die Translationen gekoppelt werden, die Rotationsfreiheitsgrade der Schalen-Elemente sind unabhängig von der Verformung der Kontakt-Elemente. Dadurch können numerische Instabilitäten entstehen, die sogar zum Rechenabbruch infolge Divergenz führen können [WR<sup>+</sup>98]. Hier spielt die Inkompatibilität der Elemente eine wichtige Rolle, da hierdurch bekanntlich eine konvergente Lösung fraglich ist [Bat86].

Eine Elementformulierung mit quadratischen Verschiebungsansätzen (vgl. Abbildung 6.5) führt ebenfalls zu einer inkompatiblen Kopplung. Hier stellt sich zunächst die Frage, wie der obere Mittelknoten der Kontakt-Elemente sinnvoll mit den Schalen-Elementen verbunden werden kann\*.

Eine Verdopplung der Anzahl der Schalen-Elemente hätte zur Folge, daß ein Schalen-Element jeweils an einen Rand- und einen Mittel-Knoten eines Kontakt-Elementes gekoppelt wäre. Da sich die Steifigkeiten von Rand- und Mittel-Knoten unterscheiden können (abhängig von der verwendeten numerischen Integration), ist dieses Vorgehen nicht empfehlenswert.

Wird dagegen die gleiche Anzahl von Schalen- und Kontakt-Elementen verwendet, kann der Mittel-Knoten mit Hilfe von *Multi-Point-Constraints* an die Randknoten gekoppelt werden. Entsprechend der Superposition von linearen und quadratischen Ansatzfunktionen ergibt sich



**Abb. 6.6:** Kinematische Kopplung von Schalen- und Kontakt-Elementen

für die Darstellung in Abbildung 6.6 folgender Ansatz für die vertikale Verschiebung des Mittelknotens:

$$u_{zm} = \frac{1}{2}u_{zl} + \frac{1}{2}u_{zr} + \frac{l_i}{8}\varphi_l - \frac{l_i}{8}\varphi_r \quad (6.1)$$

\*Der untere Knoten der Kontakt-Elemente kann sehr gut mit Hilfe der variablen Knoten-Anzahl der Boden-Elemente eingebunden werden (kompatible Ansätze)

Der Hauptvorteil dieses Ansatzes liegt in der Kopplung der Verdrehungen des Schalen-Elementes mit den Verschiebungen des Kontakt-Elementes. Hierdurch wird nicht nur das mechanische Tragverhalten gut abgebildet, sondern auch das numerische Verhalten verbessert, was sich in einer verringerten Anzahl von Iterationsschritten bemerkbar macht (ca. 30 % weniger als in der Diskretisierung mit linearen Ansätzen).

Eine in diesem Zusammenhang durchgeführte Parameterstudie (hier nicht dargestellt) zum Einfluß der numerischen Integration (*Gauß* oder *Newton-Cotes*) bestätigten die Ergebnisse, die bereits *Temme* für das Element ohne kinematische Kopplung ermittelte [Tem95, WT95]: Die 3-Punkt *Newton-Cotes* Integration erweist sich bei den hier verwendeten Kontakt-Elementen als die numerisch beste Variante.

## 6.3 Dynamische Einwirkung: Erdbeben

Die dynamischen Einwirkungen auf nichtlineare Strukturen können unterschiedlicher Art sein. Meist handelt es sich um Lasten aus Wind- oder Erdbebenanregung, sowie um harmonische Einwirkungen (z.B. aus Maschinenschwingungen). Explosionslasten stellen sehr kurzzeitige Einwirkungen dar und erfordern besondere Berechnungsverfahren, die außerhalb des Themenbereichs dieser Arbeit liegen.

Eine umfassende Abhandlung über *stochastische Einwirkungen* (insbesondere Windlasten) findet man z.B. in [Wau99]. Im Rahmen der vorliegenden Arbeit werden exemplarisch Erdbeben als Einwirkungsart gewählt. Die Entstehung von Erdbeben sowie die Ausbreitung verschiedener Wellenarten in der Erdoberfläche wird in der Literatur ausführlich beschrieben (z.B.: [Bac95, Bol95, CP93]). Für die Berechnung einer lokal eng begrenzten Struktur kann die Erdbebeneinwirkung als dreidimensionale Fußpunktbeschleunigung angenommen werden. Die zentrale Frage ist folglich, durch welche charakteristischen Kriterien eine derartige Einwirkung beschrieben wird.

Bei der Analyse linear schwingender Systeme kann das Antwortspektrum als Beschreibung der maximalen Antwort verschiedener Einmassenschwinger in Abhängigkeit der jeweiligen Eigenfrequenz dienen. Mit Hilfe einer Modalanalyse kann das charakteristische Strukturverhalten gewonnen werden. Eine geeignete Überlagerung der Eigenformen aus der Modalanalyse und dem Antwortspektrum liefert die maximale Beanspruchung der Struktur [Bac95, CP93].

Nichtlineare Systeme erfordern eine Berechnung im Zeitbereich. Für die Beschreibung der Einwirkung sind deshalb Zeitverläufe der Bodenbeschleunigung notwendig. Hierfür können reale Erdbebenaufzeichnungen verwendet werden, die für manche Gebiete in großer Zahl zur Verfügung stehen\*. Eine Anwendung dieser Daten auf andere Gebiete ist äußerst fragwürdig, da nahezu alle wesentlichen Parameter von der speziellen geologischen Formation des jeweiligen Standortes abhängen [Bol95]. Bei genauer Kenntnis der Untergrundverhältnisse eines Standortes können aber auch synthetische Beben generiert werden [Bac95, CP93].

---

\*z.B. für Kalifornien/USA in [NCE98]. Daten weltweiter Standorte sind teilweise in [FS90] verfügbar

Für die Beurteilung des Schadenspotentials eines Erdbebens gibt es mehrere charakteristische Parameter [Bac95, EM99]:

- Die maximale horizontale oder vertikale Amplitude der **Bodenbeschleunigung**. Die horizontale Komponente beträgt bei einem mittelstarken Beben ca. 1,5 - 3,0  $\frac{m}{s^2}$ , die vertikale Komponente etwa  $\frac{2}{3}$  der horizontalen Komponente. Bei sehr starken Beben können Horizontalbeschleunigungen von bis zu 18  $\frac{m}{s^2}$  auftreten (Northridge, Kalifornien 1994).
- Der **Frequenzbereich** der Beschleunigung reicht von 0,1 Hz bis etwa 30 Hz. Meist wird der Frequenzbereich als Antwortspektrum angegeben (fouriertransformierte Antwort eines Einmassenschwingers variabler Eigenfrequenz). Die Form des Spektrums hängt stark von der Art des Untergrundes ab. Bei steifen Böden dominieren hochfrequente Anteile, während bei weichen Böden die niederfrequenten Schwingungen dominieren.
- Mit der **Arias-Intensität**  $I_0$  wird eine energieäquivalente Größe eingeführt:

$$I_0 = \int_{t=0}^{t=t_e} (\ddot{x}_g(t))^2 dt, \quad (6.2)$$

wobei  $\ddot{x}_g(t)$  die Bodenbeschleunigung in Abhängigkeit von der Zeit darstellt.

- Aus dem **Husid-Diagramm** (bzw. -Funktion) läßt sich die normierte Akkumulation der Energie einer Erdbebenbeschleunigung darstellen:

$$H(t) = \frac{\int_{t=0}^t (\ddot{x}_g(t))^2 dt}{I_0} \quad (6.3)$$

- Die **Dauer der Starkbebenphase**  $T_s$  wird meist als Zeitspanne zwischen der 5% und der 95% Ordinate des *Husid*-Diagramms definiert. Diese Spanne beträgt meist 5 - 20 s, kann aber auch über eine Minute betragen.

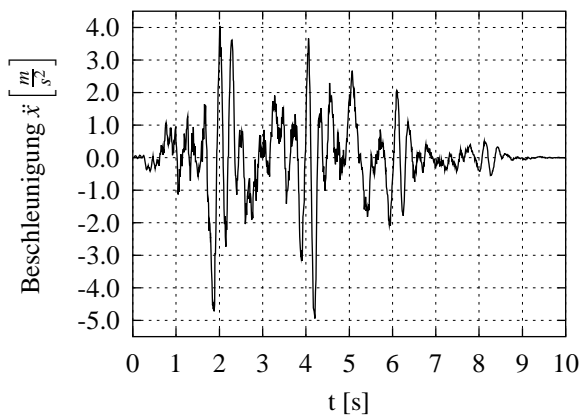
Aus historischen Aufzeichnungen lassen sich für viele Standorte mit Hilfe der Wahrscheinlichkeitstheorie Erdbeben einer bestimmten Intensität vorhersagen. Entsprechend der geologischen Formation des zu untersuchenden Standortes können synthetische Zeitverläufe der Bodenbeschleunigung generiert werden [CP93, CP97, EC893, Kle88].

Die Erzeugung spektrumskonformer Zeitverläufe ist nur für linear-elastische Strukturen sinnvoll, da diese im Vergleich zu natürlichen Erdbeben zu energiereich sind [Bac95]. Deshalb können für nichtlineare Berechnungen nur synthetische Beben verwendet werden, die sich am charakteristischen Spektrum der jeweiligen Bodenart orientieren, diese aber nicht genau abbilden. Charakteristische Spektren werden z.B. in [EC893, HRS91, Hos87] angegeben.

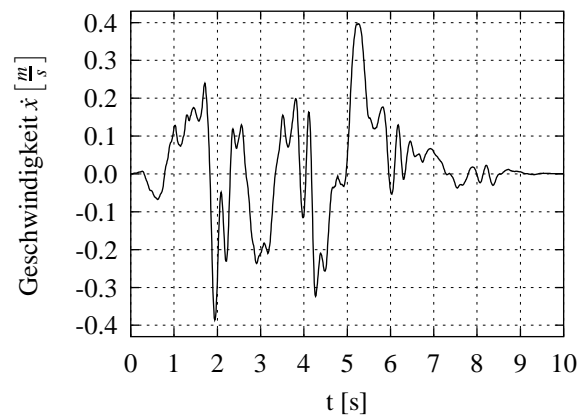
Exemplarisch wird nun ein Zeitverlauf der horizontalen Bodenbeschleunigung dargestellt, der mit Hilfe des in [Kle88] beschriebenen Verfahrens generiert wurde. Aus den Eingangsgrößen Intensität und Bodenklasse wird mit Hilfe eines stationären Zufallsprozesses ein für europäische Standorte charakteristischer Zeitverlauf erzeugt. Das künstliche Erbeben entspricht

einer *Arias-Intensität* von  $I_0 = 1.99 \left[ \frac{m^2}{s^3} \right]$ . Auf der MSK - Skala (nach *Medvedev-Sponheuer-Karnik* [Bac95]) entspricht dies wiederum einem Intensitätsgrad von etwa VIII-IX [Kle88] und ist damit einem starken Beben mit großen Zerstörungen zuzuordnen. Der Frequenzbereich ist entsprechend einer weichen Bodenklasse (A) gewählt\*.

Der Zeitverlauf der Bodenbeschleunigung (Abbildung 6.7) wird bei der numerischen Simulation als Fußpunkterregung angesetzt. Deutlich ist das starke Anwachsen der Amplituden



**Abb. 6.7:** Synthet. Beben: Zeitverlauf der Bodenbeschleunigung



**Abb. 6.8:** Synthet. Beben: Zeitverlauf der Bodengeschwindigkeit

am Anfang und das langsamere Abklingen zum Ende des Zeitverlaufs zu erkennen. Aus der Integration der Bodenbeschleunigung erhält man den Verlauf der Geschwindigkeit (Abbildung 6.8). Damit am Ende des Zeitverlaufs die Geschwindigkeit wieder zu Null wird, muß der Beschleunigungsverlauf meist mit einer *Basislinienkorrektur* verbessert werden [M+88].

Der Zuwachs der zugeführten Energie (*Arias-Intensität*) über die Zeit wird mit dem *Husid-Diagramm* grafisch dargestellt (Abbildung 6.9). Hier läßt sich auch die Dauer der Starkbebenphase mit Hilfe der 5% und der 95%-Linien ermitteln. Diese ergibt sich in dem gewählten Beispiel zu  $T_s \approx 6.0 - 1.8 = 4.2 \text{ s}$ .

Der Zeitverlauf der Beschleunigung wird zur besseren Beurteilung in ein Antwortspektrum für unterschiedliche modale Dämpfungen transformiert (Abbildung 6.10).

Falls eine Eigenfrequenz der zu untersuchenden Struktur in einem Bereich geringer Spektralbeschleunigung liegt, wird sie kaum angeregt. Beim Spektrum in Abbildung 6.10 sind beispielsweise Frequenzen im Bereich bei etwa 1.5 Hz und 10 Hz unterrepräsentiert. Deshalb ist ein einzelner Zeitverlauf wenig aussagekräftig. Für die nichtlineare Berechnung müssen mehrere Beben mit gleichen Parametern generiert werden, damit möglichst alle maßgebenden Frequenzbereiche erfaßt werden bzw. damit das elastische Bemessungsspektrum als Einhüllende der Antwortspektren möglichst lückenlos erfaßt wird. In [EC893] werden hierfür mindestens 5 Zeitverläufe gefordert.

Weitergehende Modelle berücksichtigen, daß der Zeitverlauf der Bodenbeschleunigung eigentlich ein instationärer Prozeß ist. Während der Starkbebenphase ist meist eine deutliche

\*entspricht im Eurocode 8 der Klasse C [EC893]



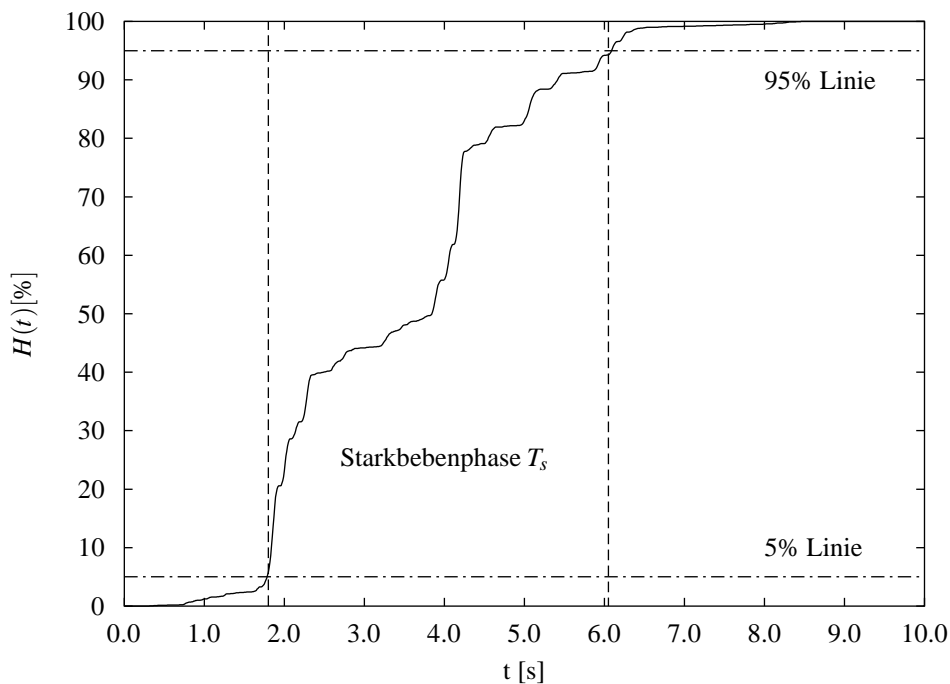


Abb. 6.9: Synthet. Beben: Husid-Diagramm und Dauer der Starkbebenphase

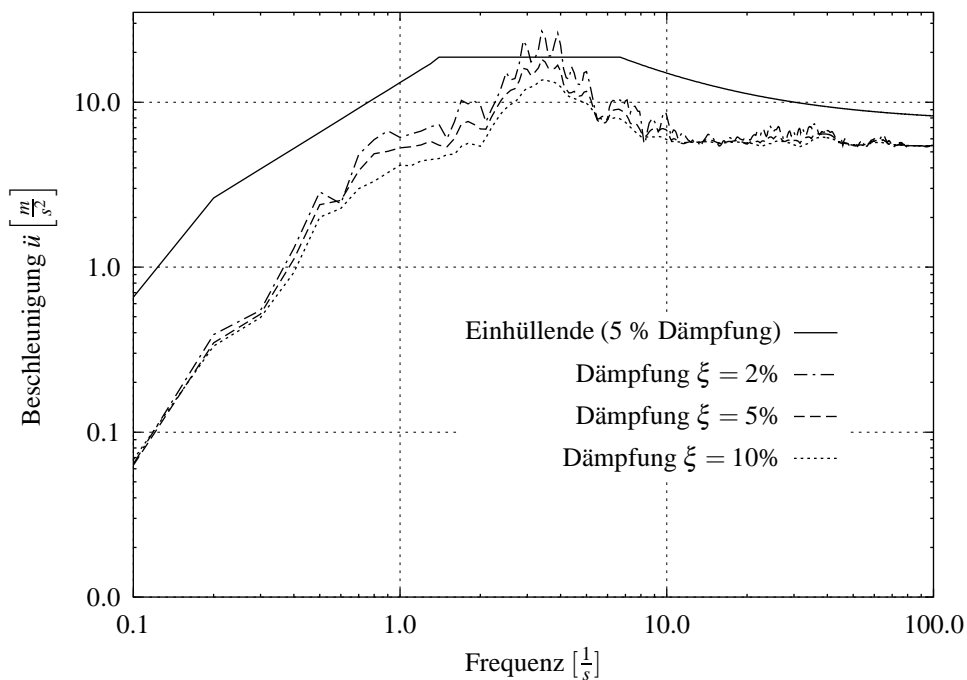


Abb. 6.10: Synthet. Beben: Antwortspektrum der Beschleunigung und zugehöriges elastisches Antwortspektrum (Einhüllende)

Veränderung des Spektrums von hohen Frequenzen zu niedrigeren Frequenzen erkennbar [CP97]. Gerade bei nichtlinearen Systemen, die infolge der veränderlichen Steifigkeiten auch veränderliche Eigenfrequenzen besitzen, kann dies einen großen Einfluß auf die Strukturantwort haben.



Insgesamt wird deutlich, daß die Bereitstellung von Zeitverläufen der Bodenbeschleunigung (*Seismogrammen*) für bestimmte Standorte sehr aufwendig ist. Die große Zahl der zu variierenden Parameter (Intensität, Frequenzbereich, Dauer usw.) erschwert die Berechnung nichtlinearer Strukturen erheblich, da mit jedem ermittelten Beschleunigungsverlauf eine eigene nichtlineare Strukturanalyse durchgeführt und ausgewertet werden muß.

## 6.4 Parameterstudie

Auch die zu berechnende Struktur bietet eine große Anzahl von Parametern. Neben der Geometrie des Behälters können unterschiedliche Baugrundverhältnisse oder konstruktive Details angesetzt werden. Deshalb werden hier nur einige exemplarische Untersuchungen dargestellt, die sowohl die besonderen Aspekte der numerischen Simulation als auch das mechanische Verhalten der Struktur beleuchten sollen.

