

RUHR-UNIVERSITÄT BOCHUM

J. Chroscielewski

L.-P. Nolte

Strategien zur Lösung
nichtlinearer Probleme der
Strukturmechanik und ihre
modulare Aufbereitung im Konzept
MESY

Heft Nr. 48



Mitteilungen
aus dem
Institut für Mechanik

INSTITUT FÜR MECHANIK
RUHR-UNIVERSITÄT BOCHUM

Jacek Chroscielewski / Lutz-Peter Nolte

STRATEGIEN ZUR LÖSUNG NICHTLINEARER PROBLEME
DER STRUKTURMECHANIK UND IHRE MODULARE
AUFBEREITUNG IM KONZEPT MESY3

Mitteilungen aus dem Institut für Mechanik Nr. 48

Oktober 1985

Herausgeber:

Institut für Mechanik der Ruhr-Universität Bochum

Die vorliegende Arbeit entstand während eines mehrmonatigen Forschungsaufenthaltes des erstgenannten Autors am Lehrstuhl für Mechanik II der Ruhr-Universität Bochum im Jahre 1985.

Herrn Prof. Dr.-Ing. H. Stumpf danken wir für die Förderung und großzügige Unterstützung dieser Arbeit.

Prof. Dr.-Ing. K.-H. Schrader sei gedankt für die Überlassung der MESY3-Quelle als auch für zusätzliche Impulse und wertvolle Hinweise in wissenschaftlichen Fragestellungen.

© 1985 Dr inz. Jacek Chroscielewski
Politechnika Gdanska, ul. Majakowskiego 11/12.
80-952 Gdansk/Poland
Dr.-Ing. Lutz-Peter Nolte
Am Gebrannten 3. 4630 Bochum 1/FRG

Alle Rechte vorbehalten. Auch die fotomechanische Vervielfältigung des Werkes oder von Teilen daraus bedarf der vorherigen Zustimmung des Herausgebers oder der Autoren.

ZUSAMMENFASSUNG

In der vorliegenden Arbeit werden verschiedene Strategien zur Lösung der diskretisierten nichtlinearen Grundgleichungen allgemeiner statischer Probleme der Strukturmechanik vorgestellt. Die Beschreibung der dazu erforderlichen Steueralgorithmen, die modulare Aufbereitung zugehöriger FORTRAN-Routinen und ihre Implementierung in das Programmsystem MESY3 bilden den inhaltlichen Schwerpunkt. Am Ende stehen umfangreiche numerische Untersuchungen aus den Problemkreisen Balken, Platte und Schale unter Berücksichtigung geometrischer und physikalischer Nichtlinearitäten.

SUMMARY

In the present report various strategies for the solution of the discretized nonlinear equations of general static structural problems are presented. The necessary trace-algorithms, the modular programming of corresponding FORTRAN-routines and their implementation into the system MESY3 are described in detail. Finally various illustrating numerical examples are analysed for bar, plate and shell problems. Geometrical and physical nonlinearities are taken into account.

INHALTSVERZEICHNIS

	<u>Seite</u>
1. EINFÜHRENDE BETRACHTUNGEN	1
2. SYSTEM - DIMENSIONIERUNGSROUTINEN MIT SICHERUNGSOPTIONEN	7
3. ELEMENT - DIMENSIONIERUNGSROUTINEN	9
4. STEUERROUTINEN ZUR LÖSUNG NICHTLINEARER PROBLEME	11
5. KONVERGENZKRITERIEN	26
6. RESTART- UND ARCHIVIERUNGSROUTINEN	30
7. VERWALTUNGSFUNKTIONEN	32
8. FUNKTIONEN ZUR BESTIMMUNG CHARAKTERISTISCHER SYSTEMGRÖSSEN	33
9. PSEUDO - VOLLAUTOMATISCHE UND INTERAKTIVE STEUERUNG	35
10. ANWENDUNGSBEISPIELE	40
Anhang A: FELDER DER GLOBALEN DATENBASIS	85
Anhang B: REFERENZLISTE DER COMMONVARIABLEN	86
Anhang C: ALPHABETISCHES ROUTINEN - LISTING	93
LITERATUR	94

1. EINFÜHRENDE BETRACHTUNGEN

1.1 EINLEITUNG

Eine Vielzahl von Arbeiten in der Literatur befaßt sich mit der Herleitung und Anwendung effizienter Verfahren zur Analyse des sogenannten Vorbeulbereichs strukturmechanischer Probleme. Die Berechnung überkritischer Zustände kann jedoch in verschiedenen Fällen von besonderem Interesse sein, so z.B. für die Bestimmung der Imperfektionsempfindlichkeit gewisser Strukturen oder die Vorhersage möglicher dynamischer Deformationsmuster. Erst in neueren Untersuchungen finden sich daher entsprechende Verfahren zur Analyse des Beul- und Nachbeulbereichs.

Unter Beschränkung auf inkrementelle und iterative Methoden sowie Kombinationen aus beiden werden in der vorliegenden Arbeit einige der gebräuchlichsten Strategien zur Lösung der diskretisierten nicht-linearen Grundgleichungen allgemeiner statischer Probleme der Strukturmechanik vorgestellt. Dazu gehören die reine Laststeuerung mit der Möglichkeit, die Gleichgewichtsiteration unter Verwendung des aktuellen Steifigkeitsparameters [1] in der Umgebung relativer Extrema der Last-Verschiebungskurve zu unterdrücken, die reine Verschiebungssteuerung [2-10] und die konstante Bogenlängenmethode [9-17] sowie beliebige Kombinationen aus diesen Verfahren. Zusätzlich werden zwei effiziente Modifikationen der konstanten Bogenlängenmethode durch selektive Vorwahl einzelner Knotengrößen und die Verwendung einer "updated" Orthogonalitätsbedingung in Verbindung mit dem Modifizierten Newton-Raphson- und dem Gauß'schen Eliminationsverfahren behandelt.

Die in dieser Arbeit vorgestellten Strategien beschränken sich auf die Analyse der primären, sekundären.... Gleichgewichtspfade, schließen jedoch den Übergang vom einen zum anderen Pfad nicht ein. Diese sogenannten Verzweigungen lassen sich z.B. durch Aufbringen einer Last- oder Verschiebungsimperfektion oder durch Veränderung des Verformungszustandes im Verzweigungspunkt mit Hilfe des zum kleinsten Eigenwert gehörenden Eigenvektors mit anschließender Gleichgewichtsiteration erreichen [18].

Unter dem Begriff MESY (Mechanik der Systeme) ist sowohl ein allgemeines Konzept zur Behandlung strukturmechanischer Aufgabenstellungen als auch der Gruppenname für eine Reihe modularer Programmsysteme zu verstehen. Dieses Konzept und die zugehörigen Programmimplementierungen

in ALGOL und FORTRAN auf verschiedenen Rechnersystemen wurden ganzheitlich seit 1966 von Schrader et al. [19-22] entwickelt.

Als Philosophie steht dahinter, ein Problem durch Aufrufe von Subroutinen des Programmsystems (MESY-Anweisungen) in Basisoperationen der Theorie problemorientiert zu formulieren. Der Benutzer erstellt dementsprechend ein sogenanntes Individualprogramm, worin er den Bearbeitungsvorgang durch MESY-Anweisungen steuert. Dabei besteht für ihn die Möglichkeit, entweder fertigen "Bauanleitungen" zu folgen, oder das System auf verschiedenste Weise benutzerseitig zu modifizieren und zu erweitern.

Im Hinblick auf die Implementierung oben genannter alternativer Lösungsverfahren wurde eine Aufbereitung der speziellen FORTRAN-Version MESY3, implementiert auf dem Bochumer Skalarrechner CDC-Cyber 855, notwendig. Zur Einarbeitung in ihre allgemeine Struktur und Anwendung sei auf [19] verwiesen. Für die Bearbeitung rechenzeitintensiver Probleme wurde zusätzlich eine auf dem Bochumer Vektorrechner CDC-Cyber 205 lauffähige Version erstellt.

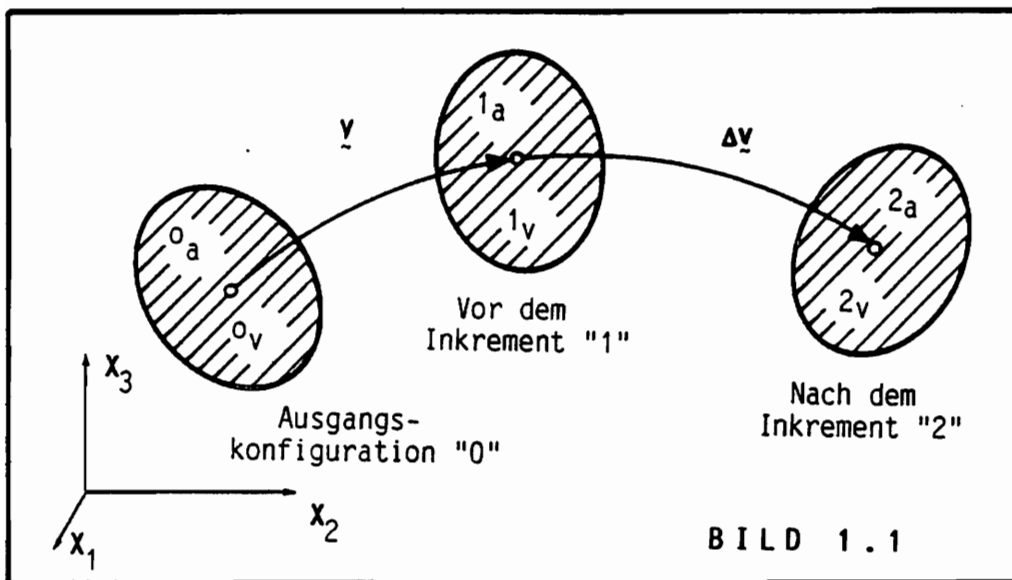
Die Beschreibung der Steueralgorithmen und zugehörigen FORTRAN-Routinen (Parameter, Wirkung, Anwendung, Syntax) bilden daher den inhaltlichen Schwerpunkt dieser Arbeit. Darüberhinaus werden einige speichertechnische Modifikationen zur deutlichen Reduzierung des Gesamtspeicherbedarfs vorgestellt. Auf weitere Modifikationen und Verbesserungen wesentlicher Routinen (Gleichungslöser, etc.) soll hier, da benutzerirrelevant, nicht eingegangen werden. Die vorgenommenen Änderungen schränken die ursprüngliche Flexibilität des als Berechnungsbaukasten konzipierten "offenen Programmsystems" MESY3 nicht ein. Daraus ergibt sich im Hinblick auf rein wissenschaftliches Programmieren ein deutlicher Vorteil gegenüber Programmen und Programmsystemen, die nach formatierten Eingabedaten für vorgegebene Problemkreise nach mehr oder weniger festem Schema ablaufen.

Abschließend wird die Leistungsfähigkeit der vorgestellten Algorithmen anhand einiger hochgradig geometrisch und physikalisch nichtlinearer Balken- und Schalenstrukturen unter Wirkung sogenannter "Totlasten" und Drucklasten überprüft. Die zugehörigen Problemstellungen, Individualprogramme, Ergebnislisten und Diagramme werden ausführlich mit Bezug zum erläuternden Text beschrieben.

1.2 GRUNDSÄTZLICHE BEZEICHNUNGEN

Zur Lösung der im Rahmen dieser Arbeit betrachteten nichtlinearen Strukturprobleme wird eine Aufbereitung der zugehörigen diskretisierten Grundgleichungen im Sinne einer inkrementellen Formulierung durchgeführt. Dies erlaubt mit Aufspaltung der Zustandsgrößen in bekannte und inkrementelle Anteile die Lösung des Gleichungssystems auf die Bestimmung der unbekanntenen Inkremente zurückzuführen.

Der inkrementelle Bewegungsvorgang eines Körpers im Raum läßt sich gemäß Bild 1.1 durch drei Zustände definieren, wobei neben dem dargestellten raumfesten kartesischen auch lokale (natürliche) und krummlinige Koordinaten oder eine konvektive Formulierung gewählt werden können.



Die Mehrzahl der Ausdrücke werden als Matrizengleichungen formuliert. Vektoren und Matrizen sind durch Fettschrift und eine Unterschläge besonders gekennzeichnet. Ein oberer linker Index steht für den "Wirkungs"zustand, der untere linke Index für den Bezugszustand, z.B.

$$\begin{matrix} m_F & m_F & \dots & m & \text{"Wirkungszustand"} \\ n_i & n_i & \dots & n & \text{Bezugszustand} \end{matrix} ,$$

womit man für die spezielle Wahl $n=0$ die totale Lagrangesche und mit $n=1$ eine mitgehende Formulierung erhält. Ein unterer rechter Index entspricht dem Index für den zugehörigen Systemfreiheitsgrad.

Inkrementelle Größen werden mit einem Δ gekennzeichnet

$$\Delta v_i , \Delta \underline{v} , \Delta \underline{R} , \Delta \lambda , \dots .$$

Zur Erfassung iterativer Prozeßgrößen innerhalb eines Inkrementzyklusses 1-2 (bei inkrementell-iterativen Verfahren [9,10] zu berücksichtigen) führen wir einen hochgestellten Index in Klammern ein

$$\Delta \underline{v}_i^{(k)}, \Delta \underline{v}^{(k)}, \Delta \underline{R}_i^{(k)}, \Delta \underline{R}^{(k)}, \Delta \lambda^{(k)}, \dots$$

Die den Übergang vom Zustand 1 zum Zustand 2 beschreibenden Inkremente erhält man dann als Summe über alle Iterationszuwächse

$$\Delta \underline{v} = \sum_{k=0}^M \Delta \underline{v}^{(k)}, \Delta \underline{R} = \sum_{k=0}^M \Delta \underline{R}^{(k)}, \Delta \lambda = \sum_{k=0}^M \Delta \lambda^{(k)}, \dots \quad (1.2.1)$$

wobei mit $k=0$ nach der Inkrementierung der erste Berechnungsschritt und mit $k=1, \dots, M$ alle weiteren Gleichgewichts-Iterationsschritte in einem Inkrementzyklus bezeichnet werden.

Mit der vorstehend erläuterten Notation ergeben sich die bekannten inkrementellen Gleichgewichtsbedingungen in linearisierter Form zu

$$\underline{K}_n^{(i)} \Delta \underline{v}^{(k)} = \underline{R}_n^{(k)} + \underline{R}_{n-alt}^{(k)} - \underline{F}_n^{(k-1)} \quad (1.2.2)$$

In (1.2.2) sind die folgenden Abkürzungen eingeführt worden:

$\underline{K}_n^{(i)}$ - Steifigkeitsmatrix in der Lage (i) bezogen auf den Zustand n, wobei $i=k-1$ die klassische Tangentialsteifigkeitsmatrix liefert,

$\Delta \underline{v}^{(k)}$ - Verschiebungszuwachs im (k)-ten Iterationszyklus,

$\underline{R}_n^{(k)}$ - Lastvektor in der Lage (k) bei allgemeiner Proportionalitätsbelastung bezogen auf den Zustand n,

$$\underline{R}_n^{(k)} = \lambda^{(k)} \underline{R}_{n-ref}^{(k-1)} \quad (1.2.3)$$

$\lambda^{(k)}$ - Proportionalitätsfaktor für den Lastvektor im Iterationszyklus (k),

$$\lambda^{(k)} = 1 + \sum_{j=0}^k \Delta \lambda^{(j)} \quad (1.2.4)$$

(Bei Laststeuerung ist für $j > 0$ $\Delta \lambda^{(j)}$ identisch null),

$\Delta \lambda^{(j)}$ - Zuwachs des Proportionalitätsfaktors pro Iterationszyklus,

$\overset{(k)}{n}\underset{-}{R}_{\text{ref}}$ - Referenzlastvektor im Iterationszyklus (k) bezogen auf den Zustand n,

$\overset{(k)}{n}\underset{-}{R}_{\text{alt}}$ - "Alter" System-Lastvektor eines Gleichgewichtszustands bezogen auf den Zustand n. Dieser Vektor wird verwendet, wenn zwischen "altem" und "neuem" Lastvektor keine Proportionalitätsrelation besteht; dies ermöglicht die Berücksichtigung pseudo-mehrparametriger Lastsysteme,

$\overset{(k)}{n}\underset{-}{F}$ - Vektor der verallgemeinerten inneren Knotenkräfte in der Lage (k) bezogen auf den Zustand n.