Für die numerischen Simulationen wird das *Generalized- $\alpha$*  Verfahren mit folgender Parametrisierung genutzt:  $\alpha_f = 0.44$   $\alpha_m = 0.33$   $\beta = 0.2772$   $\gamma = 0.611$

Es werden drei verschiedene Behältergeometrien nach [Sch90] verwendet. Die Abmessungen sind in Tabelle 6.1 dargestellt. Die Dicke des Bodenblechs beträgt bei allen Tanks 5 mm. Als

Behälter	T3	T9	T13
Höhe [m]	20	24	32
Radius [m]	40	12	8
Radius Dach [m]	96	36	34
Blechdicke Mantel [mm]	13.5 – 30.8	6.0 – 10.9	6.0 – 10.7
Blechdicke Rand [mm]	22	8	8

**Tabelle 6.1:** Abmessungen der Behälter

Material wird Stahl\* angenommen. Für die Flüssigkeit wird Wasser<sup>†</sup> verwendet. Der Tank T3 ist mit einem H/R-Verhältnis von 0.5 ein sehr gedrungener Behälter, während der Tank T13 (H/R = 4.0) schlanken Tanks zuzuordnen ist. Der Tank T9 (H/R = 2.0) liegt dazwischen. Die Klassifizierung des Untergrundes für linear-elastisches Verhalten erfolgt nach [Hos87]

Bodenklasse	E-Modul [ $\frac{N}{mm^2}$ ]	Querdehnzahl $\nu$ [-]	Dichte $\rho$ [ $\frac{t}{m^3}$ ]
A: Lockersedimente	340	0.4 – 0.5	1.8
M: mittelsteife Sedimente	1300	0.3 – 0.4	2.1
R: Fels (verfestigtes Gestein)	12400	0.2 – 0.3	2.4

**Tabelle 6.2:** Untergrundklassifizierung nach [Hos87]

\*E-Modul =  $210000 \frac{N}{mm^2}$ , Querdehnzahl  $\nu = 0.3$ , Dichte  $\rho = 7.85 \frac{t}{m^3}$ , Fließspannung  $\sigma_F = 400 \frac{N}{mm^2}$

<sup>†</sup> $\rho = 1.0 \frac{t}{m^3}$

(siehe Tabelle 6.2). Nichtlineares Baugrundverhalten kann nur für Bodenklasse A gewählt werden, da für andere Klassen die notwendigen Materialparameter nicht für das Rechenmodell verfügbar sind [CPWJ91]. Zur besseren Vergleichbarkeit der Ergebnisse wird deshalb im Folgenden nur lineares Verhalten angenommen.

### 6.4.1 Einfluß der Behältergeometrie

Die Geometrien der untersuchten Behälter (T3 gedrungen – T9 mittel – T13 schlank) unterscheiden sich wesentlich voneinander. Deshalb ist ein stark unterschiedliches Antwortverhalten zu erwarten [WR<sup>+</sup>98].

Die folgenden Ergebnisse beziehen sich auf Tanks, die auf weichem Boden (Bodenklasse A) gegründet sind und mit einer der Bodenklasse entsprechenden Erdbebenanregung beansprucht werden. Die Dauer der jeweils verwendeten Erdbeben beträgt 15 s, die Starkbebenphase etwa 6 s.

Aus der Knotenverschiebung am oberen Tankrand (Abbildung 6.11) ist das globale Verformungsverhalten erkennbar. Der schlanke Tank T13 zeigt ausgeprägte Schaukelbewegungen, die bereits bei relativ schwachen Beben mit einem partiellen Abheben des Bodenblechs von der Gründung einhergehen. Dies führt an der gegenüber liegenden Seite zu lokal sehr hohen Druckkräften in Meridianrichtung (Abbildung 6.12).

Ein ähnliches, jedoch nicht so ausgeprägtes Verhalten zeigt der Behälter T9. Die Schaukelbewegungen sind geringer, wodurch auch die lokale Beanspruchung am Behälterfuß kleiner bleibt.

Der gedrungene Tank T3 hebt bei der gewählten Anregung nicht ab. Deshalb bleiben auch die Verformungen sehr klein und es können keine lokalen Spannungskonzentrationen beobachtet werden.

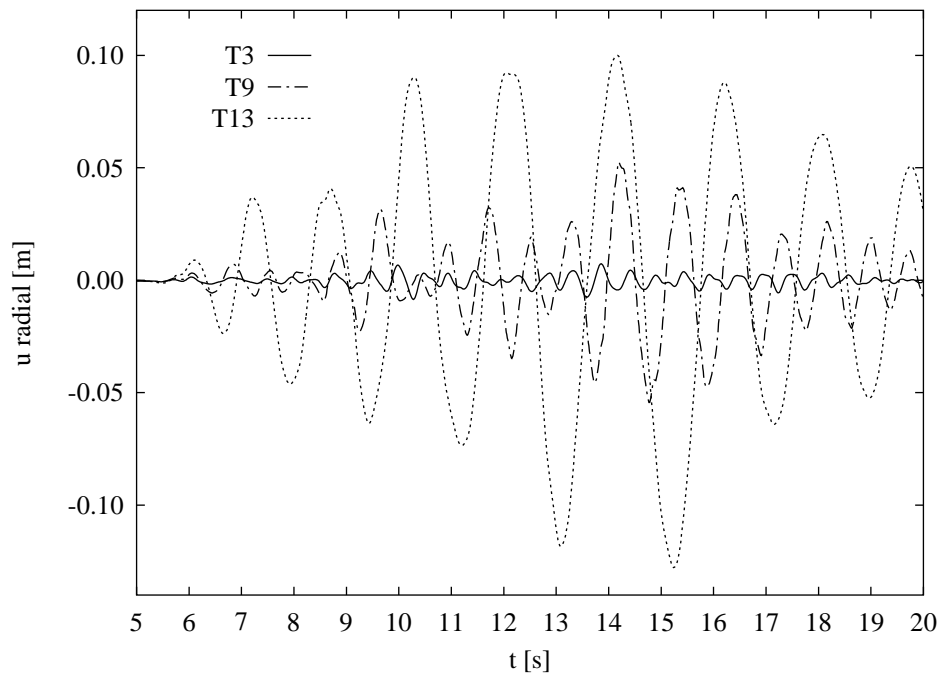
Bei gleicher Konfiguration, aber verdreifachter Erdbeben-Intensität ergibt sich beim Tank T13\* ein ähnliches Antwortverhalten (vgl. Abbildungen 6.11 und 6.13). Allerdings ist die Schaukelbewegung noch ausgeprägter, wodurch sich die Frequenz der Bewegung infolge des stärkeren Steifigkeitsverlustes deutlich erniedrigt, während sich die Amplituden nahezu verdoppeln. Die maximalen Meridianspannungen wachsen nur etwas mehr an (Abbildung 6.14). Die Ursache hierfür liegt unter anderem darin, daß es infolge der hohen lokalen Beanspruchung zu plastischen Verformungen an der unteren Behälterwandung kommt, wodurch sich eine gewisse Lastumlagerung einstellt.

Völlig anders verhält sich wiederum der gedrungene Tank T3. Auch er hebt nun partiell von der Gründung ab, prallt anschließend jedoch heftig auf die Gründung. Während dieser Vorgang an der Verschiebung des Knotens am oberen Tankrand kaum erkennbar ist (Abbildung 6.13), zeigt sich bei den Meridiankräften eine enorm hohe Impulskraft (Abbildung 6.14).

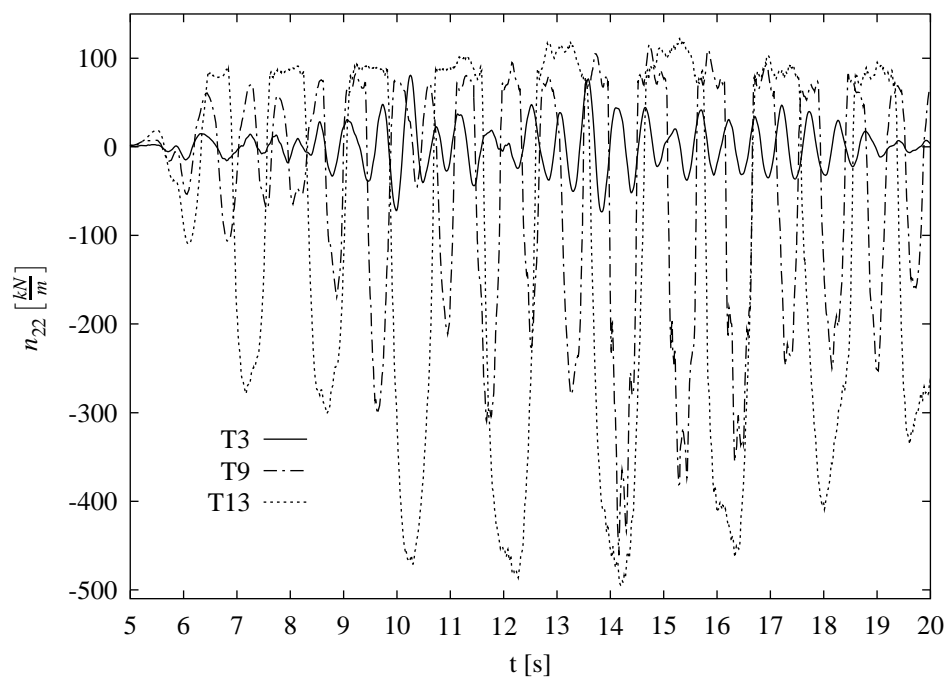
Auch die räumliche Verteilung der Meridiankräfte ist völlig unterschiedlich. Betrachtet man den gedrungenen Tank T3 zu einem Zeitpunkt kurz nach dem Aufprall (Abbildung 6.15)<sup>†</sup>,

\*Aus Gründen der Übersichtlichkeit ist bei den Abbildungen 6.13 und 6.14 der Tank T9 nicht gezeigt

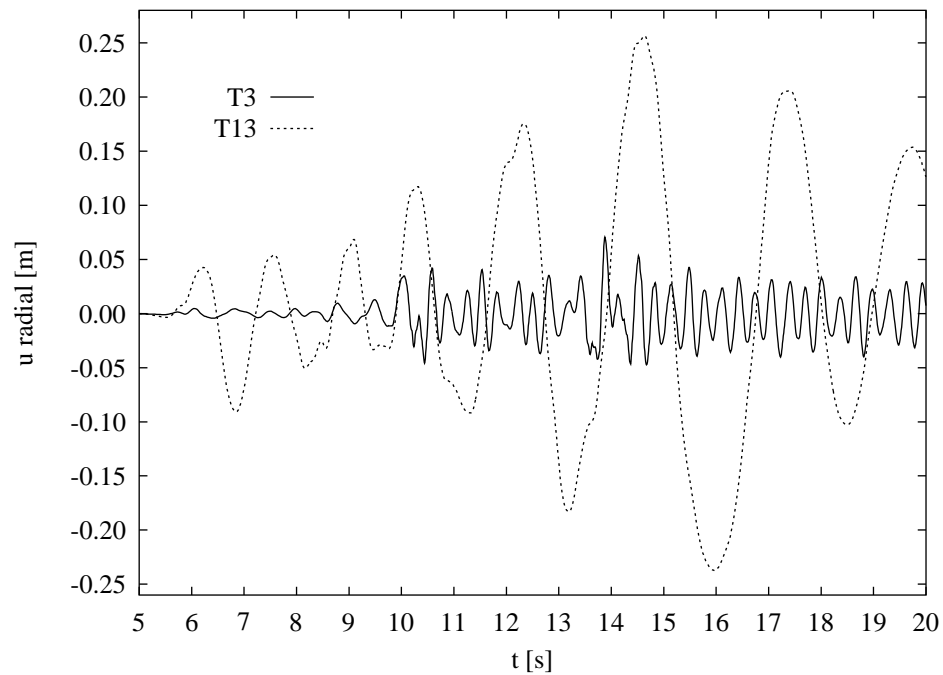
<sup>†</sup>Dargestellt ist hier nur der halbe Umfang (Symmetrie)



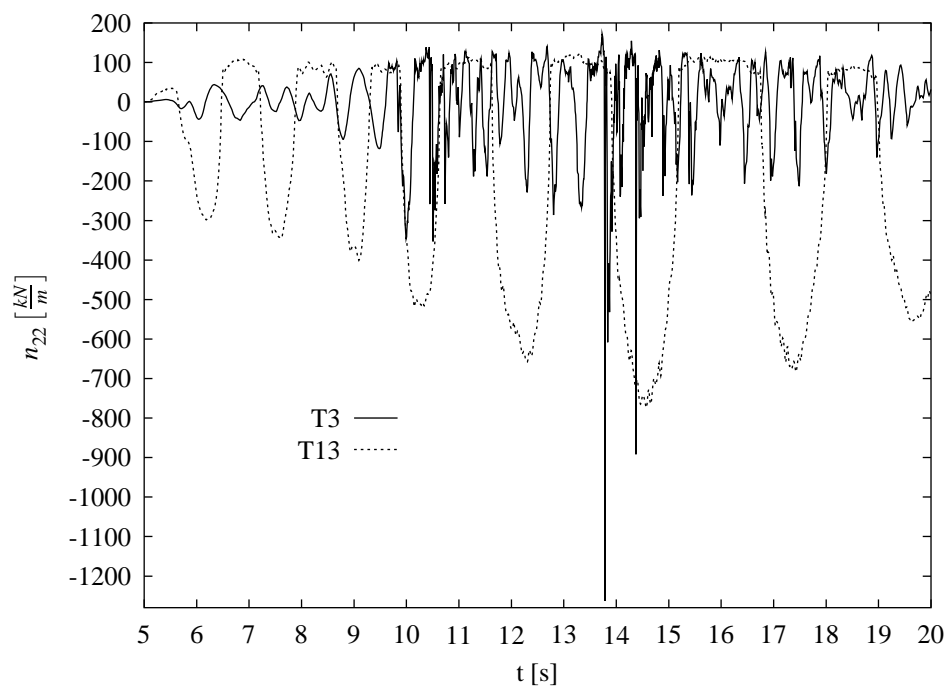
**Abb. 6.11:** Horizontale Knotenverschiebung am oberen Tankrand für verschiedene Behälter (Bodenklasse A, Arias-Intensität  $I_0 = 1.0$ )



**Abb. 6.12:** Meridiankräfte am unteren Tankrand für verschiedene Behälter (Bodenklasse A, Arias-Intensität  $I_0 = 1.0$ )

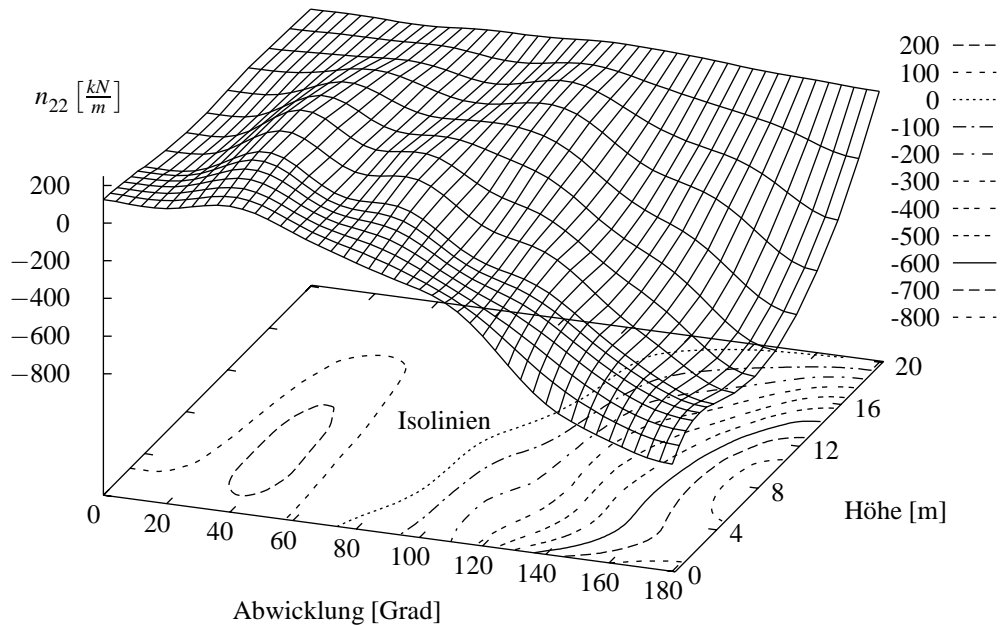


**Abb. 6.13:** Horizontale Knotenverschiebung am oberen Tankrand für verschiedene Behälter (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )



**Abb. 6.14:** Meridiankräfte am unteren Tankrand für verschiedene Behälter (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )

so läßt sich eine weiträumige Verteilung der Druckkräfte sowohl über den Umfang (ca.  $60^\circ$ - $180^\circ$ ) als auch über die Höhe erkennen. Lediglich in einem kleinen Bereich (ca.  $0^\circ$ - $60^\circ$ ) sind



**Abb. 6.15:** Abwicklung der Meridiankräfte am Tank T3 (Bodenklasse A,  $I_0 = 3.0$ )

Zugkräfte infolge der Separation von der Gründung vorhanden. Am oberen Rand (Tankdach) sind die Meridiankräfte sehr klein. Dies deutet darauf hin, daß das Dach nur einen geringen Einfluß auf das Tragverhalten in vertikaler Richtung hat.

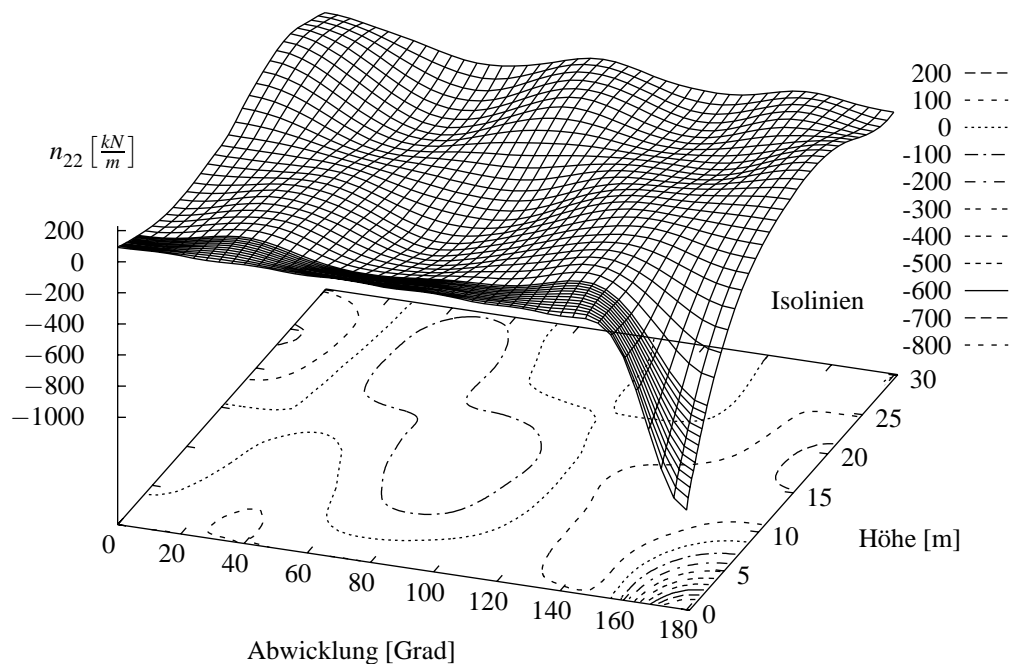
Im Gegensatz hierzu sind die vertikalen Druckkräfte beim schlanken Tank T13 sowohl in Umfangsrichtung (ca.  $150^\circ$ - $180^\circ$ ) als auch über die Höhe lokal eng begrenzt (Abbildung 6.16). Im oberen Bereich (bei ca. 16 m) sind über der maximalen Druckkraft sogar Zugkräfte vorhanden.