Zusätzlich werden für den ersten Berechnungsschritt im Inkrementzyklus (k=0) die folgenden vereinfachten Abkürzungen eingeführt

$$\overset{(-1)}{n}\underset{-}{K} \equiv \overset{1}{n}\underset{-}{K} \quad , \quad \overset{(-1)}{n}\underset{-}{F} \equiv \overset{1}{n}\underset{-}{F} \quad , \quad \overset{(-1)}{n}\underset{-}{R}_{\text{ref}} \equiv \overset{1}{n}\underset{-}{R}_{\text{ref}} \quad ,$$

und unter Verwendung der Beziehung (1.2.2)

$$\overset{(k)}{n}\underset{-}{R} \equiv \overset{2}{n}\underset{-}{R} = \overset{2}{\lambda} \overset{1}{n}\underset{-}{R}_{\text{ref}} \quad .$$

Die inkrementellen Gleichgewichtsbedingungen (1.2.2) lassen sich nun in die Form

$$\overset{(i)}{n}\underset{-}{K} \Delta \underset{-}{V} = \Delta \lambda \overset{(k)}{n}\underset{-}{R}_{\text{ref}} + \overset{(k-1)}{n}\underset{-}{J} \quad , \quad (1.2.5)$$

überführen, mit

$$\Delta \lambda \overset{(k)}{n}\underset{-}{R}_{\text{ref}} = \overset{(k)}{\Delta R} = \overset{(k)}{n}\underset{-}{R} - \overset{(k-1)}{n}\underset{-}{R} = (\lambda - \overset{(k-1)}{\lambda}) \overset{(k-1)}{n}\underset{-}{R}_{\text{ref}} \quad , \quad (1.2.6)$$

und

$\overset{(k)}{n}\underset{-}{J}$ - Vektor der Ungleichgewichtskräfte im Iterationszyklus (k) bezogen auf den Zustand n,

$$\overset{(k)}{n}\underset{-}{J} = \overset{(k)}{n}\underset{-}{R} + \overset{(k)}{n}\underset{-}{R}_{\text{alt}} - \overset{(k)}{n}\underset{-}{F} \quad . \quad (1.2.7)$$

Zur Lösung der nichtlinearen Strukturprobleme sind aus dem Raum der zulässigen generalisierten Last- und Verschiebungsgrößen solche $\overset{(k)}{\lambda}$, $\overset{(k)}{v}$ zu bestimmen, welche die diskretisierten nichtlinearen Gleichgewichtsbedingungen

$$\overset{(k)}{n}\underset{-}{J} = \underline{0} \quad , \quad (1.2.8)$$

erfüllen.

Die linearisierten Gleichgewichtsbedingungen in der Form (1.2.5) bilden

die Grundlage für ein lastgesteuertes Lösungsverfahren. Man beachte jedoch, daß im allgemeinen Fall Inkrementierung oder/und Iteration nicht nur im Raum der Verschiebungen sondern auch im Raum der Lasten stattfindet. d.h. der Proportionalitätsfaktor für den Lastvektor sich von einem Berechnungsschritt zum anderen ändern kann. Dies bildet den Ausgangspunkt zur Formulierung alternativer Lösungsstrategien (s. Kap. 4).

1.3 HINWEISE ZUR PROGRAMMBESCHREIBUNG

Die Beschreibung der verschiedenen Programmimplementierungen wird in den Kapiteln 2-9 nach übergeordneten Gesichtspunkten gegliedert, ob es sich z.B. um Eingriffe auf Element- oder Systemebene und wesentliche Algorithmen zur Steuerung und Konvergenzabfrage oder um Programmierhilfen, Verwaltungs-, Restart- und Archivierungsfunktionen handelt. Jedes Kapitel beginnt zunächst mit einem Überblick über alle dort vorgestellten Routinen sowie einem Hinweis zur Mnemotechnik. Es folgen die Einzelbeschreibungen mit Erläuterungen zu möglichen Parametern, zur Wirkung, Syntax und Anwendung. Zusätzlich wird in den Kapiteln 4 und 5 eine ausführliche Darstellung der wesentlichen Algorithmen alternativer Lösungsstrategien und verschiedener Konvergenzkriterien gegeben. Um dem Anwender die Arbeit mit dieser Programmbeschreibung zu erleichtern, enthält der Anhang ein alphabetisches Routinenverzeichnis mit Bezug zum erläuternden Text sowie Übersichten über die Felder der globalen Datenbasis und Referenzlisten der Commonvariablen.

2. SYSTEM - DIMENSIONIERUNGSROUTINEN MIT SICHERUNGSOPTIONEN

Übersicht:

LGESCHT	-	Lastgeschichte
LAAEPA	-	Laständerungsparameter
SICHERN	-	Sicherungsroutine

2.1 CALL LGESCHT (NGST)

Parameter	Typ	Bedeutung
NGST	integer	Größe des Speicherplatzes im Blank Common, Bereich CC.

LGESCHT reserviert Speicherplätze im Blank Common, Bereich CC, für alle Elementgrößen (z.B. zur Erfassung einer Verformungsgeschichte), die zur weiteren Bearbeitung benötigt werden. Der Inhalt dieses Speicherbereiches wird automatisch archiviert, wenn zusätzlich die Routine ARCHPA aufgerufen wird (s. Kap. 6). Optional in SKT1.

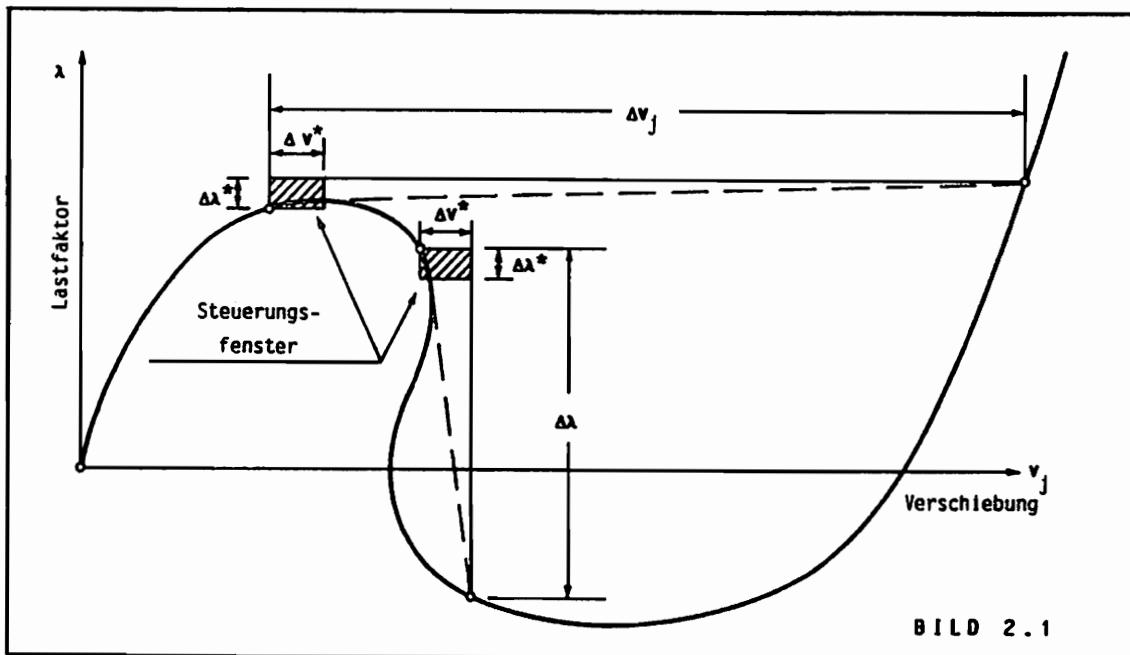
2.2 CALL LAAEPA /ohne Parameter/

LAAEPA reserviert Speicherplätze im Blank Common, Bereich CC, für den "alten" System-Lastvektor ($n_{\text{-alt}} \mathbf{R}_{\text{-alt}}$). Der Lastkoeffizient im aktuellen Zustand (${}^1\lambda$) wird Null gesetzt. Diese Routine ist anzuwenden, wenn zwischen "altem" und "neuem" Lastvektor keine Proportionalitätsfaktoren auftreten, wie z.B. bei Problemen mit mehrparametrischen Lasten oder vorgespannten Systemen, die im weiteren mit einer Nutzlast beaufschlagt werden. Optional in SKT1,3,7, jedoch beim ersten Aufruf in SKT1. !!!: Diese Analyse über den "alten" Lastvektor kann nur bei solchen Belastungen durchgeführt werden, die keinen Beitrag zum Restkraftvektor und zur Steifigkeitsmatrix liefern. Für die sogenannten "Totlasten" ($n_{\text{-alt}} \mathbf{R}_{\text{-alt}}, n_{\text{-ref}}^{(k)} \mathbf{R}_{\text{-ref}} \equiv \text{const.}$) ist dies meistens erfüllt.

2.3 CALL SICHERN (GP, GV)

Parameter	Typ	Bedeutung
GP	real	Maßstabsfaktor für den Laststeuerungsfaktor ($\Delta\lambda^*$),
GV	real	Maßstabsfaktor für die Verschiebungssteuerungsvariable (Δv_j^*).

SICHERN ermittelt die maximalen Zuwächse für das Verschiebungs- und Lastinkrement, $\Delta\lambda < \Delta\lambda^* \cdot GP$, $\Delta v_j < \Delta v_j^* \cdot GV$, (s. Bild 2.1). Es werden für alle System- und Elementgrößen, die zur weiteren Berechnung notwendig sind, Speicherplätze im Blank Common, Bereich CC, reserviert, wenn die Routine ARCHPA (s. Kap. 6) aufgerufen wurde oder wenn mit den Routinen des Kapitels 9 gesteuert wird.



Ist das Kriterium für $\Delta\lambda$ und Δv_j nicht erfüllt, werden in diesen Fällen die letzten System- und Elementgrößen in den vorbelegten Speicher gerettet und die Berechnung beendet, wenn nicht mit den Routinen des Kapitels 9 gesteuert wird. Dies eröffnet mit Hilfe eines Restarts oder bei interaktiver Steuerung die Möglichkeit, die Berechnung mit Änderung aller Steuerparameter fortzuführen. Optional in SKT1,3,7, jedoch beim ersten Aufruf in SKT1.

3. ELEMENT - DIMENSIONIERUNGSROUTINEN

Übersicht:

EMLAST	-	Elementlast
ELORGP	-	Elementorganisation für Elementlasten
NOTHEM	-	Nothilfematrix
SYMELM	-	Symmetrischer Teil der Elementmatrizen

3.1 CALL EMLAST /ohne Parameter/

EMLAST reserviert Speicherplatz im Blank Common, Bereich CC, für Elementlasten. Optional in SKT1, in Verbindung mit ELORGP oder mit IRPEAD (s. Kap. 7) obligat in SKT1.

3.2 CALL ELORGP (PED,LPE)

Parameter	Typ	Bedeutung
PED	real	Knotenlastvektor in Elementkoordinaten,
LPE	integer	Größe des Vektors PED mit $LPE > 0$.

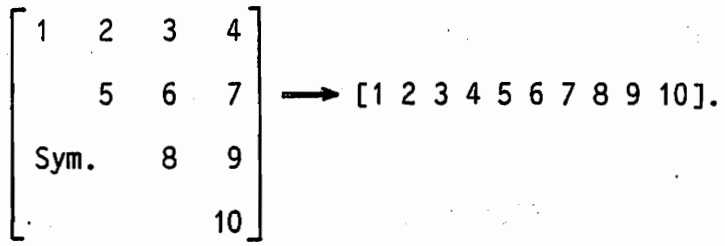
ELORGP entspricht in ihrer Wirkung der Routine ELORG (s.[19]), jedoch für die Elementlasten (EMLAST). Optional in SKT5.

3.3 CALL NOTHEM /ohne Parameter/

NOTHEM reduziert den Speicherplatz für Elementmatrizen im Blank Common, Bereich CC, falls keine Hilfsmatrix (HE(j,k) s.[19]) benötigt wird. Optional in SKT1.

3.4 CALL SYMELM /ohne Parameter/

SYMELM reduziert im Vergleich zur Standard MESY3-Version (s.[19]) den Speicherplatz der Elementmatrizen durch zeilenweises Abspeichern nur des symmetrischen Anteils, z.B.



Optional in SKT1.

4. STEUERROUTINEN ZUR LÖSUNG NICHTLINEARER PROBLEME

Übersicht:

STEUPA	-	Steuerungsparameter
SNEWRA	-	Standard Newton-Raphson Verfahren
MNEWRA	-	Modifiziertes Newton-Raphson Verfahren
EXIST	-	Existenz der Steifigkeitsmatrix
KAWERB	-	Keine Änderung der wesentlichen Randbedingungen
AKSTPR	-	Aktueller Steifigkeitsparameter
KOBSEL	-	Konstante Bogenlängenmethode mit Selektiver Vorwahl der Knotenwerte
SELWER	-	selektierte Knotenwerte

Eine Vielzahl von Arbeiten des mathematisch und ingenieurmäßig orientierten Schriftentums behandeln Lösungsverfahren für nichtlineare algebraische Gleichungssysteme der Form (1.2.8). Im Rahmen dieser Arbeit sollen jedoch nur inkrementelle und iterative Methoden sowie Kombinationen aus beiden berücksichtigt werden, welche die weiteste Verbreitung gefunden haben. Im einzelnen sind dies Euler-Verfahren

- a) als klassische Verfahren ohne Iteration mit und ohne selbstkorrigierender Relaxation I. Stufe,
- b) als kombinierte Verfahren mit Standard oder Modifizierter Newton-Raphson Iteration,

mit jeweils drei Möglichkeiten der Berechnungssteuerung

- a) Laststeuerung,
- b) Verschiebungssteuerung,
- c) konstante Bogenlängensteuerung.

4.1 LASTSTEUERUNGSMETHODE

Bei einer lastgesteuerten Lösung des linearisierten Gleichungssystems (1.2.5) wird die Lastintensität λ als Faktor eines Referenzlastvektors \mathbf{R}_{n-ref}^1 stufenweise erhöht. Zusätzlich können Gleichgewichtsiterationen

durchgeführt werden ($\overset{(k)}{R}_{n\text{-ref}}$ für z_λ), woraus sich Verbesserungen für den zugehörigen Verformungsvektor $\overset{(k)}{v}_n$ ergeben, solange bis der Gleichgewichtszustand erreicht, d.h. (1.2.8) im Rahmen einer vorgegebenen Schranke erfüllt ist. Diese Methode kann als das Standard-Berechnungsverfahren bezeichnet werden. Wir wollen uns daher in diesem Abschnitt auf die Beschreibung einer Modifikation des lastgesteuerten klassischen Eulerverfahrens durch selbstkorrigierende Relaxation I. Stufe beschränken.

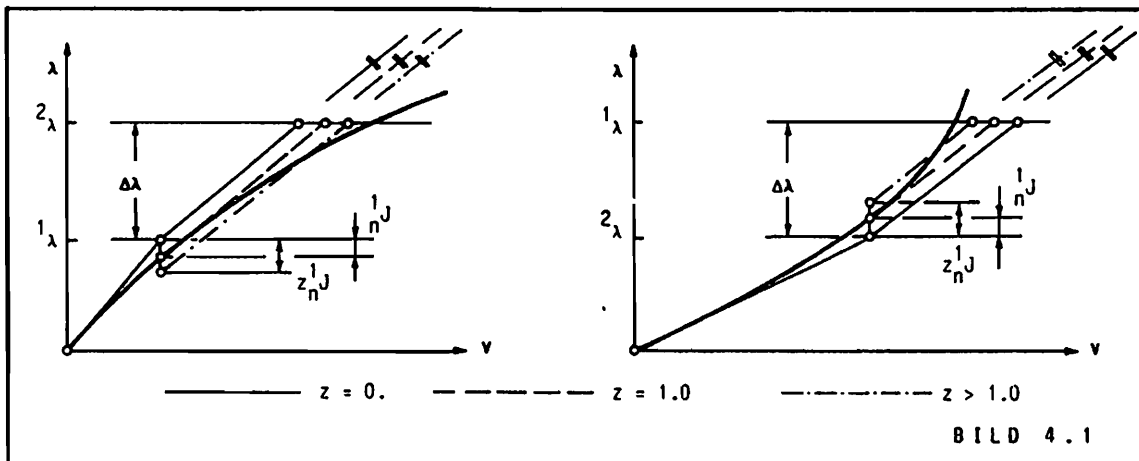
Falls keine Gleichgewichtssiterationen durchgeführt werden, gilt