Am Übergang zum Tankdach sind hier im Gegensatz zum gedrunenen Tank Meridiankräfte vorhanden. Das Dach hat beim schlanken Tank offensichtlich einen größeren Einfluß auf das Tragverhalten in vertikaler Richtung als beim gedrunenen Tank.

Im Bereich zwischen  $0^\circ$  und  $150^\circ$  hebt der Behälter zum dargestellten Zeitpunkt von der Gründung ab, weshalb am unteren Behälterrand Zugkräfte entstehen. Auffallend ist hierbei, daß diese nahezu konstant verteilt sind (ca.  $100 \frac{kN}{m}$ ).

Die Zeitverläufe der Energien (Abbildungen 6.17 und 6.18) verdeutlichen einen weiteren Unterschied zwischen gedrunenen und schlanken Tanks.

Der Tank T3 zeigt nur bei  $t \approx 10s$  und bei  $t \approx 14s$  deutliche Werte an kinetischer Energie. Dies korreliert mit dem partiellen Abheben und Aufprallen des Tanks (vgl. Abbildung



**Abb. 6.16:** Abwicklung der Meridiankräfte am Tank T13 (Bodenklasse A,  $I_0 = 3.0$ )

6.14). Zum Ende der Simulation fällt die kinetische Energie stark ab. Ursache hierfür ist vor allem die Dissipation der Energie durch die Energieabstrahlung in den Halbraum, weniger die Dämpfung durch plastische Verformungen\*. Die durch das Erdbeben zugeführte Energie wird also nahezu vollständig wieder in den Halbraum abgestrahlt.

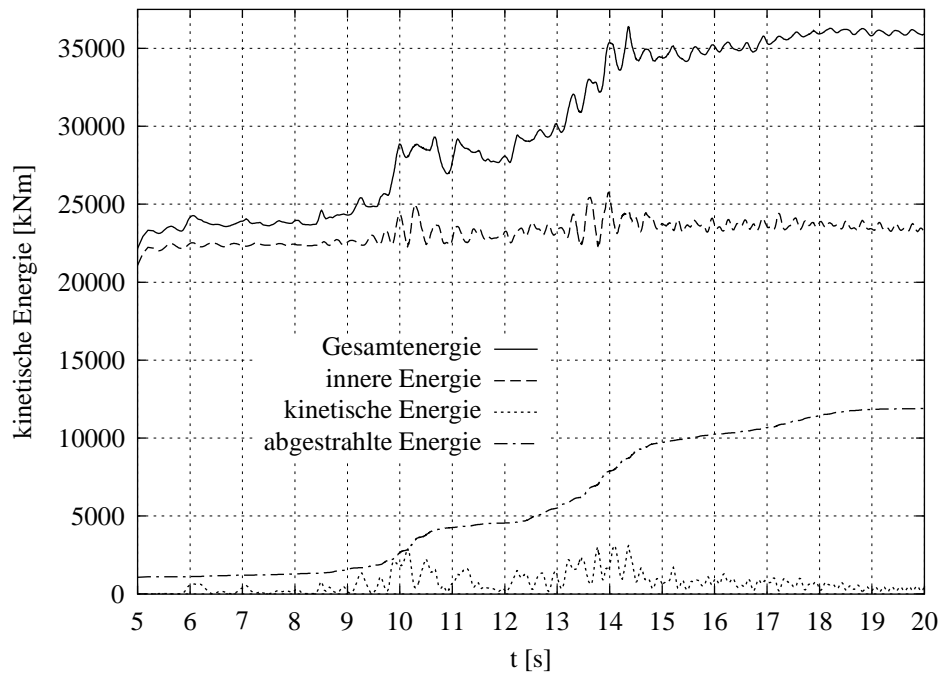
Beim schlanken Tank T13 steigt die kinetische Energie bis zur maximalen Erdbebenanregung sukzessive an und verbleibt bis zum Ende des Bebens in Wechselwirkung mit der inneren Energie größtenteils im System. Es geht lediglich ein relativ geringer Anteil durch Energieabstrahlung in den Halbraum verloren, während ein deutlicher Anteil der Energie durch plastische Verformungen physikalisch dissipiert wird†.

Die Ursache für dieses unterschiedliche Verhalten ist vor allem in der Entwicklung des Steifigkeitsparameters (Gleichung 4.2, Seite 47) zu finden (Abbildung 6.19 und 6.20). Dieser liegt beim gedrungenen Tank nahezu während der gesamten Simulation über 10 % der Ausgangssteifigkeit. Lediglich bei  $t \approx 10\text{ s}$  und bei  $t \approx 14\text{ s}$  hebt der Tank von der Gründung ab und die Systemsteifigkeit fällt drastisch ab. Der Tank bleibt also meist vollständig mit dem Boden in Kontakt und kann deshalb ähnlich wie ein verankerter Tank Energie in Form von Kompressions- und Scherwellen in den Halbraum abstrahlen.

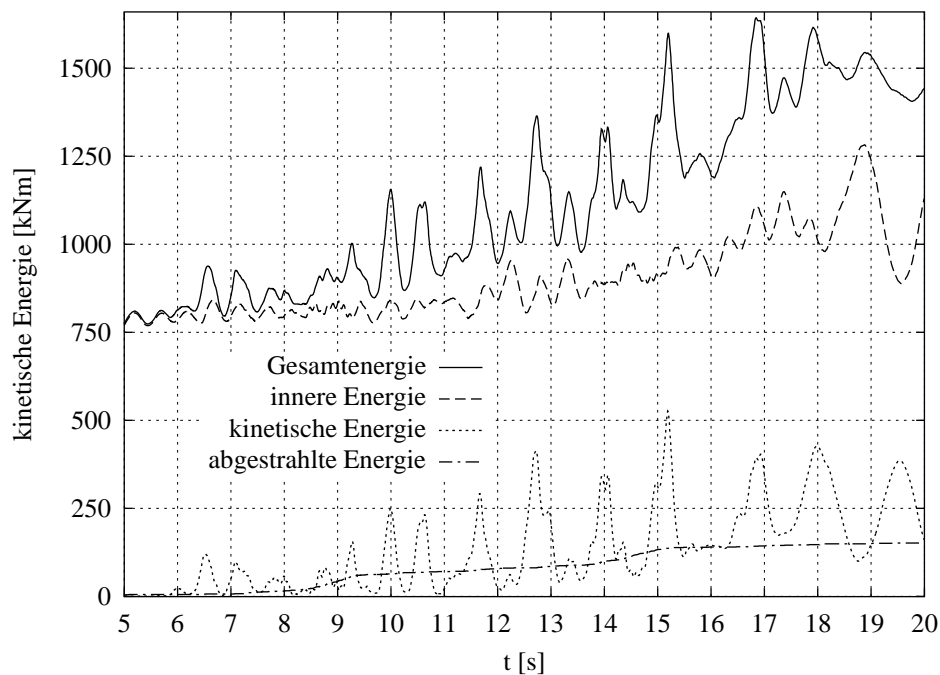
Im Gegensatz hierzu schwankt der Steifigkeitsparameter beim schlanken Tank während der gesamten Simulation zwischen der Ausgangssteifigkeit und extrem kleinen Werten. Das mittlere Niveau ist im Vergleich zum gedrungenen Tank jedoch viel niedriger. Hierdurch ist die

\*diese ist in der "inneren Energie" enthalten

†ablesbar am tendenziellen Anstieg der inneren Energie



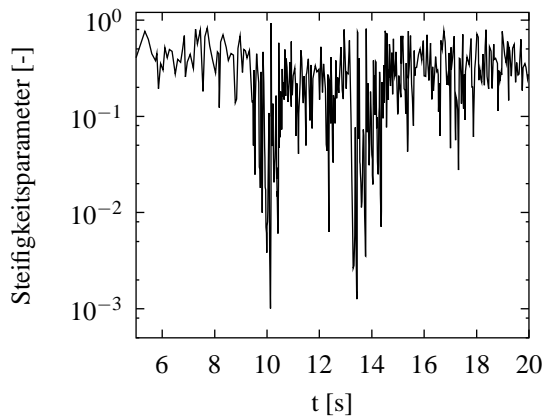
**Abb. 6.17:** Verlauf der Energien für Tank T3 (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )



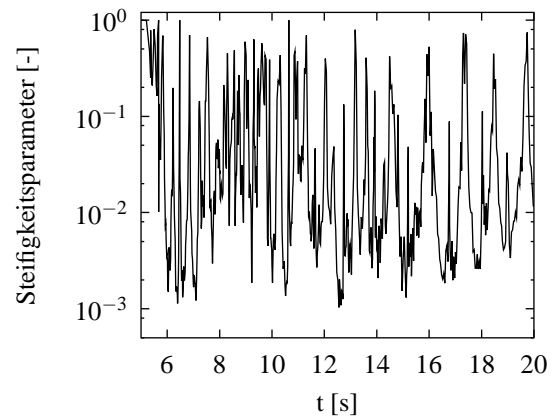
**Abb. 6.18:** Verlauf der Energien für Tank T13 (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )

mechanische Verbindung zum Untergrund kaum noch vorhanden, weshalb ein Energietransfer erschwert wird.

Das dynamische Antwortverhalten gedrungener und schlanker Tanks ist also völlig unter-



**Abb. 6.19:** Verlauf des Steifigkeitsparameters beim Tank T3 (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )



**Abb. 6.20:** Verlauf des Steifigkeitsparameters beim Tank T13 (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )

schiedlich. Dieses unterschiedliche Verhalten wirkt sich auch auf die Versagensform der Behälter (siehe Abschnitt 6.4.4) sowie auf die numerische Simulation aus.

### Numerische Simulation

Die numerische Simulation sollte automatisch auf die unterschiedlichen Anforderungen während der Berechnung reagieren. Beispielsweise erfordert das überwiegend niederfrequente Schaukeln schlanker Tanks andere Zeitschrittlängen als das hochfrequente Aufprallen gedrungener Tanks:

- Der Tank T3 wird bis zum ersten (schwach ausgeprägten) Aufprall bei  $t \approx 10\text{ s}$  mit relativ großen Schrittweiten berechnet (Abbildung 6.21). Anschließend erfordert die hochfrequente Strukturantwort relativ kleine Schritte, die erst zum Ende der Simulation im Mittel wieder vergrößert werden können ( $0.0002 \leq \Delta t \leq 0.05$ ).
- Das ausgeprägte niederfrequente Schwanken des schlanken Tanks T13 führt dagegen zu einem völlig anderen charakteristischen Verlauf der Schrittweiten (Abbildung 6.22). Die Integrationslängen schwanken stetig zwischen den a priori vorgegebenen Schranken ( $0.0002 \leq \Delta t \leq 0.02$ ).

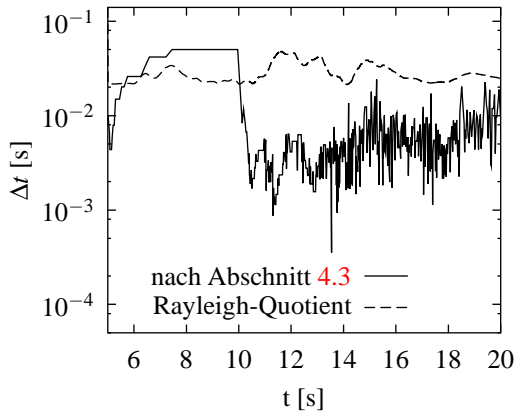
Wie bereits in Kapitel 4 ausführlich gezeigt wurde, erfolgt die Abschätzung der Schrittweite bei Verwendung des *Rayleigh-Quotienten*\* völlig unzureichend. Die in dieser Arbeit vorgestellte Schrittweitensteuerung (Abschnitt 4.3), die auf dem Fehlerschätzer nach *Zienkiewicz/Xie* basiert† sowie der in Abschnitt 4.2 (Seite 50) vorgestellten Bezugsgröße  $\|\mathbf{u}_{ref}\|_2$  (Gleichung 4.15) spiegelt das Strukturverhalten wesentlich besser wider.

In Abbildung 6.23 ist für den Tank T13 der absolute Zeitintegrationsfehler nach Gleichung (4.6) zu sehen. Der Fehler zeigt bis ca. 6.5 s relativ kleine Werte an, da der Tank in diesem

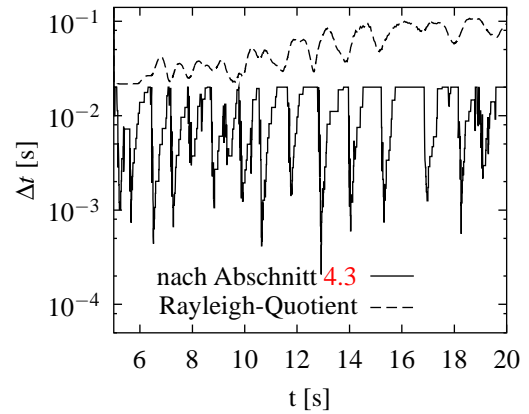
\*Gleichung (4.1), Seite 47

†siehe Gleichung (4.6) in Abschnitt 4.1, Seite 48

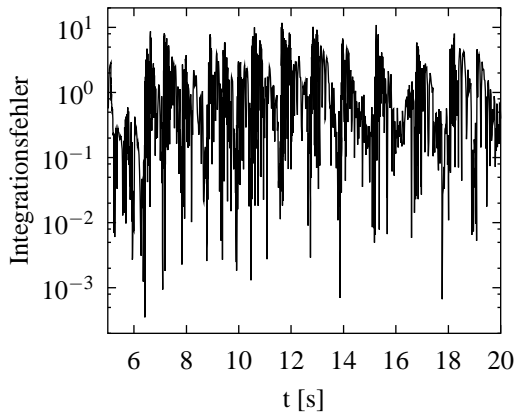




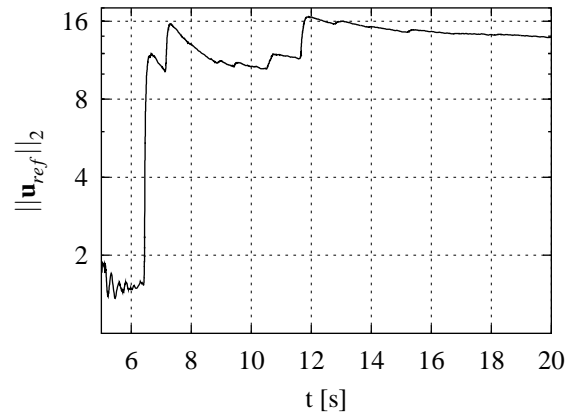
**Abb. 6.21:** Integrationsschrittweiten beim Tank T3 (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )



**Abb. 6.22:** Integrationsschrittweiten beim Tank T13 (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )



**Abb. 6.23:** Absoluter Zeitintegrationsfehler nach Gleichung (4.6) für Tank T13 bei adaptiver Schrittweitensteuerung

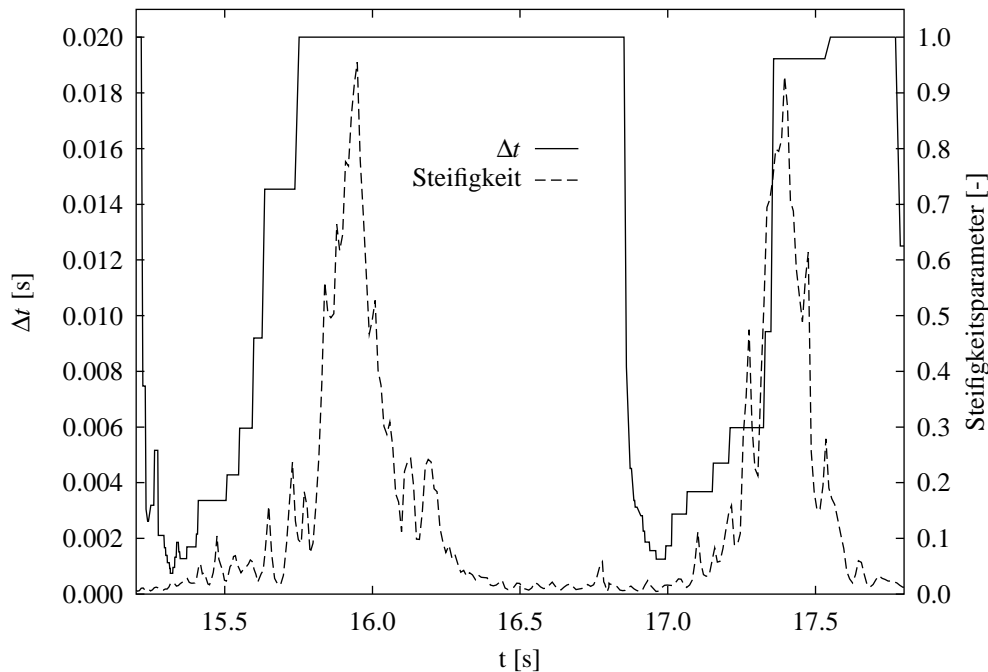


**Abb. 6.24:** Fehler-Bezugsgröße  $\|\mathbf{u}_{ref}\|_2$  nach Gleichung (4.15) für Tank T13

Bereich nur sehr schwach dynamisch antwortet. Erst beim Abheben des Tanks steigt der Fehler stark an, weshalb die Schrittweite verkürzt wird. Im weiteren Verlauf der Berechnung schwankt der Fehler abhängig von der jeweiligen Schrittweite bzw. von der Strukturantwort.

Auch die Fehler-Bezugsgröße  $\|\mathbf{u}_{ref}\|_2$  steigt zu Beginn stark an, da die inkrementelle Verschiebungsnorm infolge des Abhebens des Tanks von der Gründung stark anwächst (Abbildung 6.24). Im weiteren Verlauf der Berechnung bleibt die Bezugsgröße auf einem nahezu konstanten Niveau. Die in Abschnitt 4.2 vorgestellte Bezugsgröße ist also auch für die adaptive Zeitintegration komplexer Finite-Element-Systeme hervorragend geeignet, da sie ein für die jeweilige Berechnung charakteristisches Maß angibt.