$$\overset{1}{n}\underline{J} \neq \underline{0} . \quad (4.1.1)$$

Um nun (1.2.8) zumindest näherungsweise zu erfüllen, benutzt man bei der selbstkorrigierenden Relaxationsmethode I. Stufe einen Korrekturvektor nach Stricklin, Haisler und Reisemann [23]. Damit läßt sich Gleichung (1.2.5) in die folgende Form überführen

$$\overset{1}{n}\underline{K} \Delta \underline{v} = \Delta \lambda \overset{1}{n}\underline{R}_{\text{ref}} + z \overset{1}{n}\underline{J} , \quad (4.1.2)$$

mit z als selbstkorrigierenden Faktor I. Stufe. Als Ergebnis umfangreicher numerischer Testrechnungen wird in [24] der Erfahrungswert $z = 1.2$ vorgeschlagen. Zum besseren Verständnis ist in Bild 4.1 eine einfache Interpretation des Faktors z dargestellt.



Der erläuterte lastgesteuerte Berechnungsalgorithmus mit einer Gleichgewichtssiteration nach Newton-Raphson kann keine lokalen Extremstellen des Last-Verformungsverlaufes überwinden. Wir werden uns daher in den Abschnitten 4.2-4 mit einigen alternativen Lösungsstrategien beschäftigen.

4.2 VERSCHIEBUNGSSTEUERUNGS-METHODE

Die Methode der Verschiebungssteuerung wurde erstmals von Argyris et al. [2] zunächst für lineare Probleme und Einzellasten eingeführt und später von Pian und Tong [3] in eine Zweischrittmethode für allgemeine Belastungen überführt. Eine physikalische Interpretation dieses Verfahrens gibt Zienkiewicz [4]. Erwähnenswert ist außerdem die Arbeit von Desai und Abel [5]. Butterwords [6] untersuchte Möglichkeiten der Steuerung für die Variablen $\Delta\lambda/\Delta v^* \rightarrow \infty$ sowie $\det |n\tilde{K}| \rightarrow 0$. Weitere Fortschritte erzielten Lock und Sabir [7] durch Berücksichtigung der Ungleichgewichtskräfte, sowie Haisler, Stricklin und Key [8], die mit einem selbstkorrigierenden Verfahren arbeiteten. Eine ausführliche Darstellung der Verschiebungsmethode findet sich in den Arbeiten von Ramm [9] und Waszczyszyn [10].

Bei der Verschiebungssteuerung schreibt man ein einzelnes, jedoch zunächst beliebiges, Inkrement einer Verschiebungskomponente Δv_j als unabhängige Steuervariable vor. Diese Wahl ermöglicht die Aufspaltung der Matrizengleichung (1.2.5) in zwei Systeme

$$\text{I. } \begin{bmatrix} \begin{matrix} (i) \\ n\tilde{K}_{AA} \end{matrix} & \begin{matrix} (i) \\ n\tilde{K}_{Aj} \end{matrix} \\ \hline \begin{matrix} (i) \\ n\tilde{K}_{jA} \end{matrix} & \begin{matrix} (i) \\ n\tilde{K}_{jj} \end{matrix} \end{bmatrix} \cdot \begin{bmatrix} (k) \\ \Delta v_{-A} \end{bmatrix} = \Delta\lambda \begin{bmatrix} (k-1) \\ n\tilde{R}_{ref,A} \end{bmatrix} + z \begin{bmatrix} (k-1) \\ n\tilde{J}_{-A} \end{bmatrix}, \quad (4.2.1)$$

$$\text{II. } \begin{bmatrix} \begin{matrix} (i) \\ n\tilde{K}_{AA} \end{matrix} & \begin{matrix} (i) \\ n\tilde{K}_{Aj} \end{matrix} \\ \hline \begin{matrix} (i) \\ n\tilde{K}_{jA} \end{matrix} & \begin{matrix} (i) \\ n\tilde{K}_{jj} \end{matrix} \end{bmatrix} \cdot \begin{bmatrix} (k) \\ \Delta v_j \end{bmatrix} = \Delta\lambda \begin{bmatrix} (k-1) \\ n\tilde{R}_{ref,j} \end{bmatrix} + z \begin{bmatrix} (k-1) \\ n\tilde{J}_j \end{bmatrix}, \quad (4.2.2)$$

mit $A = N-1$ und N der Anzahl der Systemfreiheitsgrade.

Für den ersten Berechnungsschritt oder eine rein inkrementelle Lösung ohne Gleichgewichtsiteration gilt

$$k=0 \left\{ \begin{array}{l} \begin{matrix} (0) \\ \Delta v_j \end{matrix} = \Delta v^* \\ \begin{matrix} (-1) \\ n\tilde{R}_{ref} \end{matrix} = \begin{matrix} 1 \\ n\tilde{R}_{ref} \end{matrix} \\ \begin{matrix} (-1) \\ n\tilde{J} \end{matrix} = \begin{matrix} 1 \\ n\tilde{J} \end{matrix}, \quad z = 0 \text{ oder } z > 1.0. \end{array} \right. \quad (4.2.3)$$

Gleichung (4.2.1) liefert in einem I. Schritt für verschiedene Systemgrößen

$$\begin{matrix} (i) \\ n\tilde{K}_{AA} \end{matrix} \begin{bmatrix} (0) \\ \Delta v_{-A} \end{bmatrix} - \begin{matrix} (i) \\ n\tilde{K}_{Aj} \end{matrix} \begin{bmatrix} (0) \\ \Delta v_{-A} \end{bmatrix} + \begin{matrix} (0) \\ n\tilde{R}_{ref,A} \end{matrix} - \begin{matrix} (0) \\ n\tilde{J}_{-A} \end{matrix} = \begin{bmatrix} (i) \\ -n\tilde{K}_{Aj} \end{bmatrix} \begin{bmatrix} 1 \\ n\tilde{R}_{ref,A} \end{bmatrix} + \begin{bmatrix} 1 \\ n\tilde{J}_{-A} \end{bmatrix} \quad (4.2.4)$$

die Verschiebung

$$\begin{matrix} (0) \\ \Delta v_{-A} \end{matrix} \begin{bmatrix} 1 \\ n\tilde{K}_{AA} \end{bmatrix} = \begin{matrix} (i) \\ -1 \end{matrix} \begin{bmatrix} 1 \\ n\tilde{K}_{AA} \end{bmatrix} \cdot \begin{matrix} (0) \\ \Delta v_{-A} \end{matrix} \quad (4.2.5)$$

In einem II. Schritt gemäß (4.2.2) lassen sich mit (4.2.5) aus

$${}^{(i)}K_{jA} [\Delta v_A^{(0)} ({}^{(i)}K_{Aj}) \vdots \Delta v_A^{(0)} ({}^{(1)}R_{ref,A}) \vdots \Delta v_A^{(0)} ({}^{(1)}J_J)] = [{}^{(i)}K_{jj} \vdots {}^{(1)}R_{ref,j} \vdots {}^{(1)}J_j] \quad (4.2.6)$$

die folgenden fiktiven Reaktionen berechnen

$${}^{(0)}Q_j ({}^{(1)}n^*) = {}^{(i)}K_{jA} \Delta v_A^{(0)} ({}^{(1)}n^*) - ({}^{(1)}n^*_j) \quad (4.2.7)$$

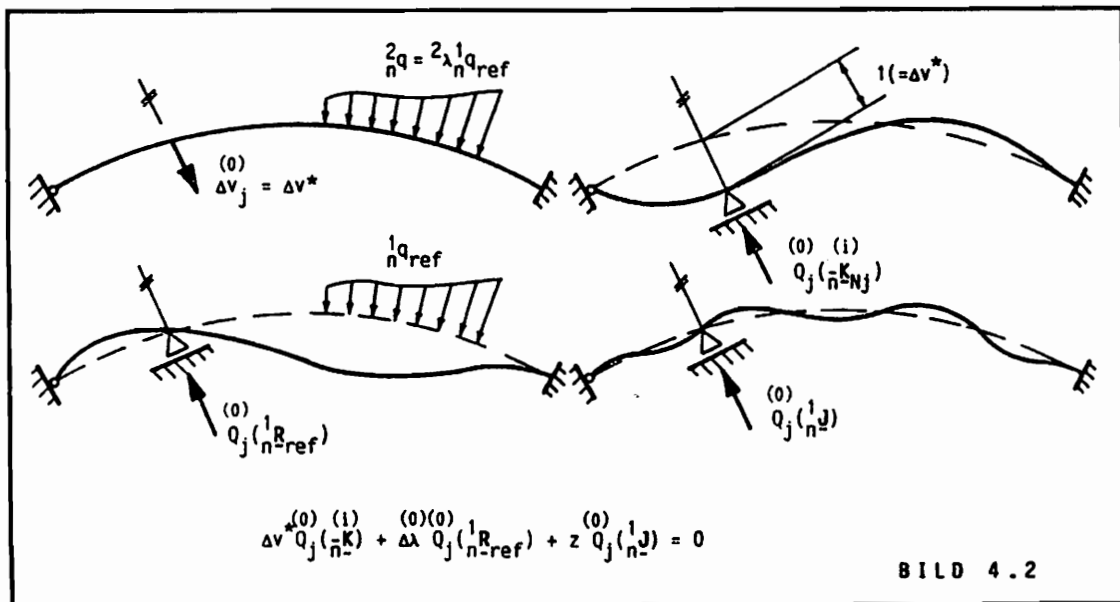
Durch Superposition erhalten wir schließlich die Gleichgewichtsbedingungen in Größen der fiktiven Reaktionen

$$\Delta v^* {}^{(0)}Q_j ({}^{(i)}K_{Nj}) + \Delta \lambda {}^{(0)}Q_j ({}^{(1)}R_{ref}) + z {}^{(0)}Q_j ({}^{(1)}J) = 0 \quad (4.2.8)$$

woraus sich der gesuchte Zuwachs proportional zum Lastvektor $\Delta \lambda^{(0)}$ ergibt

$$\Delta \lambda^{(0)} = - \frac{\Delta v^* {}^{(0)}Q_j ({}^{(i)}K_{Nj}) + z {}^{(0)}Q_j ({}^{(1)}J)}{{}^{(0)}Q_j ({}^{(1)}R_{ref})} \quad (4.2.9)$$

Die beschriebene Vorgehensweise kann graphisch gemäß Bild 4.2 interpretiert werden.



Für einen nachgeschalteten Iterationsprozeß findet man mit $k > 0$; $\Delta v_j^{(k)} = 0$, $z = 1.0$ analog zum ersten Berechnungsschritt ($k=0$) die Beziehung

$$\Delta \lambda^{(k)} Q_j ({}^{(k)}R_{ref}) + Q_j ({}^{(k-1)}J) = 0 \quad (4.2.10)$$

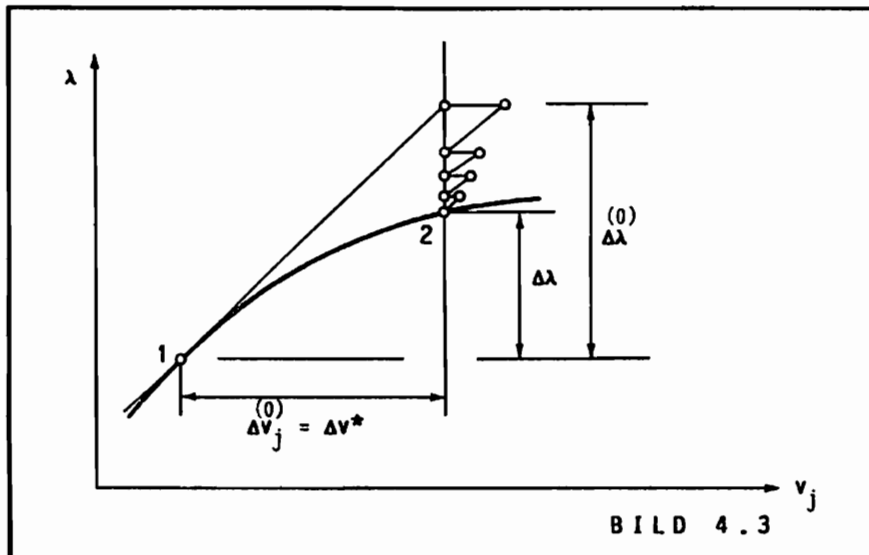
und den entsprechenden Zuwachs proportional zum Lastvektor $\Delta\lambda^{(k)}$

$$\Delta\lambda^{(k)} = - \frac{Q_j^{(k)} \left(\begin{matrix} (k) \\ n \end{matrix} \right)}{Q_j^{(k)} \left(\begin{matrix} (k-1) \\ n \end{matrix} \right)} \cdot \quad (4.2.11)$$

Nach $k=1, \dots, M$ Gleichgewichtssiterationen kann das gesuchte Inkrement $\Delta\lambda$ vom Zustand 1 zum Zustand 2 aus der Summe über alle Iterationszuwächse

$$\Delta\lambda = \sum_{k=0}^M \Delta\lambda^{(k)} \quad (4.2.12)$$

berechnet werden. Dieser inkrementell iterative Prozeß ist anschaulich im Bild 4.3 dargestellt.



4.3 KONSTANTE BOGENLÄNGENMETHODE

Dieses Verfahren wurde gleichzeitig von Riks [11] und Wempner [12] eingeführt und später in den Arbeiten [13-15] weiterentwickelt. Ausgehend von geometrischen Betrachtungen schlagen Wessels [16], Crisfield [17] und Ramm [9] eine Zweischrittlösung für diese Methode vor.

Die Grundidee für das Verfahren der konstanten Bogenlänge liegt darin, nicht mit einzelnen Verschiebungskomponenten oder dem Lastzuwachs zu steuern sondern die Zuwächse für Systemgrößen und Lastfaktoren als gleichwertige Unbekannte zu betrachten. Dann ist der Zuwachs der

Zustandsgrößen im k-ten Iterationszyklus und im aktuellen Inkrement definiert als

$$\Delta \underline{r}_i^{(k)T} = [\Delta \underline{v}_i^{(k)T}, \Delta \lambda] \quad \Delta r_i^{(k)} \quad i=1, N+1, \quad (4.3.1)$$

$$\Delta \underline{r}^{(k)T} = \sum_{k=0}^M [\Delta \underline{v}^{(k)T}, \Delta \lambda] \quad (4.3.2)$$

Zusätzlich wird eine konstante Bogenlänge Δs für das Inkrement aller unbekanntenen Größen angestrebt

$$\Delta s^2 = \Delta \underline{r}^{(k)T} \Delta \underline{r}^{(k)} \quad , \quad \Delta s = \text{konst.} \quad (4.3.3)$$

mit dem zugehörigen Einheitsvektor in Richtung von $\Delta \underline{r}^{(k)}$

$$\underline{t} = \frac{\Delta \underline{r}^{(k)}}{\|\Delta \underline{r}^{(k)}\|} \quad , \quad \underline{t}^T \Delta \underline{r}^{(k)} = \Delta s. \quad (4.3.4)$$

Unter Berücksichtigung von (4.3.3) gelangt man zu einer äquivalenten Darstellung der linearisierten Gleichgewichtsbeziehungen (1.2.5)

$$\left\{ \begin{array}{l} \begin{array}{l} (j) \quad (k) \quad (k) \quad (k-1) \\ n_K \Delta \underline{v} = \Delta \lambda \quad n_{\text{-ref}} \underline{R} + z \quad n_{\text{-}} \underline{J} \end{array} \\ \underline{t}^T \Delta \underline{r} = \Delta s \end{array} \right. \quad (4.3.5)$$