Der Zusammenhang zwischen der Integrationsschrittweite und dem Steifigkeitsparameter ist in Abbildung 6.25 für den Tank T13 dargestellt. Offensichtlich wird die Schrittweite im Bereich zunehmender Steifigkeit vergrößert, während bei abnehmender Steifigkeit mit konstan-



**Abb. 6.25:** Integrationsschrittweiten und Steifigkeitsparameter bei Tank T13

ter (maximaler) Schrittweite gerechnet wird. Die Begrenzung der Steifigkeitsänderung nach Gleichung (4.18)\* spielt hier nur eine untergeordnete Rolle, da sich die Werte innerhalb der jeweiligen Schrittweiten nicht sprunghaft ändern ( $\Delta s < 0.2 = \Delta s_{max}$ ). Die Schrittweitensteuerung erfolgt deshalb nahezu ausschließlich durch die Fehlerschätzung. Die Schrittweite ist vor allem von der absoluten Größe des Steifigkeitsparameters unabhängig. Eine Steuerung wie sie in [WGS85] vorgeschlagen wird, führt deshalb zu falsch angepaßten Schrittweiten (vgl. Abschnitt 4, Seite 47).

Ein Vergleich mit Abbildung 6.14 zeigt, daß die drastische Verkürzung der Schrittweite (z.B. bei  $t \approx 16.8s$ ) jeweils mit dem Aufprall des Tanks auf die Gründung einhergeht. Es wird also in den Bereichen mit einer sehr kleinen Schrittweite gerechnet, wo große Spannungsgradienten abzubilden sind.

Die Beispiele zeigen, daß die in dieser Arbeit vorgestellte Schrittweitensteuerung zu einer optimalen Anpassung der Integrationslängen an die Strukturantwort führen. Die Erhöhung der Genauigkeit geht zudem mit einer Minimierung des numerischen Aufwands einher.

## 6.4.2 Verhalten bei unterschiedlicher Bodensteifigkeit

Verankerte Behälter antworten abhängig von der Bodensteifigkeit in geringfügig unterschiedlichen Frequenzbereichen [Sch96b, WCST95]. Vor allem die Stärke der geometrischen Dämpfung hängt stark von der Bodenklasse ab [WR<sup>+</sup>98]. Hierbei ist zu beachten, daß sich vor allem beim gedrunenen Tank mit einer festen Verankerung die Antwortfrequenz kaum, die geometrische Dämpfung jedoch relativ stark ändert.

\*Abschnitt 4.4, Seite 57

Ein analoges Verhalten zeigen unverankerte Tanks unter schwacher Anregung, sofern sie nicht von der Gründung abheben und sich folglich wie verankerte Tanks verhalten. Dies gilt vor allem für den gedrungenen Tank, da er erst bei sehr starken Anregungen abhebt. Aber auch bei einer stärkeren Anregung (Tank hebt partiell ab) weist der gedrunzene Tank ein ähnliches Antwortverhalten auf (Abbildung 6.26). Er antwortet in einem kaum unterschiedlichen Frequenzbereich (vgl. Nulldurchgänge in Abbildung 6.26), die Größe der Amplituden unterscheidet sich jedoch erheblich voneinander. Vor allem wird der Tank mit Bodenklasse A sehr viel stärker gedämpft, was am abklingen der Amplituden zum Ende der Simulation gut zu erkennen ist.

Für die Beanspruchung der Tankwand durch Meridiankräfte  $n_{22}$  (Abbildung 6.27) ergibt sich ein analoges Bild. Die maximalen Amplituden sind bei Bodenklasse M im Verhältnis zu den Verschiebungen noch größer, da die steifere Gründung zu höheren Kräften beim Aufprall des Tanks auf die Gründung führen.

Auch hier verhalten sich schlankere Tanks völlig anders. Ist die Erdbebenbeschleunigung so stark, daß der Tank vom Boden abhebt, wirkt sich die Steifigkeit des Baugrunds kaum noch aus (hier exemplarisch am Tank T9 dargestellt). Bei nahezu identischer Antwortfrequenz sind lediglich geringfügig unterschiedliche Amplituden bei der charakteristischen Verformung des Tanks vorhanden (Abbildung 6.28). Allerdings treten bei steifen Gründungen lokal höhere Beanspruchungen auf, da sich die Kräfte infolge einer geringeren Umlagerungsmöglichkeit stärker auf einen kleinen Bereich konzentrieren (Abbildung 6.29). Hierdurch können eher lokale Beulen entstehen als bei weichen Böden.

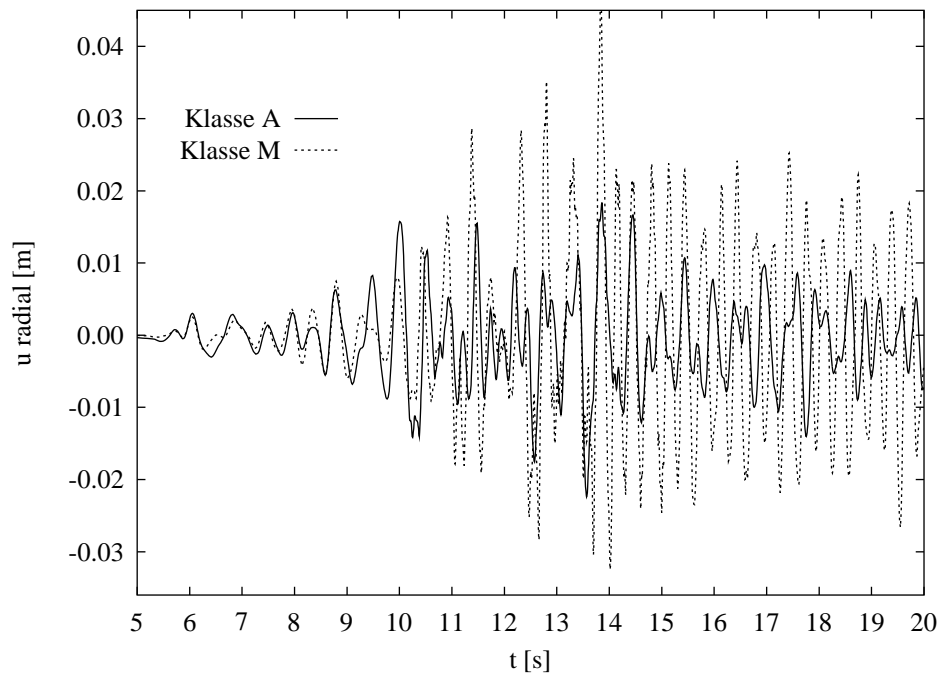
Das globale Schwingverhalten wird vorrangig durch den Steifigkeitsabfall des Systems beim Abhebevorgang beeinflußt. Es tritt somit eine Entkoppelung zwischen Boden und Bauwerk ein, wodurch der Einfluß der Bodensteifigkeit stark an Bedeutung verliert.

Auch die Energieabstrahlung (Wellenausbreitung) in den Halbraum spielt eine relativ geringe Rolle. Diese Erkenntnis ist kongruent zur Arbeit von *Schäpertöns* [Sch96b]. Er hat festgestellt, daß die relativ weichen Stahltanks wesentlich weniger Energie abstrahlen als steife Behälter aus Beton. Unverankerte Tanks sind bezogen auf den Baugrund jedoch noch viel weicher als verankerte Stahltanks.

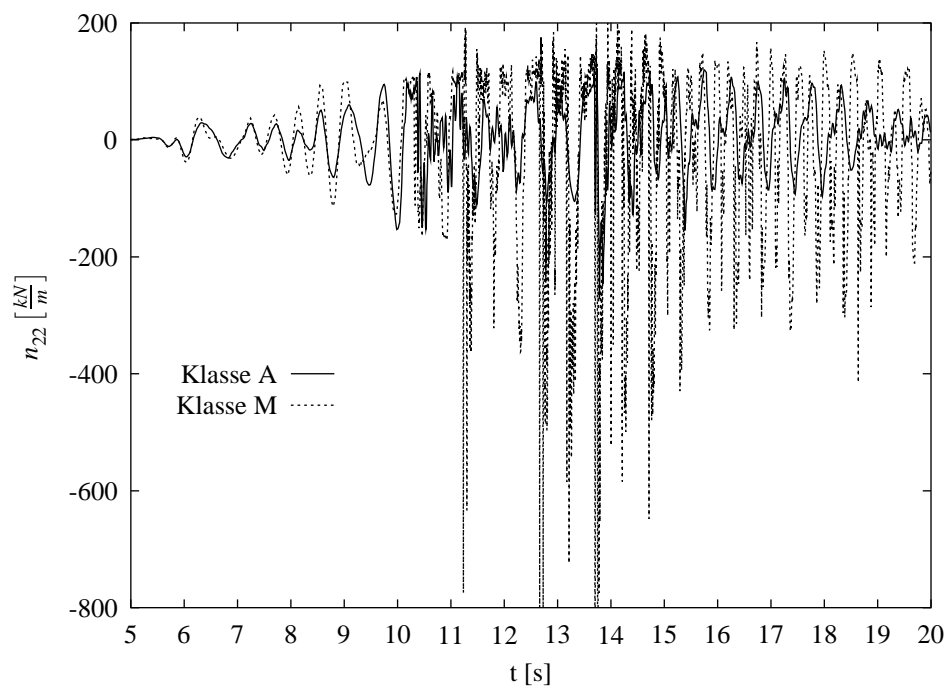
Bei den gezeigten Ergebnissen muß beachtet werden, daß zur eindeutigen Zuordnung der Effekte eine identische Erdbebenanregung angesetzt ist. Eine dem jeweiligen Boden zugehörige, spektrumskonforme Anregung kann zu anderen Resultaten führen, wie der folgende Abschnitt zeigt.

### 6.4.3 Einfluß der Einwirkung

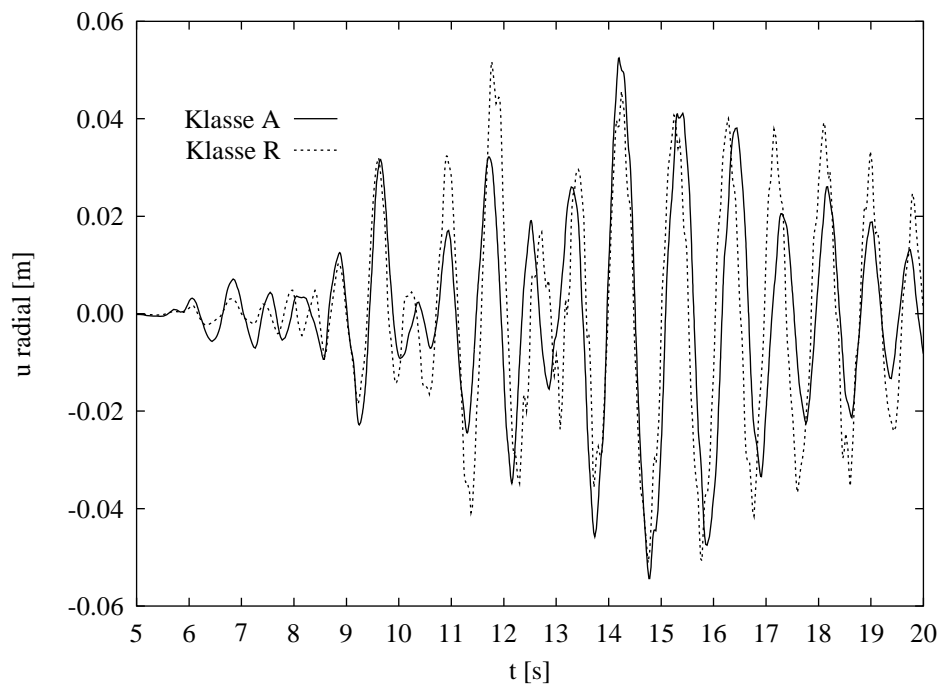
Die Auswirkung unterschiedlich stark skaliertter Anregungen wurde bereits in Abschnitt 6.4.1 dargestellt. Aus dem Vergleich der Abbildungen 6.11 und 6.13 ist deutlich die Verschiebung der Antwortfrequenz in den niederfrequenten Bereich bei zunehmender Beben-Intensität zu erkennen (insbesondere beim Tank T13), ebenso die Veränderung der Amplituden. Die Frequenzverschiebung ist eine Folge des zunehmenden Steifigkeitsverlustes bei einer stärkeren Anregung, da sich hierdurch die Eigenfrequenzen erniedrigen.



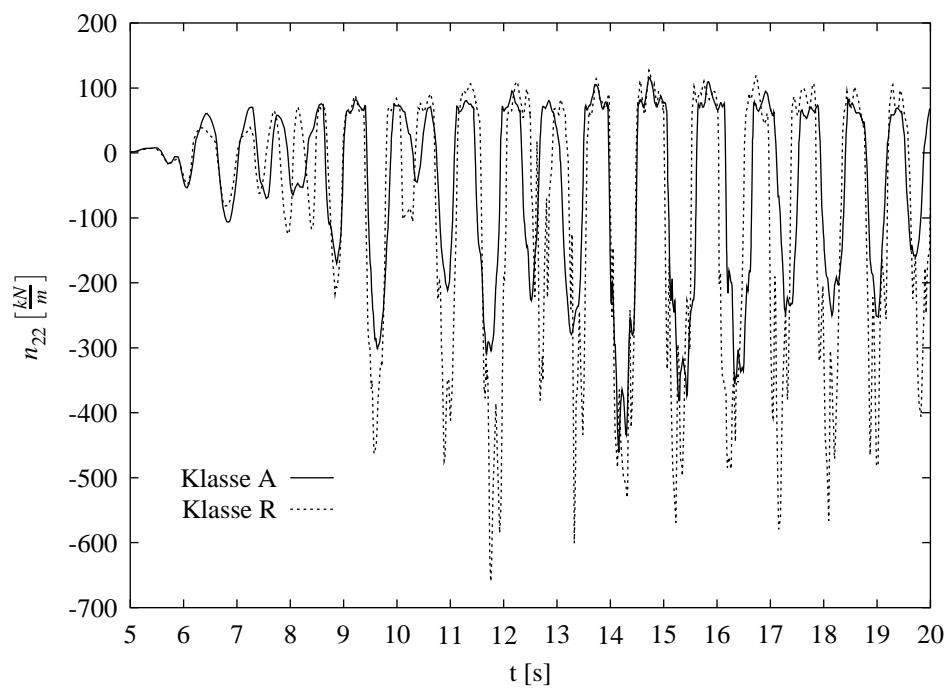
**Abb. 6.26:** Horizontale Knotenverschiebung am oberen Tankrand für Tank T3 (Bodenklasse A und M, Arias-Intensität  $I_0 = 2.0$ )



**Abb. 6.27:** Meridiankräfte am unteren Tankrand für Tank T3 (Bodenklasse A und M, Arias-Intensität  $I_0 = 2.0$ )

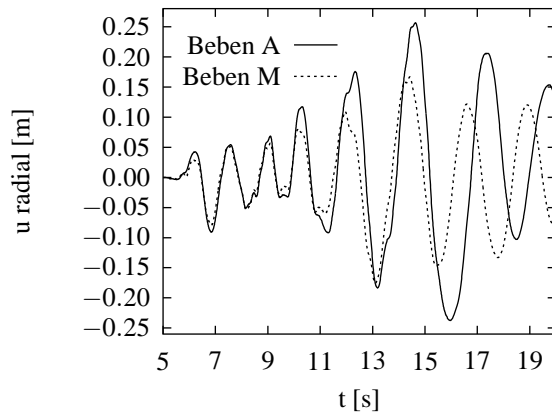


**Abb. 6.28:** Horizontale Knotenverschiebung am oberen Tankrand für Tank T9 (Bodenklasse A und R, Arias-Intensität  $I_0 = 1.0$ )

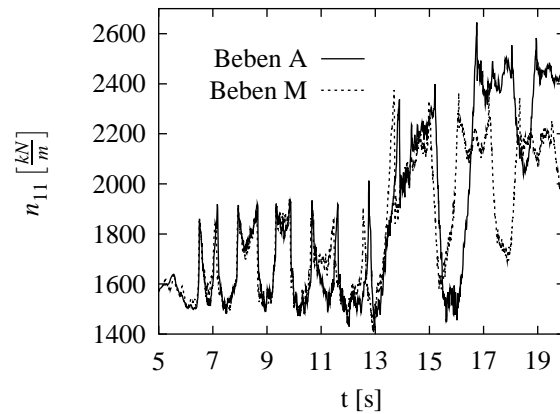


**Abb. 6.29:** Meridiankräfte am unteren Tankrand für Tank T9 (Bodenklasse A und R, Arias-Intensität  $I_0 = 1.0$ )

Um den Einfluß der Anregungsfrequenz zu verdeutlichen, wird der Behälter T13 (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ ) mit einem Beben der Bodenklasse M beansprucht. Beim Beben der Klasse A dominieren gegenüber dem Beben der Klasse M niederfrequente Anteile. Aus dem Vergleich der charakteristischen Verschiebungen (Abbildung 6.30) ist in den ersten fünf Sekunden der Simulation nur ein sehr geringer Unterschied erkennbar. Im wei-



**Abb. 6.30:** Horizontale Knotenverschiebung am oberen Rand beim Tank T13 für verschiedene Beben (Bodenklasse A, Arias-Intensität  $I_0 = 3.0$ )



**Abb. 6.31:** Ringkräfte  $n_{11}$  am unteren Rand (1.0 [m] über dem Boden)

teren Verlauf tritt ein deutlicher Unterschied in der Frequenz und in der Amplitude auf. Das Beben der Klasse A führt zu größeren Amplituden, wodurch sich infolge der ausgeprägten Schaukelbewegungen die Antwortfrequenz deutlich erniedrigt.

Die Materialbeanspruchung in Umfangsrichtung (Abbildung 6.31) ist dadurch beim Beben der Klasse A merklich höher. Die charakteristische Form der Zeitverläufe ist durch das Abheben des Tanks bedingt.

Auch die anderen Tanks (T3 und T9) reagieren unterschiedlich auf verschiedene Einwirkungen. Die Bebenanregung dominiert somit das nichtlineare Antwortverhalten aller Tanks. Insbesondere die Frequenz sowie die Arias-Intensität sind hier wichtige Parameter.

Die Dauer der Einwirkung beeinflusst vor allem das Versagen der Struktur (siehe folgenden Abschnitt).

#### 6.4.4 Strukturversagen

Das Abheben des Tanks von der Gründung kann bereits als Versagen der Struktur gedeutet werden, vor allem wenn hierdurch Betriebseinrichtungen (z.B. Leitungsanschlüsse) beschädigt werden (vgl. Abbildung 6.2). Wird das Abheben jedoch bei der Konstruktion eines Behälters berücksichtigt, kann man als Versagen einen irreversiblen Zustand definieren, bei

dem der Behälter infolge plastischer Verformungen (z.B. Beulen) seine Gebrauchstauglichkeit bzw. seine Tragfähigkeit verliert.

Die Untersuchungen zeigen, daß sich durch das partielle Abheben des Tanks von der Gründung eine Fließgelenkkette im Bodenblech ausbildet (siehe hierzu auch [Mal96, Sch90]). Hierdurch können – besonders bei mehrmaligem Lastwechsel – Risse entstehen und damit zum Verlust der Gebrauchstauglichkeit (Dichtheit) führen.

Beulen an der Tankwand entstehen – abhängig von der Tankgeometrie – durch unterschiedliche Mechanismen. Wie bereits in Abschnitt 6.4.1 erläutert wurde, verhalten sich gedrungene und schlanke Behälter völlig unterschiedlich während eines Erdbebens. Dies wirkt sich auch auf die Entstehung und die Art von Beulen aus.