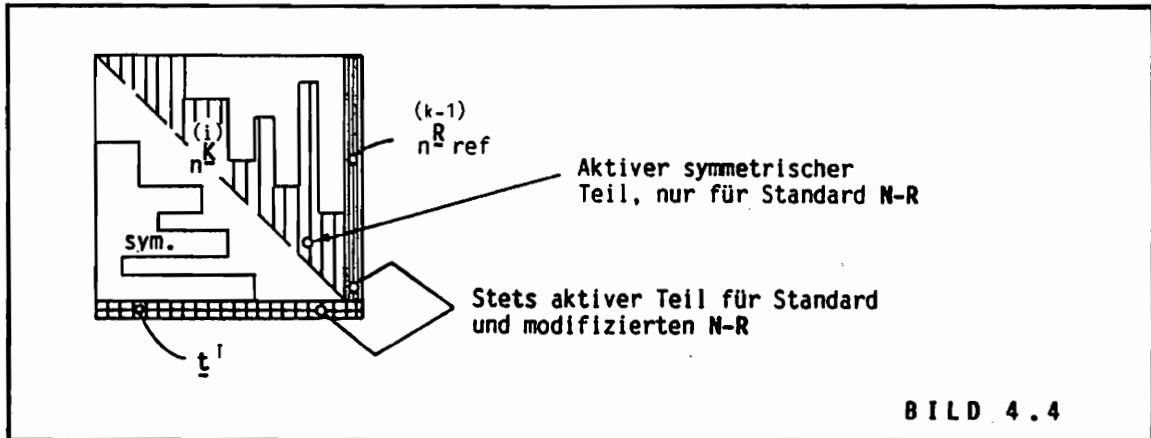
Aus der Bedingung (4.3.3) selbst folgt, daß für den Zuwachs der Zustandsgrößen (4.3.1) die Bedingung (4.3.3) genähert erfüllt ist, wenn

$$\Delta \underline{r}^{(k)T} \Delta \underline{r}^{(k)} = 0 \quad k > 0, \quad (4.3.6)$$

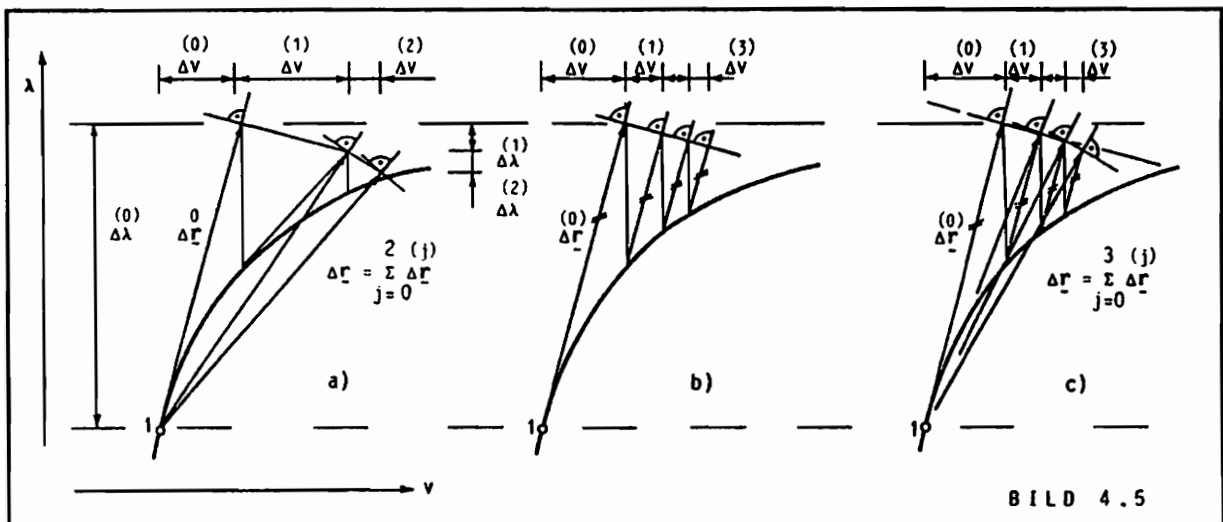
d.h. eine Orthogonalitätsbedingung zwischen dem Inkrement und dem Zuwachs des Inkrements besteht. Damit erhalten wir (4.3.5) in der modifizierten Form

$$\left[\begin{array}{c|c} \begin{array}{l} (j) \quad (k-1) \\ n_K \quad n_{\text{-ref}} \end{array} & \begin{array}{l} (k) \\ \Delta \underline{v} \end{array} \\ \hline \underline{t}^T \end{array} \right] \cdot \left[\begin{array}{l} (k) \\ \Delta \lambda \end{array} \right] = \left[\begin{array}{l} z \quad (k-1) \\ n_{\text{-}} \underline{J} \\ \Delta s \end{array} \right] \quad \begin{array}{l} k=0 \rightarrow z \geq 0 \\ k > 0 \rightarrow z = 1.0, \Delta s = 0. \end{array} \quad (4.3.7)$$

Dieses Gleichungssystem kann in eine charakteristische Form gemäß Bild 4.4 überführt werden [10]. In Verbindung mit dem Gauß'schen Eliminationsverfahren ermöglicht dies die Anwendung der Orthogonalitäts-



bedingung (4.3.6) zwischen dem im Schritt k aktualisierten Einheitsvektor $\underline{t}^{(k)}$ und dem gesuchten Zuwachs des Inkrementes $\Delta \underline{r}^{(k+1)}$ beim Modifizierten-Newton-Raphson Verfahren. Man beachte, daß in diesem Fall der Referenzlastvektor $\underline{n}_{ref}^{(k)}$ zu aktualisieren ist, wenn verformungsabhängige Lasten vorliegen. Eine graphische Darstellung der konstanten Bogenlängenmethode mit der Orthogonalitätsbedingung in typischer Form unter Verwendung des Standard- und des Modifizierten Newton-Raphson Verfahrens geben die Bilder 4.5 a,b. Bild 4.5 c zeigt schließlich die Anwendung der Orthogonalitätsbedingung für die aktuellen Größen in Verbindung mit dem Modifizierten Newton-Raphson Verfahren.



Ein Nachteil der oben beschriebenen konstanten Bogenlängenmethode in typischer Form besteht jedoch darin, daß die verschiedenen Komponenten des Systemvektors $\Delta \underline{v}$ im allgemeinen von ihrer Größenordnung nicht immer vergleichbar sind. So ergeben sich im Fall des in Kapitel 10

verwendeten doppelt gekrümmten Schalenelementes bei ungeschickter Wahl der konvektiven Gauss'schen Flächenkoordinaten Unterschiede um mehrere Größenordnungen.

Grundsätzlich gibt es zwei Möglichkeiten, die daraus resultierenden numerischen Instabilitäten zu beheben. Erstens, die Steuerung mit Hilfe einer bezogenen Bogenlänge oder die Vorgabe einer selektiven konstanten Bogenlänge. Im folgenden wollen wir uns auf die Beschreibung der zweiten Vorgehensweise beschränken.

Mit Hilfe einer Diagonalmatrix \underline{S}

$$\underline{S} = \begin{bmatrix} S_1 & & & & & & & & & \\ & S_2 & & & & & & & & \\ & & \cdot & & & & & & & \\ & & & \cdot & & & & & & \\ & & & & \cdot & & & & & \\ & & & & & \cdot & & & & \\ & & & & & & \cdot & & & \\ & & & & & & & \cdot & & \\ & & & & & & & & S_N & \\ \hline & & & & & & & & & 1 \end{bmatrix} \quad S_i = 1 \text{ oder } 0 \quad (4.3.8)$$

erhalten wir analog zu (4.3.1-2) den selektierten Zuwachs der Zustandsgrößen im k-ten Iterationszyklus und im aktuellen Inkrement als

$$\Delta \underline{r}^{(k)} = \underline{S} \cdot \begin{bmatrix} \Delta \underline{V}^{(k)} \\ \Delta \lambda^{(k)} \end{bmatrix}, \quad (4.3.9)$$

$$\Delta \underline{r} = \sum_{k=0}^M \Delta \underline{r}^{(k)}. \quad (4.3.10)$$

Der allgemeine Algorithmus folgt dann dem oben dargestellten Schema, wobei jedoch in den Beziehungen (4.3.3-7) die $\Delta \underline{r}^{(k)}$ und $\Delta \underline{r}$ gemäß (4.3.9-10) zu verwenden sind.