### Schlanke Behälter

Bei den schlanken Behältern wirkt sich eine Erhöhung der Stärke der Anregung (Bodenbeschleunigung) nur bis zu einem gewissen Grad aus, da sich der Behälter durch das ausgeprägte Abheben von der Gründung der Beanspruchung gut entziehen kann. Allerdings können bereits beim ersten Abheben kleine, lokale Beulen entstehen. Erst nach mehrmaligem Schaukeln des Behälters entstehen ausgeprägte Beulen (vgl. Abbildung 6.34).

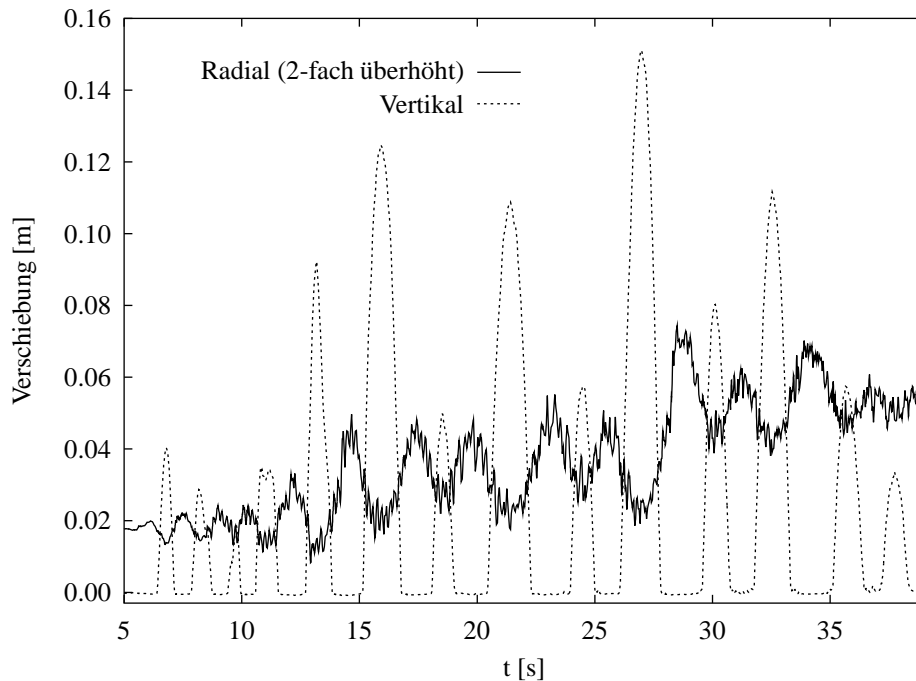
Die Dauer der Einwirkung ist deshalb von großer Bedeutung. Aber auch nach dem Abklingen der Erdbebenanregung schwingt der Behälter weiter. Die größten plastischen Verformungen treten sogar erst ca. 10 Sekunden nach dem Ende der Starkbebenphase auf. Dieser Effekt basiert auf der äußerst geringen Energieabstrahlung in den Halbraum, wodurch das System im Vergleich zu verankerten Behältern kaum gedämpft wird.

Abbildung 6.32 zeigt, daß bereits bei  $t \approx 14$  Sekunden eine erste Beule auftritt, die im weiteren Verlauf mehrmals nahezu verschwindet und wieder entsteht. Erst bei  $t \approx 28$  Sekunden entwickelt sich eine ausgeprägte plastische Beule (Abbildung 6.34). Dieser Vorgang korreliert mit der vertikalen Verschiebung der Tankwand. Insbesondere entsteht die maximale horizontale Knotenverschiebung kurz nach der maximalen vertikalen Verschiebung.

### Gedrungene Behälter

Gedrungene Tanks prallen nach dem Abheben über einen großen Umfangsbereich nahezu gleichzeitig auf die Gründung. In diesem Bereich entstehen ausgeprägte Druckspannungen in Meridianrichtung (Abbildung 6.15). Gleichzeitig entstehen starke Zugspannungen in Umfangsrichtung infolge eines erhöhten dynamischen Flüssigkeitsdrucks. Die Kombination dieser Beanspruchung ergibt eine hohe Vergleichsspannung  $\sigma_v$ . Bei einem Überschreiten der Fließspannung  $\sigma_F$  entsteht eine ausgeprägte Wulstbeule über einen weiten Bereich des Behälters (Abbildung 6.33).

Auch bei den in Abbildung 6.2 gezeigten Schadensfällen dürfte sich dieser Versagensmechanismus abgespielt haben. Hier könnte eine Verstärkung des ersten Schusses Abhilfe schaffen.



**Abb. 6.32:** Vertikale und Horizontale Knotenverschiebung an der lokalen Beulstelle beim Tank T13 (Bodenklasse R, Arias-Intensität  $I_0 = 3.0$ )

### Form der plastischen Beulen

Am Ende der numerischen Simulation verbleiben in den Tanks schließlich jeweils die charakteristischen plastischen Beulen (Abbildung 6.35). Der Ausgangszustand bezieht sich auf den Tank ohne Füllung. Der konstante Anteil in Umfangsrichtung der hier gezeigten Verformung entspricht deshalb ungefähr der elastischen Verformung der Tankwand infolge des Flüssigkeitsdrucks.

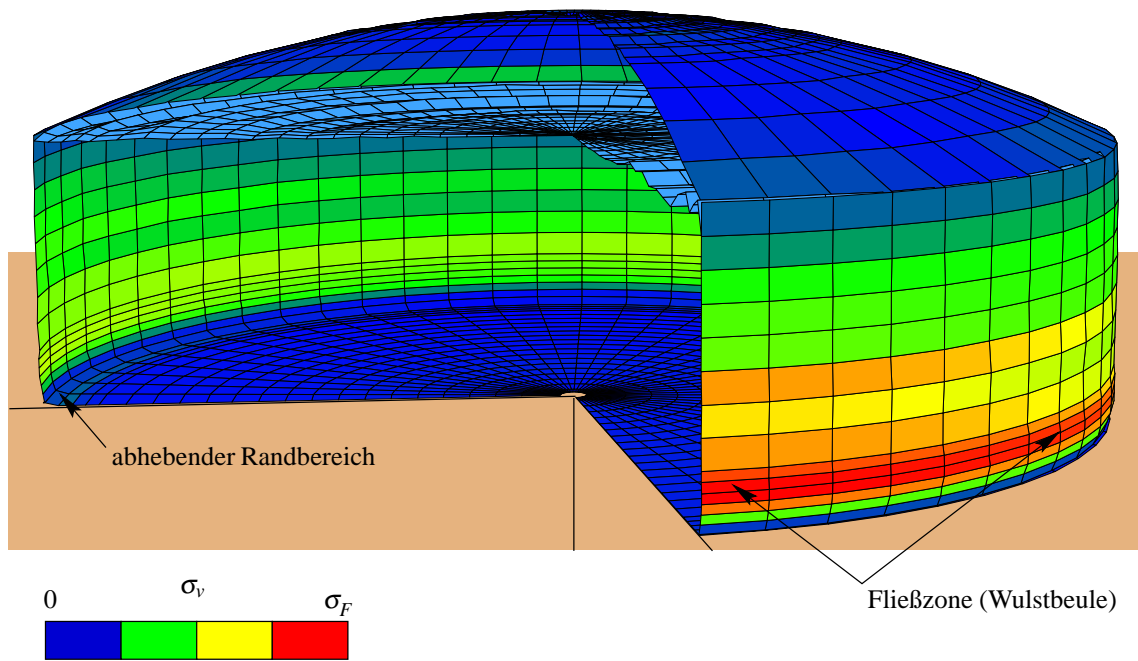
Wie bereits erläutert, ist die Ursache dieser Verformungen das Überschreiten der zulässigen Fließspannung. Die Abbildungen 6.33 und 6.34 zeigen jeweils die maximale Vergleichsspannung der Tanks. Die Zeitpunkte entsprechen jeweils zugleich dem Zeitpunkt der größten Meridiankräfte (vgl. Abbildungen 6.14, 6.15 und 6.16). Die maximalen Meridiankräfte treten wiederum meist zum Zeitpunkt der größten Ringkräfte auf (vgl. hierzu den Tank T13 in den Abbildungen 6.14 und 6.31, zu den Zeitpunkten  $t \approx 14\text{ s}$  bzw.  $t \approx 17\text{ s}$ ).

Aus den Abbildungen 6.33 und 6.34 ist auch das unterschiedliche Verhalten der Tanks bezüglich des Schwappens der Flüssigkeit deutlich zu sehen. Während beim Tank T3 nur kleine, hochfrequente Wellen erkennbar sind, entsteht beim Tank T13 ein ausgeprägtes Schwappen.

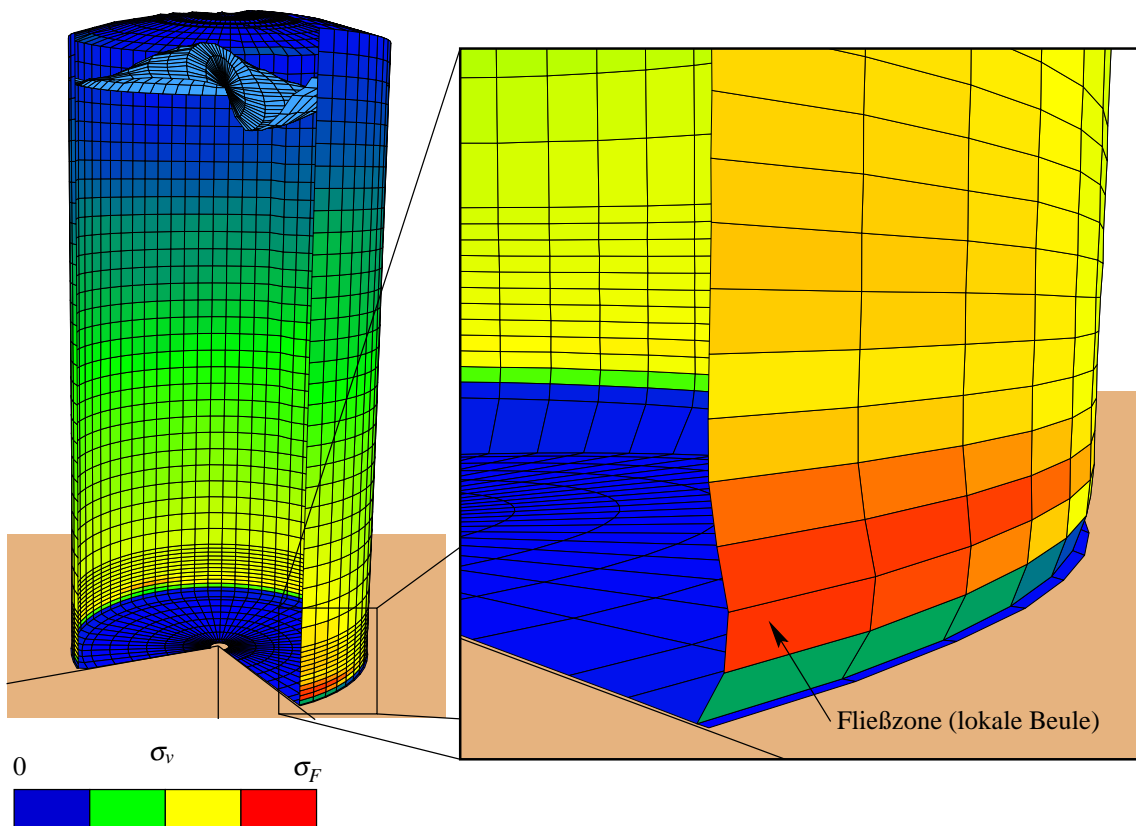
### 6.4.5 Schlußfolgerungen aus den Berechnungen

Das dynamische Antwortverhalten unverankerter, flüssigkeitsgefüllter Behälter unter Erdbeeinwirkung hängt vor allem von der Geometrie – also von der Schlankheit – des zu untersuchenden Behälters ab. Aber auch die Steifigkeit des Bodens spielt vor allem bei gedrunge-

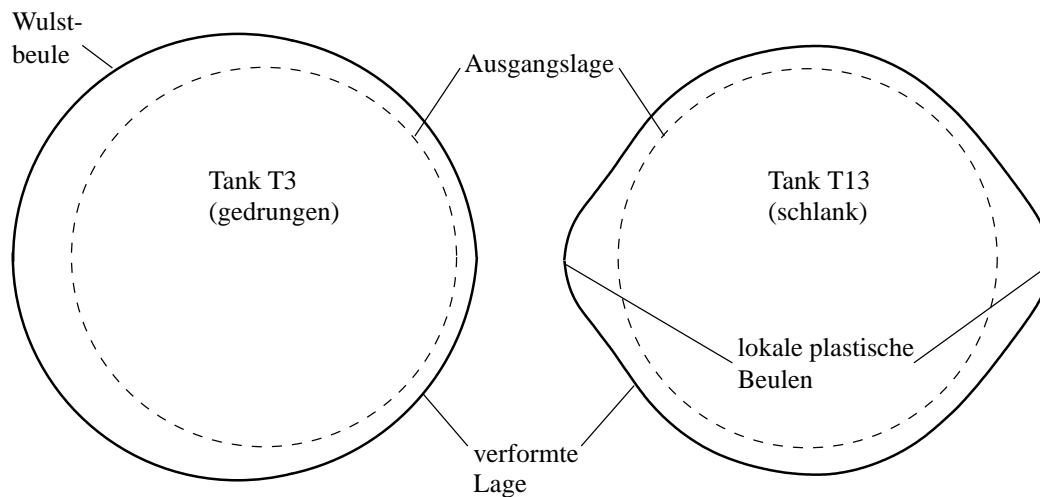




**Abb. 6.33:** Maximale Vergleichsspannung am verformten Tank T3 (5-fach überhöht)



**Abb. 6.34:** Maximale Vergleichsspannung am verformten Tank T13 (2-fach überhöht)



**Abb. 6.35:** Horizontale Knotenverschiebungen am Tank T3 und T13 am Ende der Simulation (100-fach überhöht)

nen Tanks eine wichtige Rolle. Schließlich beeinflusst auch die Erdbebeneinwirkung (Dauer, Frequenz, Intensität) die Strukturantwort.

Das Versagen der Behälter wird durch zwei verschiedene Beulformen geprägt. Während gedrungene Tanks über einen weiten Umfangsbereich eine plastische Wulstbeule aufweisen ("Elefantenfuß"), entstehen bei schlanken Behältern lokal eng begrenzte Beulen.

Für die numerische Simulation werden effiziente und robuste Verfahren benötigt. Vor allem die adaptive Schrittweitensteuerung bei der Zeitintegration muß automatisch die unterschiedlichen Strukturantworten erfassen und zu einer genauen Lösung bei einem minimierten numerischen Aufwand beitragen.

Die Untersuchungen zeigen auch, daß es aufgrund der enormen Parametersensitivität der Strukturantwort kaum möglich ist, mit Hilfe von detaillierten Parameterstudien ein vereinfachtes Bemessungsverfahren abzuleiten.

# Kapitel 7

## Zusammenfassung und Ausblick

In der vorliegenden Arbeit werden zahlreiche Methoden zur numerisch robusten und effizienten Simulation von großen Finite-Element-Systemen unter dynamischer Beanspruchung diskutiert. Im Mittelpunkt steht dabei die Simulation von Kontaktvorgängen sowie die nichtlineare Berechnung von komplexen Bauwerken einschließlich ihrer Gründung.

Die Auswahl eines geeigneten Zeitintegrationsverfahrens stellt sich als zentrale Aufgabe heraus. Die Anforderungen an den Algorithmus sind vielfältig. Er muß ein numerisch robustes und mechanisch korrektes Verhalten aufweisen und gleichzeitig effizient und flexibel einsetzbar sein.

Leider ist derzeit kein Verfahren bekannt, das alle Anforderungen optimal erfüllt. Einen guten Kompromiß stellt die *Generalized- $\alpha$  Methode* dar, die gegenüber der meistens benutzten *Newmark* Methode wesentlich verbesserte numerische Eigenschaften besitzt. Trotzdem bleibt die Verwendbarkeit für unterschiedliche Elementformulierungen erhalten. Bei nichtlinearen Berechnungen kann mit Hilfe der algorithmischen Dissipation ein numerisch robustes Verhalten erzielt werden.

Die Methode wird so modifiziert, daß sie allgemein zur einfachen Implementierung in unterschiedliche Finite-Element-Programme zur Verfügung steht. Durch diese Modifikation eignet sich das Verfahren auch für spezielle, semi-analytische Finite-Element-Methoden und für die nichtlineare dynamische Berechnung, bei der die nichtlinearen Anteile über Pseudo-Lasten erfaßt werden ("Iteration über die rechte Seite").

Es werden für zahlreiche Problemstellungen (z.B. Wellenausbreitung, Kontaktsimulation) geeignete Parametrisierungen sowie die Grenzen der Anwendbarkeit aufgezeigt.

Die Verbesserung der Genauigkeit der Simulation bei gleichzeitiger Minimierung des numerischen Aufwands erfordert eine adaptive Schrittweitensteuerung. Nachdem die Abschätzung der Schrittweite über den *Rayleigh-Quotienten* teilweise zu unbefriedigenden Ergebnissen führt, werden andere Methoden diskutiert. Dabei stellt sich der Fehlerschätzer nach *Zienkiewicz/Xie* als am besten geeignet heraus. Zur Berechnung der Schrittweite wird der absolute Fehler auf einen charakteristischen Wert bezogen. Hiermit können die Probleme der bisher verwendeten Bezugsgrößen umgangen werden. Für Systeme mit plötzlichen Steifigkeitsänderungen wird ein Steifigkeitsparameter in die Schrittweitensteuerung miteinbezogen.

Die aufwendige Finite-Element-Berechnung im Zeitbereich wird mit Hilfe der parallelen Programmierung beschleunigt. Im Gegensatz zu den üblichen parallelen Finite-Element-Programmen, bei denen das zu berechnende System in Teilgebiete zerlegt wird, erfolgt hier die

Parallelisierung auf Elementebene. Diese Methode ist bei der verwendeten semi-analytischen Element-Formulierung besonders vorteilhaft, da die numerisch aufwendige nichtlineare Berechnung auf der Elementebene durchgeführt wird. Die Gleichungslösung sowie andere, numerisch wenig aufwendige Operationen werden sequentiell abgearbeitet.

Dieses Vorgehen eignet sich deshalb für alle Finite-Element-Methoden, bei denen die Elementberechnung sehr viel mehr Rechenzeit beansprucht als die Gleichungslösung (Beispielsweise bei der p-Adaption).

Für den Einsatz auf Workstation-Clustern wird ein flexibles und einfach zu implementierendes System zum dynamischen Lastausgleich vorgestellt.

Die numerische Simulation von unverankerten, flüssigkeitsgefüllten Behältern unter Erdbebeneinwirkung stellt exemplarisch einen Anwendungsbereich der diskutierten Algorithmen dar. Das mechanische Antwortverhalten kann sich dabei auch auf die numerische Simulation auswirken, was sich unter anderem in einem jeweils charakteristischen Funktionsverlauf der adaptiven Zeitschrittlänge widerspiegelt.

Die Bereitstellung von geeigneten Lastmodellen wird am Beispiel der Erdbebeneinwirkung gezeigt. Mit einer exemplarischen Parameterstudie werden anschließend einige besondere Effekte unverankerter Tanks diskutiert. Insbesondere steht das völlig unterschiedliche Verhalten von schlanken und gedrunenen Behältern im Mittelpunkt. Aber auch die Einflüsse verschiedener Erdbeben sowie der Bodensteifigkeit werden dargestellt.

Es ist mit den in dieser Arbeit vorgestellten Methoden möglich, alle wichtigen Einflüsse zu erfassen, die bei der Berechnung von unverankerten Behältern vorhanden sind. Die dargestellten numerisch effizienten und robusten Algorithmen ermöglichen die Simulation von Effekten, die in diesem Kontext bisher nicht möglich waren (z.B. Beulvorgang).

Die vielfältigen Variationsmöglichkeiten verschiedener Parameter (Strukturmodell und Last) erschweren die sichere Prognose der Zuverlässigkeit der untersuchten Strukturen enorm. Mit dem gezeigten Modell können deshalb nur für konkrete Bauwerke unter klar definierten Bedingungen detaillierte Untersuchungen vorgenommen werden.

Eine umfangreiche Parameterstudie für alle Arten von Behältern, Bodenarten und Erdbebeneinwirkungen ist jedoch noch immer nicht möglich, da dies sowohl die Rechenzeit als auch die auszuwertende Datenmenge derzeit nicht zuläßt. Außerdem ist zu beachten, daß sich die Strukturantwort äußerst sensitiv auf manche Parameter verhält (z.B. Geometrie und Erdbebenfrequenz). Allerdings ist es fragwürdig, ob die Ergebnisse aus derartig aufwendigen Parameterstudien zu einem vereinfachten Bemessungsmodell führen können, da hierbei wichtige Effekte der nichtlinearen Dynamik unberücksichtigt bleiben.

# Notationsübersicht

Matrizen und Vektoren erscheinen im **Fettdruck**, sofern nicht die Indexschreibweise verwendet wird oder es sich um griechische Buchstaben handelt. Die Tensorindizes stehen rechts von der jeweiligen Variablen. Für eine beliebige Variable (hier:  $\mathbf{x}$ ) gilt außerdem:

$\mathbf{X}$	Ausgangszustand
$\mathbf{x}$	Grundzustand
$\hat{\mathbf{x}}$	Nachbarzustand
$\bar{\mathbf{x}}$	eingeprägte Größe
$\mathbf{x}_{,i}$	partielle Ableitung (nach $i$ )
${}^t\mathbf{x}$	Zustand zum Zeitpunkt $t$ ( $\equiv$ Grundzustand)
${}^{t+\Delta t}\mathbf{x}$	Zustand zum Zeitpunkt $t + \Delta t$ ( $\equiv$ Nachbarzustand)

## Mechanische Größen

### Arabische Buchstaben

$\mathbf{A}$	Vergrößerungsmatrix
$A$	Querschnittsfläche
$c$	Wellengeschwindigkeit
$\mathbf{C}$	Materialtensor (inkrementell), meist in Indexschreibweise
$\mathbf{C}$	Dämpfungsmatrix
$E, \mathbf{E}$	Elastizitätsmodul, Elastizitätstensor
$E$	Energie, mit Index für genaue Bezeichnung
$\mathbf{e}$	linearer Anteil der Verzerrungen
$f$	(Eigen-) Frequenz
$\mathbf{F}$	innere Knotenkräfte
$\mathbf{G}, \mathbf{g}, \hat{\mathbf{g}}$	Basisvektor im Ausgangs- Grund- und Nachbarzustand
$\mathbf{K}$	Steifigkeitsmatrix
$\mathbf{K}_{eff}$	effektive Steifigkeitsmatrix
$\mathbf{K}_L$	linearer Anteil der Steifigkeitsmatrix
$\mathbf{K}_{NL}$	nichtlinearer Anteil der Steifigkeitsmatrix
$\mathbf{K}_T$	Tangentialsteifigkeitsmatrix
$l$	Länge

<b>M</b>	Massenmatrix
$m$	Masse, auf Länge bezogen
$m$	Schalen-Biegemoment (mit Indizes für Ort und Richtung)
<b>n</b>	Normalenvektor
$n$	Schalen-Normalkraft (mit Indizes für Ort und Richtung)
<b>N</b>	Ansatzfunktionen
$p$	Impuls
$\bar{p}$	eingeprägte äußere Linien- bzw. Flächenlast
$\bar{\mathbf{P}}$	Lastvektor (eingeprägte bzw. äußere Lasten)
$\mathbf{P}_{eff}$	effektiver Lastvektor
$\mathbf{P}_{NL}$	nichtlinearer Pseudo-Lastvektor
$q$	Schalen-Querkraft (mit Indizes für Ort und Richtung)
<b>R, r, <math>\hat{\mathbf{r}}</math></b>	Ortsvektor im Ausgangs- Grund- und Nachbarzustand
<b>R</b>	Residuum
$R$	Radius (Maß)
$r$	Radius (Koordinate)
$s$	Steifigkeitsparameter
$\mathbf{s}$	2. Piola-Kirchhoff'scher Spannungstensor
$t$	Zeitpunkt, Wanddicke einer Schale
<b>t</b>	Spannungsvektor
$T$	Periodendauer
<b>t</b>	Spannungsvektor
<b>T</b>	Vektor der Trägheitskräfte (Massenkräfte)
<b>u, <math>\hat{\mathbf{u}}</math>, <math>\Delta \mathbf{u}</math></b>	Verschiebungen im Grund- und Nachbarzustand, Verschiebungsinkrement
$\dot{\mathbf{u}}$	Zeitableitung der Verschiebung ( $\equiv$ Geschwindigkeit)
$\ddot{\mathbf{u}}$	zweifache Zeitableitung der Verschiebung ( $\equiv$ Beschleunigung)
$V$	Volumen
$W$	Arbeit
<b>X, x, <math>\hat{\mathbf{x}}</math></b>	Koordinaten im Ausgangs- Grund- und Nachbarzustand
$z$	Koordinate in vertikaler Richtung (bei Rotationskörpern)
<b>z</b>	Zustandsvektor

## Griechische Buchstaben

$\alpha_f$	1. Skalierungsfaktor für das Generalized- $\alpha$ Verfahren
$\alpha_m$	2. Skalierungsfaktor für das Generalized- $\alpha$ Verfahren
$\alpha_R$	1. Parameter für die Rayleigh-Dämpfung
$\beta$	3. Skalierungsfaktor für das Generalized- $\alpha$ Verfahren
$\beta_R$	2. Parameter für die Rayleigh-Dämpfung
$\gamma$	4. Skalierungsfaktor für das Generalized- $\alpha$ Verfahren
$\delta$	allgemein für eine virtuelle Größe oder einen inkrementelle Differenz
$\Delta$	allgemein für eine inkrementelle Größe oder eine Differenz
$\varepsilon$	Verzerrungen (Dehnungen)
$\varepsilon$	Fehler
$\eta$	nichtlinearer Anteil der Verzerrungen

$\vartheta$	Koordinate in Umfangsrichtung (bei Rotationskörpern)
$\nu$	Querdehnzahl
$\xi$	physikalische Dämpfung
$\bar{\xi}$	numerische Dämpfung
$\rho$	Dichte eines Stoffes
$\rho$	Spektralradius
$\sigma$	Spannungen
$\sigma^{ij}$	Cauchy'scher Spannungstensor
$\tau$	Skalierungswert für die Schrittweitensteuerung
$\varphi$	Knotenverdrehung
$\omega$	Eigenfrequenz

## Parallele Programmierung

$E$	Effizienz
$\mathbf{H}$	spezieller Vektor zur Steuerung des Lastausgleichs
$n$	Problemgröße (abstrakter Wert)
$p$	Anzahl der Prozessoren
$S$	Speedup
$t$	relative Rechenzeit
$T$	absolute Rechenzeit
$w$	Wichtungparameter

## Erdbeben-Kenngrößen

$H$	Husid-Verteilung
$I_0$	Arias-Intensität
$T_s$	Dauer der Starkbebenphase
$\dot{x}$	Bodengeschwindigkeit
$\ddot{x}$	Bodenbeschleunigung





# Literaturverzeichnis

- [Amd67] G. Amdahl. Validity of the Single-Processor approach to achieving large-scale computing capabilities. In *Proceedings of the AFIPS Conference*, 483–485, New York, 1967.
- [AP97] F. Armero und E. Petöcz. On the formulation of stable time-stepping algorithms for contact problems. In D.R.J. Owen et al., Hrsg., *Fundamentals and Applications*, Jgg. 1 of *Computational Plasticity*, 865–872. CIMNE, 1997.
- [AR99] F. Armero und I. Romero. Dissipative integration algorithms for nonlinear elastodynamics. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM’99 (CD-ROM)*, TU München, 1999.
- [Arm99] F. Armero. Persönliches Gespräch, Tagung ECCM’99, TU München, September 1999.
- [Bac95] H. Bachmann. *Erdbebensicherung von Bauwerken*. Birkhäuser Verlag Basel, 1995.
- [Bat86] K.J. Bathe. *Finite-Elemente-Methoden. Matrizen und lineare Algebra, die Methode der finiten Elemente, Lösung von Gleichgewichtsbedingungen und Bewegungsgleichungen*. Springer-Verlag, 1986. Dt. Übersetzung des engl. Originaltitels: *Finite Element Procedures in Engineering Analysis* von P. Zimmermann.
- [BDF<sup>+</sup>93] S. Burkhardt, K.-D. Drey, V. Friedrich, M. Fritzsche und O. Nowak. *Parallele Rechnersysteme: Programmierung und Anwendung*. Praktische Informatik. Verlag Technik GmbH Berlin, 1993.
- [BDT95] O.A. Bauchau, G. Damilano und N.J. Theron. Numerical integration of nonlinear elastic multi-body systems. *International Journal for Numerical Methods in Engineering*, 38:2727–2751, 1995.
- [Bet93] J. Betten. *Kontinuumsmechanik. Elast-, Plasto- und Kriechmechanik*. Springer-Verlag, 1993.
- [BH83] T. Belytschko und T.J.R Hughes, Hrsg. *Computational Methods for Transient Analysis*. Computational Methods in Mechanics. North-Holland, 1983.

- [BM84] P.G. Bergan und E. Mollestad. Static and dynamic solution strategies in nonlinear analysis. In C. Taylor, E. Hinton und D.R.J. Owen, Hrsg., *Numerical Methods for Non-linear Problems*, Jgg. 2, 3–17. Pineridge Press, Swansea, 1984.
- [BM85] P.G. Bergan und E. Mollestad. An automatic time-stepping algorithm for dynamic problems. *Computer Methods in Applied Mechanics and Engineering*, 49:299–318, 1985.
- [Boe95] A. Boersma. *Parallele Finite-Elemente-Methoden in der Festkörpermechanik und ihre Lösung mit algebraischen Mehrgitterverfahren*. Dissertation, Fachbereich Mechanik der Technischen Hochschule Darmstadt, 1995.
- [Bol95] B.A. Bolt. *Erdbeben: Schlüssel zur Geodynamik*. Spektrum Akademischer Verlag, Heidelberg, 1995.
- [BP87] D.C. Barton und J.V. Parker. Finite Element analysis of the seismic response of anchored and unanchored liquid storage tanks. *Earthquake Engineering and Structural Dynamics*, 15:299–322, 1987.
- [BR99] R. Butenuth und E. Rehling. Bündnis für Arbeit: Robuste Lastverteilung für große und kleine Cluster. *Linux-Magazin*, Februar 1999.
- [Bre96] N. Breitsamter. *Ersatzmodelle zur Bestimmung der Schwingungsantwort von Gebäuden unter Anregung durch Bodenerschütterungen*. Dissertation, Lehrstuhl für Baumechanik, Technische Universität München, 1996.
- [BS96] A. Beguelin und V. Sunderam. Tools for monitoring, debugging, and programming in PVM. In A. Bode, J. Dongarra, T. Ludwig und V. Sunderam, Hrsg., *Parallel Virtual Machine – EuroPVM’96. Proceedings of the Third European PVM Conference*, Lecture Notes in Computer Science, 7–13. Springer Verlag, 1996.
- [Bul93] T. Bulenda. *Behandlung nichtlinearer Lösungspfade in der Statik mit Hybriden Krylov-Newton-Verfahren*. Dissertation, Lehrstuhl für Statik, Technische Universität München, 1993.
- [BW73] K.J. Bathe und E.L. Wilson. Stability and accuracy analysis of direct integration methods. *Earthquake engineering and structural dynamics*, 1:283–291, 1973.
- [BW99] H. Bachmann und T. Wenk. Schwächen anstatt Verstärken: Die Erdbebensanierung eines Flüssiggas-Grosstanks. In *Festschrift zum 60. Geburtstag von Prof. Dr. Edoardo Anderheggen*, 1–10, Institut für Baustatik und Konstruktion der ETH Zürich, 1999.
- [CA96] J.M. Cela und J.M. Alfonso. Parallelization of the SPAI preconditioner in a Master-Slave configuration. In A. Bode, J. Dongarra, T. Ludwig und V. Sunderam, Hrsg., *Parallel Virtual Machine – EuroPVM’96. Proceedings of the Third European PVM Conference*, Lecture Notes in Computer Science, 150–157. Springer Verlag, 1996.

- [CC99] C.-K. Choi und H.-J. Chung. Control of global relative error of spatial-temporal discretization in dynamic analysis. In H.A. Mang und F.G. Rammerstorfer, Hrsg., *Discretisation Methods in Structural Mechanics II*, Jgg. 68 of *Solid Mechanics and its Applications*, 107–114. IUTAM/IACM Symposium 1997, Kluwer Academic Publishers, 1999.
- [CCEA95] Y.P. Chien, F. Carpenter, A. Ecer und H.U. Akay. Load-balancing for parallel computation of fluid dynamics problems. *Computer Methods in Applied Mechanics and Engineering*, 120:119–130, 1995.
- [CF88] K.-N. Chiang und R.E. Fulton. Nonlinear dynamics methods for parallel computers. In K.M. Will, Hrsg., *Computing in Civil Engineering: Microcomputers to Supercomputers. Proceedings of the Fifth Conference*. Society for Computer Applications in Engineering, Planning and Architecture (CEPA), ASCE, New York, 1988.
- [CFW99] H. Cramer, R. Findeiß und W. Wunderlich. Elastic–plastic consolidation analysis of saturated soil by a time discontinuous method. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM’99 (CD-ROM)*, TU München, 1999.
- [CH93] J. Chung und G.M. Hulbert. A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized- $\alpha$  method. *Journal of Applied Mechanics*, 60:371–375, 1993.
- [CJ98] M.A. Crisfield und G. Jelenic. An invariant Energy-Momentum conserving procedure for dynamics of 3D beams. In S. Idelson, E. Onate und E. Dvorkin, Hrsg., *Computational Mechanics – New trends and applications. Conference Proceedings of the Fourth World Congress on Computational Mechanics (CD-ROM)*, Barcelona, Spanien, 1998. CIMNE.
- [CJG99] M.A. Crisfield, G. Jelenic und U. Galvanetto. Finite Elements with nonlinear statics and dynamics. In H.A. Mang und F.G. Rammerstorfer, Hrsg., *Discretisation Methods in Structural Mechanics II*, Jgg. 68 of *Solid Mechanics and its Applications*, 91–98. IUTAM/IACM Symposium 1997, Kluwer Academic Publishers, 1999.
- [CL94] J. Chung und J.M. Lee. A new family of explicit time integration methods for linear and non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 37:3961–3976, 1994.
- [CL98] V. Chawla und T.A. Laursen. Energy consistent algorithms for frictional contact problems. *International Journal for Numerical Methods in Engineering*, 42:799–827, 1998.
- [CP93] R.W. Clough und J. Penzien. *Dynamics of Structures*. Civil Engineering Series. Mc Graw-Hill, Inc., 2. Auflage, 1993.
- [CP97] J.P. Conte und B.F. Peng. Fully nonstationary analytical earthquake ground-motion model. *Journal of Engineering Mechanics*, 1997.

- [CPWJ91] H. Cramer, M.J. Prabucki, W. Wunderlich und H.L. Jessberger. Tragfähigkeit von Gründungen im feinkörnigen Baugrund bei dynamischen Belastungen. Berichte zum Teilprojekt C2, SFB 151, 1984-1991.
- [Cri91] M.A. Crisfield. *Non-linear Finite Element analysis of solids and structures. Volume 1: Essentials*. Wiley and Sons, Inc., 1991.
- [Cri97] M.A. Crisfield. *Non-linear Finite Element analysis of solids and structures. Volume 2: Advanced Topics*. Wiley and Sons, Inc., 1997.
- [CRSW96] H. Cramer, M. Rudolph, G. Steinl und W. Wunderlich. A hierarchical adaptive Finite Element strategy for elastic-plastic problems. In B.H.V. Topping, Hrsg., *Advances in Finite Element Technology*, 151–159, Edinburgh, 1996. Civil-Comp Press.
- [CS94] M.A. Crisfield und J. Shi. A co-rotational element/time-integration strategy for non-linear dynamics. *International Journal for Numerical Methods in Engineering*, 37:1897–1913, 1994.
- [Dam98] F.B. Damjanic. On non-linear dynamic thin shell analysis. In S. Idelson, E. Onate und E. Dvorkin, Hrsg., *Computational Mechanics – New trends and applications. Conference Proceedings of the Fourth World Congress on Computational Mechanics (CD-ROM)*, Barcelona, Spanien, 1998. CIMNE.
- [dBM95] R. de Borst und C. Meyer. Numerische Probleme bei nichtlinearem Tragwerksverhalten. In G. Mehlhorn, Hrsg., *Rechnerorientierte Baumechanik*, Der Ingenieurbau, 427–488. Ernst u. Sohn, 1995.
- [DDNR96] R. Diekmann, U. Dralle, F. Neugebauer und T. Römke. PadFEM: A Portable Parallel FEM-Tool. In *HPCN Europe 96*, Brüssel, Belgien, April 1996.
- [Dem97] M. Deml. *Ein Finite-Element-Konzept zur Traglastanalyse imperfektionssensitiver Schalenstrukturen*. Dissertation, Lehrstuhl für Statik, Technische Universität München, 1997.
- [DGMS96] J.J. Dongarra, G.A. Geist, R. Manchek und V.S. Sunderam. Integrated PVM framework supports heterogeneous network computing. In A.Y. Zomaya, Hrsg., *Parallel Computing. Paradigms and Applications*, 435–451. International Thomson Computer Press, 1996.
- [DIN79] DIN 4119: Oberirdische zylindrische Flachboden-Tankbauwerke aus metallischen Werkstoffen. Deutsche Norm, 1979.
- [Din88] D. Dinkler. *Stabilität dünner Flächentragwerke bei zeitabhängigen Einwirkungen*. Habilitationsschrift. Institut für Statik der Technischen Universität Braunschweig, 1988.
- [DNO<sup>+</sup>97] G.A. Duffett, L. Neamtu, E. Onate, J. Rojek und F. Zarate. STAMPAR: A parallel processing approach for the analysis of sheet stamping problems. In D.R.J. Owen et al., Hrsg., *Fundamentals and Applications*, Jgg. 1 of *Computational Plasticity*, 1782–1798. CIMNE, 1997.

- [DS96] A. Deylami und Sarrafzadeh. Seismic analysis of cylindrical liquid storage tanks. In *Proceedings of the eleventh World Conference on Earthquake Engineering*. Elsevier Science Ltd, 1996.
- [EC893] Eurocode 8: Earthquake resistant design of structures. 2. Vorentwurf zur Europäischen Erdbebennorm, 1993.
- [Eib95] S. J. Eibl. *Schockwellenbeanspruchung von Stahlbetonwänden durch Kontakt-Detonation*. Dissertation, Lehrstuhl für Statik, Technische Universität München, 1995.
- [EM99] A. Elenas und K. Meskouris. Korrelation zwischen seismischen Beschleunigungsparametern und nichtlinearem Tragwerksverhalten. *Bautechnik*, 665–672, August 1999.
- [EQE92] EQE. International Inc., Oakland. The Landers and Big Bear Earthquakes of June 28, 1992. Internet: <http://www.eqe.com/publications/bigbear/bigbear.html>, 1992.
- [FB96] G. Feltrin und H. Bachmann. A technique for the rational approximation of complex functions and its application in soil- and fluid-structure interaction problems. In *Proceedings of the Third European Conference on Structural Dynamics (Eurodyn 96)*. A.A. Balkema, Rotterdam, Juni 1996.
- [FES98] M. Field, Y. Ezawa und T Satake. Solving large structural analysis problems using a parallel iterative solver. In S. Idelson, E. Onate und E. Dvorkin, Hrsg., *Computational Mechanics – New trends and applications. Conference Proceedings of the Fourth World Congress on Computational Mechanics (CD-ROM)*, Barcelona, Spanien, 1998. CIMNE.
- [Fis81] D. Fischer. Ein Vorschlag zur erdbebensicheren Bemessung von flüssigkeitsgefüllten zylindrischen Tankbauwerken. *Der Stahlbau*, 13–20, 1981.
- [FP90] C. Farhat und E. Pramono. Mapping, solving and visualizing Finite Element problems on the Connection Machine. In R.A. Adey, Hrsg., *Parallel Processing in Engineering Applications. Proceedings of the first International Conference on Parallel Processing for Computational Mechanics*, 95–107. Computational Mechanics Publications and Springer-Verlag, 1990.
- [FR94] C. Farhat und F.-X. Roux. Implicit parallel processing in structural mechanics. *Computational Mechanics Advances*, 2:1–124, Juni 1994.
- [FRS90] F.D. Fischer, F.G. Rammerstorfer und K. Scharf. Earthquake resistant design of anchored and unanchored liquid storage tanks under three-dimensional earthquake excitation. In G.I. Schuëller, Hrsg., *Structural Dynamics: Recent Advances*, 317–371. Springer-Verlag Berlin, 1990.
- [FS88] F.D. Fischer und R. Seeber. Dynamic response of vertically excited liquid storage tanks considering liquid-soil interaction. *Earthquake Engineering and Structural Dynamics*, 16:329–342, 1988.

- [FS90] P.A. Friberg und C.A.T. Susch. *A User's Guide to Strongmo: Version 1.0 of NCEER's Strong-Motion Data Access Tool for PCs and Terminals*. Lamont-Doherty Geological Observatory, Columbia University, Palisades, New York 10964, August 1990.
- [G<sup>+</sup>94] A. Geist et al. *PVM: Parallel Virtual Machine. A Users Guide and Tutorial for Networked Parallel Computing*. Scientific Engineering Computation. MIT Press, Massachusetts Institute of Technology, 1994.
- [Gal38] G. Galilei. *Discorsi e dimonstrazioni matematiche*. Leiden, 1638.
- [GCM99] G.A. Galarza, J. Carrera und A. Medina. Computational techniques for optimization of problems involving non-linear transient simulations. *International Journal for Numerical Methods in Engineering*, 45:319–334, 1999.
- [Gea71] C.W. Gear. *Numerical initial value problems in ordinary differential equations*. Prentice–Hall, Inc., 1971.
- [Goe91] W. Goebel. *Reduktion numerischer Modelle in der nichtlinearen Strukturdynamik mit globalen Ansätzen*. Dissertation, Lehrstuhl für Statik, Technische Universität München, 1991.
- [Gro95] D. Groß. Werkstoffmechanik. In G. Mehlhorn, Hrsg., *Werkstoffe, Elastizitätstheorie*, Der Ingenieurbau, 133–163. Ernst u. Sohn, 1995.
- [Gru97] H. Grundmann. Skriptum Baudynamik. Lehrstuhl für Baumechanik, Technische Universität München, 1997.
- [Gus88] J.L. Gustafson. Reevaluating Amdahl's Law. *Communications of the ACM*, (31):532–533, Mai 1988.
- [HA96] S.-H. Hsieh und J.F. Abel. Computer-aided parallel structural dynamic simulations in a workstation network. In C.-K. Choi et al., Hrsg., *Proceedings of the Third Asian-Pacific Conference on Computational Mechanics*, Jgg. 3, 2383–2388, September 1996.
- [Ham80] E. Hampe. *Flüssigkeitsbehälter, Band 1: Grundlagen*. Verlag Wilhelm Ernst & Sohn, Berlin, 1980.
- [Har96] U. Harms. Parallelisierte Software verkürzt Entwicklungszeiten. *VDI-Nachrichten*, (28):13, Juli 1996.
- [HCL78] T.J.R. Hughes, T.K. Caughey und W.K. Liu. Transient Finite Element formulations that preserve energy. *Journal of Applied Mechanics*, 1978.
- [Hei94] H.-U. Heiss. *Prozessorzuteilung in Parallelrechnern*. Reihe Informatik, Band 98. BI-Wissenschaftsverlag Mannheim, 1994.
- [Hem96] R. Hempel. The status of the MPI Message-Passing Standard and its relation to PVM. In A. Bode, J. Dongarra, T. Ludwig und V. Sunderam, Hrsg., *Parallel Virtual Machine – EuroPVM'96. Proceedings of the Third European PVM Conference*, Lecture Notes in Computer Science, 14–21. Springer Verlag, 1996.

- [HF98] A. Hamsen und Rochinha F. A transient domain decomposition method for the analysis of structural dynamic problems. In S. Idelson, E. Onate und E. Dvorkin, Hrsg., *Computational Mechanics – New trends and applications. Conference Proceedings of the Fourth World Congress on Computational Mechanics (CD-ROM)*, Barcelona, Spanien, 1998. CIMNE.
- [HH78] H.M. Hilber und T.J.R Hughes. Collocation, dissipation and ‘Overshoot’ for time integration schemes in structural dynamics. *Earthquake engineering and structural dynamics*, 6:99–117, 1978.
- [HH79] G.W. Housner und M.A. Haroun. Vibration tests of full-scale liquid storage tanks. In *Proceedings of the 2nd US National Conference on Earthquake Engineering*, 137–145, 1979.
- [HH88] T.J.R. Hughes und G. Hulbert. Space–Time Finite Element Methods for elastodynamics: Formulations and error estimates. *Computer Methods in Applied Mechanics and Engineering*, (66):339–363, 1988.
- [HH90] G. Hulbert und T.J.R. Hughes. Space–Time Finite Element Methods for second order hyperbolic equations. *Computer Methods in Applied Mechanics and Engineering*, (84):327–348, 1990.
- [HHHP89] C. Hoff, T.J.R. Hughes, G. Hulbert und P.J. Pahl. Extended comparison of the Hilber-Hughes-Taylor  $\alpha$ -method and the  $\Theta_1$ -Method. *Computer Methods in Applied Mechanics and Engineering*, 87–93, 1989.
- [HHT77] H.M. Hilber, J.R. Hughes und R.L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake engineering and structural dynamics*, 5:283–292, 1977.
- [Hos87] D. Hosser. Realistische seismische Lastannahmen für Bauwerke: Ergebnisse einer interdisziplinären Forschungsarbeit. *Bauingenieur*, (62):567–574, 1987.
- [HP88a] C. Hoff und P.J. Pahl. Development of an implicit method with numerical dissipation from a generalized single-step algorithm for structural dynamics. *Computer Methods in Applied Mechanics and Engineering*, 367–385, 1988.
- [HP88b] C. Hoff und P.J. Pahl. Practical performance of the  $\Theta_1$ -method and comparison with other dissipative algorithms in structural dynamics. *Computer Methods in Applied Mechanics and Engineering*, 87–110, 1988.
- [HRS91] E. Hampe, C. Riedel und J. Schwarz. Zur Berücksichtigung der Boden-Bauwerk-Wechselwirkung bei der seismischen Untersuchung von Tragwerken. *Bauingenieur*, (66):503–515, 1991.
- [HS98] U. Hornung und H. Saal. Ergebnisse von Beulversuchen mit zwei Großtanks. *Stahlbau*, (6):408–413, 1998.
- [Hug87] Th.J.R. Hughes. *The Finite Element Method. Linear static and dynamic Finite Element analysis*. Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1987.

- [Hul94] G.M. Hulbert. A unified set of single-step asymptotic annihilation algorithms for structural dynamics. *Computer methods in Applied Mechanics and Engineering*, 113:1–9, 1994.
- [HWC<sup>+</sup>96] E. Hampe, W. Wunderlich, H. Cramer, B. Kabalan, J. Schwarz und C. Seiler. Berechnung von Behälter- und Silotragwerken unter seismischer Einwirkung. 1. Bericht zum DFG Forschungsvorhaben Wu 67/24-1 und Ha 1839/1-1, 1996.
- [IST97] P. Ivanyi, J. Sziveri und B.H.V. Topping. Parallel non-linear Finite Element analysis of reinforced concrete plates under dynamic loading. In G. Thierauf, Hrsg., *Engineering Structures under Earthquake Loading*, 69–84, Edinburgh, 1997. Civil-Comp Press.
- [Jül99] Forschungszentrum Jülich. Supercomputer am Forschungszentrum Jülich. Internet: <http://www.kfa-juelich.de/forschung/supercomputer.html>, Dezember 1999.
- [JT98] A.A. Johnson und T.E. Tezduyar. Parallel computing methods for large-scale 3D simulations of Fluid-Object Interactions. In S. Idelson, E. Onate und E. Dvorkin, Hrsg., *Computational Mechanics – New trends and applications. Conference Proceedings of the Fourth World Congress on Computational Mechanics (CD-ROM)*, Barcelona, Spanien, 1998. CIMNE.
- [Kar93] W. Karl. *Parallele Prozessorarchitekturen: Codegenerierung für superskalare, superpipelined und VLIW-Prozessoren*. Reihe Informatik, Band 93. BI-Wissenschaftsverlag Mannheim, 1993.
- [KC99] D. Kuhl und M.A. Crisfield. Energy-conserving and decaying algorithms in non-linear structural dynamics. *International Journal for Numerical Methods in Engineering*, 45:569–599, 1999.
- [Kla96] O. Klaas. *Adaptive Finite-Element-Methoden für MIMD-Parallelrechner zur Behandlung von Strukturproblemen mit Anwendung auf Stabilitätsprobleme*. Dissertation, Fachbereich Bauingenieur- und Vermessungswesen der Universität Hannover, Juli 1996.
- [Kle88] H.-H. Klein. Realistische seismische Lastannahmen für Bauwerke: Alternative Lastbeschreibung mittels Fourierspektren. *Bauingenieur*, (63):221–228, 1988.
- [Kli97] Deutsches Klimarechenzentrum. Paralleles Rechnen am DKRZ. Internet: <http://www.dkrz.de/dkrz/parallel/parallel.html>, Oktober 1997.
- [KLMW98] D. Kasper, H. Lange, B. Müller und T. Winterstetter. Neuere Beispiele großer zylindrischer Behälter. *Stahlbau*, (6):404–407, 1998.
- [KR96] D. Kuhl und E. Ramm. Constraint Energy Momentum Algorithm and its application to non-linear dynamics of shells. *Computer Methods in Applied Mechanics and Engineering*, 136:293–315, 1996.



- [KR98] D. Kuhl und E. Ramm. Generalized Energy Momentum Method for non-linear dynamics of shells. In S. Idelson, E. Onate und E. Dvorkin, Hrsg., *Computational Mechanics – New trends and applications. Conference Proceedings of the Fourth World Congress on Computational Mechanics (CD-ROM)*, Barcelona, Spanien, 1998. CIMNE.
- [Kre96] M. Kreienmeyer. *Kopplung von Rand- und Finite-Element-Methoden für ebene Elastoplastizität mit Implementierung auf Parallelrechnern*. Dissertation, Fachbereich Bauingenieur- und Vermessungswesen der Universität Hannover, Juni 1996.
- [Kuh96] D. Kuhl. *Stabile Zeitintegrationsalgorithmen in der nichtlinearen Elastodynamik dünnwandiger Tragwerke*. Dissertation, Institut für Baustatik der Universität Stuttgart, 1996.
- [LB97] L. Lämmer und M. Burghardt. Parallel mesh generation. In G. Bilardi et al., Hrsg., *Solving irregularly structured problems in parallel: 4th international symposium irregular 97*, Lecture Notes in Computer Science, 1–12. Springer Verlag, 1997.
- [Li96] X.D. Li. *Adaptive Finite Element procedures in structural dynamics*. Dissertation, Departement of Structural Mechanics, Chalmers University of Technology, Göteborg, Schweden, 1996.
- [Lin84] M. Link. *Finite Elemente in der Statik und Dynamik*. Teubner Verlag, Stuttgart, 1984.
- [Läm97a] L. Lämmer. *Parallelisierung von Anwendungen der Finite-Element-Methode im Bauingenieurwesen*. Habilitationsschrift. Institut für Numerische Methoden und Informatik im Bauwesen, Technische Hochschule Darmstadt, 1997.
- [LM97b] L. Lämmer und U. Meißner. Dynamic load balancing in parallel adaptive Finite Element computations. In D.R.J. Owen et al., Hrsg., *Fundamentals and Applications*, Jgg. 1 of *Computational Plasticity*, 1798–1805. CIMNE, 1997.
- [LMR<sup>+</sup>96] P. Luksch, U. Maier, S. Rathmayer, M. Weidmann und F. Unger. Parallelization of a state-of-the-art industrial CFD package for execution on networks of workstations and massively parallel processors. In A. Bode, J. Dongarra, T. Ludwig und V. Sunderam, Hrsg., *Parallel Virtual Machine – EuroPVM'96. Proceedings of the Third European PVM Conference*, Lecture Notes in Computer Science, 45–52. Springer Verlag, 1996.
- [LT94] P. Le Tallec. Domain decomposition methods in computational mechanics. *Computational Mechanics Advances*, 1(2):121–220, Februar 1994.
- [LZW93] X.D. Li, L.F. Zeng und N.E. Wiberg. A simple local error estimator and an adaptive time-stepping procedure for direct integration method in dynamic analysis. *Communications in Numerical Methods in Engineering*, 9:273–292, 1993.

- [M<sup>+</sup>88] K. Meskouris et al. *Mikrocomputerunterstützte Erdbebenuntersuchung von Tragwerken*. SFB 151 Tragverhalten und Tragfähigkeit von Baukonstruktionen unter dynamischen Einwirkungen. Ruhr-Universität Bochum, Februar 1988.
- [Mal96] P.K. Malhotra. Seismic uplifting of flexibly supported liquid-storage tanks. In *Proceedings of the eleventh World Conference on Earthquake Engineering*. Elsevier Science Ltd, 1996.
- [Man86] G.C. Manos. Earthquake tank-wall stability of unanchored tanks. *Journal of Strutural Engineering*, 112(8):1863–1880, 1986.
- [Mey97] S. Meynen. *Zur numerischen Simulation von nichtlinearen Problemen der Kontinuumsmechanik auf Parallelrechnern*. Dissertation, Fachbereich Mechanik der Technischen Hochschule Darmstadt, 1997.
- [Mey98] L. Meyer. *Formoptimierung in der Strukturdynamik*. Dissertation, Institut für Baumechanik und Numerische Mechanik, Universität Hannover, 1998.
- [MGC82] D.C. Ma, J. Gvildys und Y.W. Chang. Seismic behavior of liquid-filled shells. *Nuclear Engineering and Design*, (70):437–455, 1982.
- [MKCB99] K. Meskouris, C. Könke, R. Chudoba und V. Bettzieche. Seismic safety of Rockfill Dams. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM'99 (CD-ROM)*, TU München, 1999.
- [MN90] L. Ma und W.A. Nash. Nonlinear uplift analysis of a liquid storage tank. In W.B. Kräzig et al., Hrsg., *Structural Dynamics. Proceedings of the European Conference on Structural Dynamics (Eurodyn '90)*, Bochum, 503–508. A.A. Balkema, Rotterdam, 1990.
- [MPA97] H.P. Mouzakis, M. Papadrakakis und A.S. Alevridis. Three dimensional nonlinear analysis of building pounding during earthquakes. In G. Thierauf, Hrsg., *Engineering Structures under Earthquake Loading*, 43–48, Edinburgh, 1997. Civil-Comp Press.
- [MPI95] Message Passing Interface Forum MPIF. *MPI: A Message-Passing Interface Standard (Version 1.1)*. University of Tennessee, Knoxville, Juni 1995.
- [MPI97] Message Passing Interface Forum MPIF. *MPI-2: A Message-Passing Interface Standard*. University of Tennessee, Knoxville, Juli 1997.
- [Mär94] C. Martin. *Rechnerarchitektur: Struktur, Organisation, Implementierungstechnik*. Studienbücher Informatik. Carl Hanser Verlag München, 1994.
- [MS79] H. Matthies und G. Strang. The solution of nonlinear Finite Element equations. *International Journal for Numerical Methods in Engineering*, 14:1613–1626, 1979.
- [MV94] P.K. Malhotra und A.S. Veletsos. Beam model of base-uplifting analysis of cylindrical tanks. *Journal of Strutural Engineering*, 120(12):3471–3547, 1994.

- [NC82] A. Niwa und R.W. Clough. Buckling of cylindrical liquid-storage tanks under earthquake loading. *Earthquake Engineering and Structural Dynamics*, 10:107–122, 1982.
- [NCE98] Northern California Earthquake Data Center NCEDC. Internet: <http://quake.geo.berkeley.edu> (University of California, Berkeley Seismological Laboratory), 1998.
- [New59] N.M. Newmark. A method of computation for structural dynamics. *Journal of the Engineering Mechanics Division, ASCE*, (EM 3):67–94, 1959.
- [NH92] K. Nasitta und H. Hagel. *Finite Elemente. Mechanik, Physik und nichtlineare Prozesse*. Springer-Verlag Berlin, 1992.
- [NNDR77] E. Naug, O.S. Nguyen und A.L. De Rouvray. An improved energy conserving implicit time integration algorithm for nonlinear dynamic structural analysis. In *Proceedings of the Fourth Conference on Structural Mechanics in Reactor Technology*, San Francisco, 1977.
- [NS99] J. Neumann und K. Schweizerhof. Analysis of shell structures under transient loading using adaptivity in time and space. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM’99 (CD-ROM)*, TU München, 1999.
- [OG99] J.C.G. Orden und J.M. Goicolea. Non-frictional contacts in multibody dynamics with conserving schemes. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM’99 (CD-ROM)*, TU München, 1999.
- [Pee88] R. Peek. Analysis of unanchored liquid storage tanks under lateral loads. *Earthquake Engineering and Structural Dynamics*, 16:1087–1100, 1988.
- [Pfa89] D. Pfaffinger. *Tragwerksdynamik*. Springer-Verlag Wien, 1989.
- [PL94] J. Pruyne und M. Livny. Providing resource management services to parallel applications. In *Proceedings of the Second Workshop on Environments and Tools for Parallel Scientific Computing*, Mai 1994.
- [PP93] M. Papadrakis und G. Pantazopoulos. A survey of quasi-newton methods with reduced storage. *International Journal for Numerical Methods in Engineering*, 36:1573–1596, 1993.
- [PSS90] H.J. Pradlwarter, G.I. Schuëller und R.J. Scherer. Earthquake Loading. In G.I. Schuëller, Hrsg., *Structural Dynamics: Recent Advances*, 28–51. Springer-Verlag Berlin, 1990.
- [PW94] W.D. Pilkey und W. Wunderlich. *Mechanics of structures: Variational and computational methods*. CRC Press, 1994.

- [RALS99] T. Rottner, G. Alefeld, I. Lenhardt und K. Schweizerhof. Traversing limit points in nonlinear structural analysis using efficient iterative equation solvers. In H.A. Mang und F.G. Rammerstorfer, Hrsg., *Discretisation Methods in Structural Mechanics II*, Jgg. 68 of *Solid Mechanics and its Applications*, 99–106. IUTAM/IACM Symposium 1997, Kluwer Academic Publishers, 1999.
- [Ram81] E. Ramm. Strategies for tracing the nonlinear response near limit points. In W. Wunderlich, E. Stein und K.-J. Bathe, Hrsg., *Nonlinear Finite Element Analysis in Structural Mechanics; Proceedings of the Europe-U.S. Workshop Ruhr-Universität Bochum, 28.-31. Juli 1980*, 553–571. Springer-Verlag, 1981.
- [Rap98] M. Rapolder. Parallelisierung semi-analytischer Ringelemente für heterogene Workstation-Cluster. Vortrag bei: Seminar FE im Schnee 6, Hirschegg (Kleinwalsertal), Januar 1998.
- [Red87] W. Redanz. *Nichtlineares dynamisches Verhalten und Stabilität von Systemen der Strukturmechanik*. Dissertation, Institut für konstruktiven Ingenieurbau, Ruhr-Universität Bochum, 1987.
- [Ric97] J. Riccius. *Adaptive Methoden zur statischen und dynamischen Analyse von Flächentragwerken mit linearen Finiten Elementen*. Dissertation, Institut für Mechanik TH Karlsruhe, 1997.
- [RK98] M. Rücker und M. Krafczyk. Parallelisierung eines FE-Programms für hohe Polynomgrade. In A. Grosche et al., Hrsg., *Forum Bauinformatik, Junge Wissenschaftler forschen*, Fortschrittberichte VDI, Reihe 4, Nr. 147, 125–132, Düsseldorf, 1998. VDI-Verlag GmbH.
- [RKR98] M. Rücker, M. Krafczyk und E. Rank. A parallel p-version FE-approach for structural engineering. In B.H.V. Topping, Hrsg., *Advances in Computational Mechanics with High Performance Computing*, 73–78, Edinburgh, 1998. Civil-Comp Press.
- [RRDB99] E. Rank, M. Rücker, A. Düster und H. Bröker. The efficiency of the p-version Finite Element Method. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM'99 (CD-ROM)*, TU München, 1999.
- [RS97] R. Rannacher und E. Stein. Finite Elemente: Die Ideen. *Spektrum der Wissenschaft*, 90–98, März 1997.
- [RS99] T. Rottner und K. Schweizerhof. Transient FE-Analyses of silo buckling behaviour using parallel computers. In B.H.V. Topping, Hrsg., *Developments in Computational Mechanics with High Performance Computing*, 79–86, Edinburgh, 1999. Civil-Comp Press.
- [RSLA98] T. Rottner, K. Schweizerhof, I. Lenhardt und G. Alefeld. Fully parallel nonlinear Finite Element computations using the pre-conditioned Lanczos Method for

- solving linear equations. In B.H.V. Topping, Hrsg., *Advances in Computational Mechanics with High Performance Computing*, 95–101, Edinburgh, 1998. Civil-Comp Press.
- [RW99] M. Rapolder und W. Wunderlich. Non-linear dynamic finite element analysis of coupled large-scale structural models. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM’99 (CD-ROM)*, TU München, 1999.
- [SA95] H. Saal und E. Andelfinger. Seismische Auslegung von verankerten, zylindrischen, oberirdischen Flachbodentankbauwerken aus Stahl. *Stahlbau*, (4):97–103, 1995.
- [San96] J. Sansour. *Nichtlineare Dynamik geometrisch exakter Schalen und Stäbe: Theorie, Numerik finiter Elemente und Integrationsalgorithmen*. Dissertation, Fachbereich Mechanik der Technischen Hochschule Darmstadt, 1996.
- [SB95] E. Stein und F.-J. Barthold. Elastizitätstheorie. In G. Mehlhorn, Hrsg., *Werkstoffe, Elastizitätstheorie*, Der Ingenieurbau, 165–428. Ernst u. Sohn, 1995.
- [Sch89] K. Schweizerhof. *Quasi-Newton Verfahren und Kurvenverfolgungsalgorithmen für die Lösung nichtlinearer Gleichungssysteme in der Strukturmechanik*. Dissertation, Universität Fridericiana Karlsruhe (TH), 1989.
- [Sch90] K. Scharf. Beiträge zur Erfassung des Verhaltens von erdbebenerregten, oberirdischen Tankbauwerken. In *Fortschrittsberichte VDI*, Reihe 4, Nr. 97, Düsseldorf, 1990. VDI-Verlag GmbH.
- [Sch94] T. Schnekenburger. *Adaptive Lastverteilung für parallele Programme*. Dissertation, Fakultät für Informatik der Technischen Universität München, 1994.
- [Sch95] T. Schnekenburger. The ALDY Load Distribution System. SFB-Bericht Nr. 342/11/95 A, Institut für Informatik der Technischen Universität München, 1995. Sonderforschungsbereich 342: Methoden und Werkzeuge für die Nutzung paralleler Rechnerarchitekturen.
- [Sch96a] J. Schüle. Parallel Lanczos Algorithm on a CRAY-T3D Combining PVM and SHMEM Routines. In A. Bode, J. Dongarra, T. Ludwig und V. Sunderam, Hrsg., *Parallel Virtual Machine – EuroPVM’96. Proceedings of the Third European PVM Conference*, Lecture Notes in Computer Science, 158–165. Springer Verlag, 1996.
- [Sch96b] B. Schäpertöns. *Über die Wellenausbreitung im Baugrund und deren Einfluß auf das Tragverhalten von flüssigkeitsgefüllten Behältern*. Dissertation, Lehrstuhl für Statik, Technische Universität München, 1996.
- [SLP94] B. Schnor, H. Langendörfer und S. Petri. Einsatz neuronaler Netze zur Lastbalancierung in Workstationclustern. In H. Langendörfer, Hrsg., *Praxisorientierte Parallelverarbeitung - Beiträge zum 3. Workshop über Wissenschaftliches Rechnen*, 154–165, Braunschweig, Oktober 1994. Hanser Verlag.

- [SSR94] E. Stein, B. Seifert und W. Rust. *Adaptive Verfeinerung von Finite-Element-Netzen für Stabilitätsprobleme von Flächentragwerken*. Forschungs- und Seminarberichte aus dem Bereich der Mechanik der Universität Hannover. Universität Hannover, 1994.
- [ST92] J.C. Simo und N. Tarnow. The discrete energy-momentum method. Conserving algorithms for nonlinear elastodynamics. *Zeitschrift für angewandte Mathematik und Physik (ZAMP)*, 43:757–793, 1992.
- [ST94] J.C. Simo und N. Tarnow. A new energy and momentum conserving algorithm for the non-linear dynamics of shells. *International Journal for Numerical Methods in Engineering*, 37:2527–2549, 1994.
- [ST95] J.C. Simo und N. Tarnow. Non-linear dynamics of three-dimensional rods: Exact energy and momentum conserving algorithms. *International Journal for Numerical Methods in Engineering*, 38:1431–1473, 1995.
- [Ste71] K.V. Steinbrugge. Internet: <http://www.eerc.berkeley.edu> (Steinbrugge Collection, Earthquake Engineering Research Center, University of California, Berkeley), 1971.
- [Sti93] S. Stille. Lastbalancierung in verteilten Systemen. Diplomarbeit, Institut für Betriebssysteme und Rechnerverbund der Technischen Universität Braunschweig, 1993.
- [Tem95] C. Temme. *Numerische Behandlung der Interaktionseffekte von erdbebenerregten, unverankerten Flüssigkeitsbehältern*. Dissertation, Lehrstuhl für Statik, Technische Universität München, 1995.
- [Thi95] U Thiemann. Workstations im Cluster. *UNIX open*, 82–86, Juni 1995.
- [VV90] T. Vu Van. *Zur Behandlung von Stoß- Kontaktproblemen mit Reibung unter Verwendung der Finite-Element-Methode*. Dissertation, Fachbereich für Bauingenieur- und Vermessungswesen der Universität Hannover, 1990.
- [Wau99] H. Waubke. *Moment-Closure Technik zur Abschätzung der elasto-plastischen Reaktion von Stockwerksrahmen auf zufällige Belastungen aus Windereignissen*. Habilitationsschrift. Fakultät für Bauingenieur- und Vermessungswesen der Technischen Universität München, 1999.
- [WB95] B. Weber und H. Bachmann. Efficient Analysis of Dam-Reservoir Interaction in the Time Domain. In *Proceedings of the 10th European Conference on Earthquake Engineering*. A.A. Balkema, Rotterdam, 1995.
- [WBD96] K. Wechsler, M. Breuer und F. Durst. A parallel Multigrid Method for the prediction of incompressible flows on workstation clusters. In A. Bode, J. Dongarra, T. Ludwig und V. Sunderam, Hrsg., *Parallel Virtual Machine – EuroPVM’96. Proceedings of the Third European PVM Conference*, Lecture Notes in Computer Science, 53–58. Springer Verlag, 1996.

- [WBZ81] W.L. Wood, M. Bossak und O.C. Zienkiewicz. An alpha modification of Newmark's method. *International Journal for Numerical Methods in Engineering*, 15:1562–1566, 1981.
- [WCO85] W. Wunderlich, H. Cramer und H. Obrecht. Application of ring elements in the nonlinear analysis of shells of revolution under nonaxisymmetric loading. *Computer Methods in Applied Mechanics and Engineering*, 51:259–275, 1985.
- [WCST95] W. Wunderlich, H. Cramer, B. Schäpertöns und C. Temme. Dynamisches Tragverhalten flüssigkeitsgefüllter Behälter im Zusammenwirken mit der Gründung. 2. Bericht zum DFG Forschungsvorhaben Wu 67/21-2, 1995.
- [WEL99] N.-E. Wiberg, T. Ekevid und X.D. Li. Wave propagation in solids by adaptive DG-FE-procedure. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM'99 (CD-ROM)*, TU München, 1999.
- [WGK99] T. Wallmersperger, B.A. Grohmann und B. Kröplin. Time-Discontinuous Stabilized Space-Time Finite Elements for PDEs of First- and Second-Order in Time. In W. Wunderlich, Hrsg., *Proceedings of the European Conference on Computational Mechanics – ECCM'99 (CD-ROM)*, TU München, 1999.
- [WGS85] W. Wunderlich, W. Goebel und H. Springer. Anwendungen von Schrittweitensteuerungen bei nichtlinearen dynamischen Berechnungen von Rotationsschalen. In *Baudynamik. Kurzfassungen der Referate des SFB 151/FSP 30 Seminars*, 55–57. Universität Innsbruck, Mai 1985.
- [WH81] L.C. Wellford und S.M. Hamdan. An analysis of the stability and convergence properties of a Crank-Nicholson algorithm for nonlinear elasto-dynamics. In W. Wunderlich, E. Stein und K.-J. Bathe, Hrsg., *Nonlinear Finite Element Analysis in Structural Mechanics; Proceedings of the Europe-U.S. Workshop Ruhr-Universität Bochum, 28.-31. Juli 1980*, 502–518. Springer-Verlag, 1981.
- [Wib99] N.-E. Wiberg. Wave propagation in solids by adaptive DG-FE-procedure. Keynote-Lecture während der Tagung ECCM'99, TU München, September 1999.
- [WL94] N.E. Wiberg und X.D. Li. A postprocessed error estimation and adaptive procedure for the semidiscrete Finite Element Method in dynamic analysis. *International Journal for Numerical Methods in Engineering*, 37:3585–3603, 1994.
- [WL96] N.E. Wiberg und X.D. Li. Structural dynamic analysis by a Time-Discontinuous Galerkin Finite Element Method. *International Journal for Numerical Methods in Engineering*, 39:2131–2152, 1996.
- [WL98] N.-E. Wiberg und X. Li. Adaptive Discontinuous Galerkin FE procedures for linear and nonlinear structural dynamics. In S. Idelson, E. Onate und E. Dvorkin, Hrsg., *Computational Mechanics – New trends and applications. Conference Proceedings of the Fourth World Congress on Computational Mechanics (CD-ROM)*, Barcelona, Spanien, 1998. CIMNE.

- [WL99] N.E. Wiberg und X.D. Li. Nonlinear Structural dynamic analysis by a Time–Discontinuous Galerkin Finite Element Method. In H.A. Mang und F.G. Rammerstorfer, Hrsg., *Discretisation Methods in Structural Mechanics II*, Jgg. 68 of *Solid Mechanics and its Applications*, 115–122. IUTAM/IACM Symposium 1997, Kluwer Academic Publishers, 1999.
- [WMS65] V.I. Weingarten, E.J. Morgan und P. Seide. Elastic stability of thin-walled cylindrical and conical shells under axial compression. *AIAA Journal*, 3:500–505, 1965.
- [WN93] J.D. Wörner und H.-W. Nordhues. Parallel computing for nonlinear dynamic analysis. In T. Moan et al., Hrsg., *Structural Dynamics. Proceedings of the second European Conference on Structural Dynamics (Eurodyn '93), Trondheim*, 635–640. A.A. Balkema, Rotterdam, 1993.
- [Wol88] J.P. Wolf. *Soil-structure-interaction analysis in time domain*. Civil Engineering and Engineering Mechanics. Prentice-Hall, Inc., 1988.
- [WOSL87] W. Wunderlich, H. Obrecht, H. Springer und Z. Lu. A semi-analytical approach to the nonlinear analysis of shells of revolution. In A.K. Noor, T. Belytschko und J.C. Simo, Hrsg., *Analytical and Computational Methods for Shells*, New York, 1987. ASME.
- [WR95] W. Wunderlich und W. Redanz. Die Methode der Finiten Elemente. In G. Mehlhorn, Hrsg., *Rechnerorientierte Baumechanik*, Der Ingenieurbau, 141–247. Ernst u. Sohn, 1995.
- [WR<sup>+</sup>98] W. Wunderlich, M. Rapolder et al. Dynamisches Tragverhalten flüssigkeitsgefüllter Behälter im Zusammenwirken mit der Gründung. Abschlußbericht zum DFG Forschungsvorhaben Wu 67/21-3, 1998.
- [WR99] W. Wunderlich und M. Rapolder. Numerical treatment of the dynamic soil-structure-fluid interaction of unanchored liquid storage tanks - Parallel computing of a semi-analytical FE-model. In H.A. Mang und F.G. Rammerstorfer, Hrsg., *Discretisation Methods in Structural Mechanics II*, Jgg. 68 of *Solid Mechanics and its Applications*, 99–106. IUTAM/IACM Symposium, Wien 1997, Kluwer Academic Publishers, 1999.
- [Wri81] P. Wriggers. *Zur Berechnung von Stoß- und Kontaktproblemen mit Hilfe der Finite-Element-Methode*. Dissertation, Fachbereich für Bauingenieur- und Vermessungswesen der Universität Hannover, Januar 1981.
- [WSG89] W. Wunderlich, H. Springer und W. Goebel. Discretization and solution techniques for liquid filled shells of revolution under dynamic loading. In H.A. Mang und G. Kuhn, Hrsg., *Discretization Methods in Structural Mechanics*, 145–155, IUTAM-Symposium Wien, 1989. Springer-Verlag Heidelberg.
- [WSST91] W. Wunderlich, B. Schäpertöns, H. Springer und C. Temme. Dynamic soil-structure-interaction of nonlinear shells of revolution in the time domain. In



- Soil Dynamics and Earthquake Engineering V*, Computational Mechanics Publications, 455–465, Southampton, Boston, 1991. Elsevier-Verlag.
- [WST92] W. Wunderlich, B. Schäpertöns und C. Temme. Dynamisches Tragverhalten flüssigkeitsgefüllter Behälter im Zusammenwirken mit der Gründung. 1. Bericht zum DFG Forschungsvorhaben Wu 67/21-1, 1992.
- [WST94] W. Wunderlich, B. Schäpertöns und C. Temme. Dynamic stability of nonlinear shells of revolution under consideration of the fluid-structure-interaction. *International Journal for Numerical Methods in Engineering*, 37:2679–2697, 1994.
- [WT95] W. Wunderlich und C. Temme. Soil-fluid-structure interaction of uplifting tanks. In N.E. Wiberg, Hrsg., *Advances in finite element technology*, 189–211, Barcelona, 1995. CIMNE.
- [Wun67] W. Wunderlich. Zur Berechnung von Rotationsschalen mit Übertragungsmatrizen. *Ingenieur-Archiv*, 36(4):262–279, 1967.
- [XW93] Y.M. Xie und W.L. Wood. On the accuracy of time-stepping schemes for dynamic problems with negative stiffness. *Communications in Numerical Methods in Engineering*, 9:131–137, 1993.
- [YKV99] M. Yazdchi, N. Khalili und S. Valliappan. Non-Linear seismic behaviour of concrete gravity dams using coupled Finite Element-Boundary Element Technique. *International Journal for Numerical Methods in Engineering*, 44:101–130, 1999.
- [YV96] A.H. Younan und A.S. Veletsos. Dynamic response of cylindrical tanks storing a viscoelastic material. In *Proceedings of the eleventh World Conference on Earthquake Engineering*. Elsevier Science Ltd, 1996.
- [Zie84] O.C. Zienkiewicz. *Methode der finiten Elemente*. Carl Hanser Verlag München, 1984.
- [ZWH84] O.C. Zienkiewicz, W.L. Wood und N.W. Hine. A unified set of single step algorithms (Part 1). *International Journal for Numerical Methods in Engineering*, 20:1529–1552, 1984.
- [ZX91] O.C. Zienkiewicz und Y.M. Xie. A simple error estimator and adaptive time-stepping procedure for dynamic analysis. *Earthquake engineering and structural dynamics*, 20:871–887, 1991.