Abschließend sei darauf hingewiesen, daß der Grenzfall, N-1 Diagonalglieder $S_i=0$ und nur ein Diagonalglied $S_k=1$, den graphischen Interpretationen gemäß Bild 4.5 entspricht.

4.4 AKTUELLER STEIFIGKEITSPARAMETER

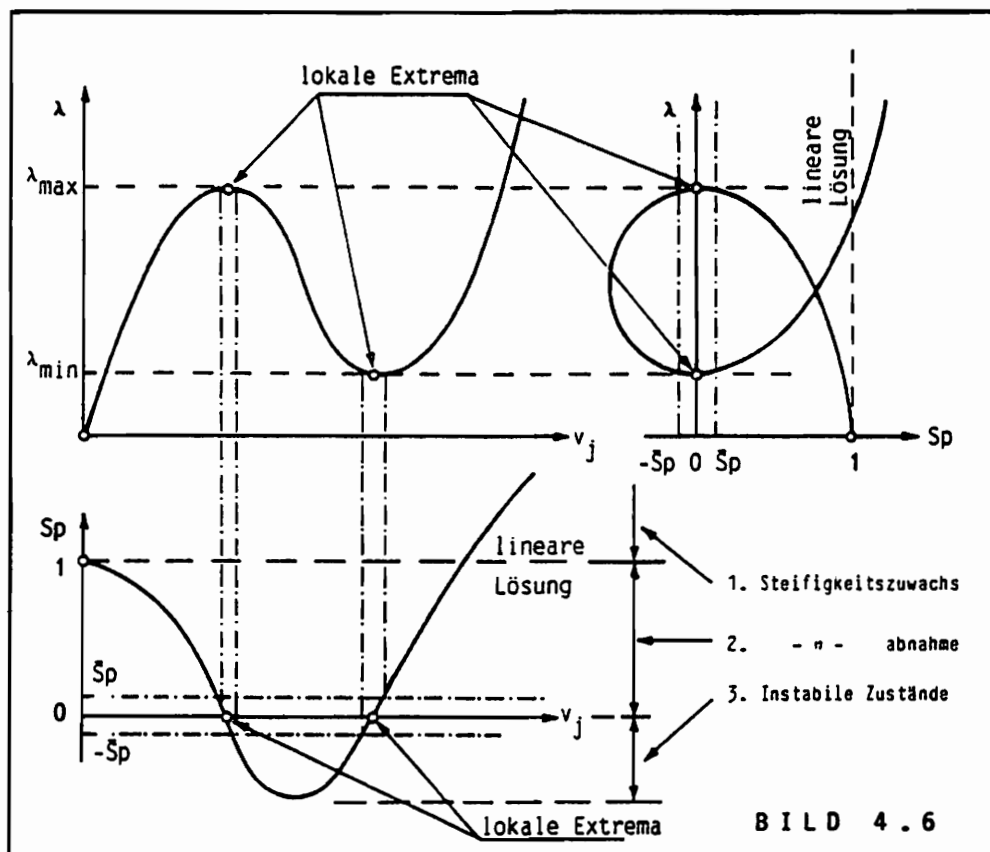
Die Verwendung des aktuellen Systemsteifigkeitsparameters zur Überwindung lokaler Extrema wurde erstmalig von Bergan und Soreide [1] vorgeschlagen. Es handelt sich dabei um ein Verfahren, den Zustand einer

Konstruktion durch den Quotienten der relativen inkrementellen virtuellen Arbeit der äußeren Kräfte im 1-ten Schritt zur relativen inkrementellen virtuellen Arbeit der äußeren Kräfte im aktuellen (N-ten) Schritt zu beschreiben:

$$S_p^N = \frac{\Delta W_{rel}^1}{\Delta W_{rel}^N} = \frac{\frac{(\Delta \underline{v}^1)^T}{\Delta \lambda^1} \cdot \frac{\Delta \underline{R}^1}{\Delta \lambda^1}}{\frac{(\Delta \underline{v}^N)^T}{\Delta \lambda^N} \cdot \frac{\Delta \underline{R}^N}{\Delta \lambda^N}} = \frac{\Delta \lambda^N}{\Delta \lambda^1} \cdot \frac{(\Delta \underline{v}^1)^T \underline{R}_{n-ref}^1}{(\Delta \underline{v}^N)^T \underline{R}_{n-ref}^N}$$

$$= \left(\frac{\Delta \lambda^N}{\Delta \lambda^1} \right)^2 \cdot \frac{(\Delta \underline{v}^1)^T \underline{K}_{n-ref}^1 \Delta \underline{v}^1}{(\Delta \underline{v}^N)^T \underline{K}_{n-ref}^N \Delta \underline{v}^N} \quad (4.4.1)$$

Eine graphische Darstellung gibt Bild 4.6. Der aktuelle Steifigkeitsparameter wird in Verbindung mit Aufhebung der Gleichgewichtsiteration benutzt und zwar gemäß Bild 4.6.



4.5 CALL STEUPA (PARA, DELTV, DELTP, ALAZY, AITZY, NFG, FEST)

Parameter	Typ	Bedeutung
PARA	external	STANLA - Standard-Last-Steuerung STANV - Standard-Verschiebungs-Steuerung LAVUSW - Last-Verschiebungs-usw-Steuerung VLAUSW - Verschiebungs-Last-usw-Steuerung KOBLME - Konstante Bogenlängenmethode
DELTV (Δv^*)	real	a) Verschiebungsinkrement, das als unabhängige Variable bei der Verschiebungssteuerung verwendet wird, b) Maximalverschiebungsinkrement, das als Kontrollvariable verwendet wird, c) Bogenlängenkrement,
DELTP ($\Delta \lambda^*$)	real	a) Lastfaktorinkrement, das bei der Laststeuerung und im ersten Schritt des Bogenlängenverfahrens verwendet wird, b) Maximallastfaktorinkrement, das als Kontrollvariable verwendet wird,
ALAZY	integer	Maximale Anzahl der Inkremente (nur im Modus Interaktiv),
AITZY	integer	Maximale Anzahl der Iterationen im Inkrementzyklus,
NFG	integer	Position des Verschiebungsinkrements (Δv^*) im Verschiebungsvektor,
FEST	logical	=.TRUE. Keine Änderung von NFG bei der Verschiebungssteuerung, =.FALSE. Das Programm sucht immer den Maximalverschiebungszuwachs und setzt dessen Position im Verschiebungsvektor auf NFG.

STEUPA ermöglicht die Wahl des inkrementell-iterativen Lösungsver-

fahrens und die Einstellung zugehöriger Steuerparameter. Optional in SKT1,3,7 jedoch beim ersten Aufruf in SKT1. !!!: Die automatische Umschaltung von Last- auf Verschiebungssteuerung und umgekehrt in den Modi LAVUSW und VLAUSW erfolgt

bei Verschiebungssteuerung, wenn $\Delta\lambda > \Delta\lambda^*$ (\rightarrow Laststeuerung)

bei Laststeuerung, wenn $\Delta v_{NFG} > \Delta v^*$ (\rightarrow Verschiebungssteuerung)

unter Berücksichtigung des Steuerungsfensters ($\Delta\lambda^*$, Δv^*) und des Sicherungsfensters ($\Delta\lambda^* \cdot GP$, $\Delta v^* \cdot GV$).

4.6 CALL SNEWRA (ASOITP, ITSOA, Z, MKBL)

Parameter	Typ	Bedeutung
ASOITP	integer	Anzahl der Lastinkremente ohne Iteration gemäß Bild 4.7,
ITSOA	integer	Anzahl der Iterationszyklen für das Standard-Newton-Raphson Verfahren ohne Aktualisierung der Steifigkeitsmatrix gemäß Bild 4.8,
Z	real	Selbstkorrigierender Faktor I. Stufe bei Lösung ohne Iteration [23],
MKBL	logical	=.TRUE. Konstante Bogenlängenmethode ohne Änderung der gesamten Steifigkeitsmatrix gemäß Bild 4.5 b. =.FALSE. Konstante Bogenlängenmethode mit Änderung der Orthogonalitätsbedingung (4.3.6) als aktivem Teil der Steifigkeitsmatrix gemäß Bild 4.4 und Bild 4.5 c.

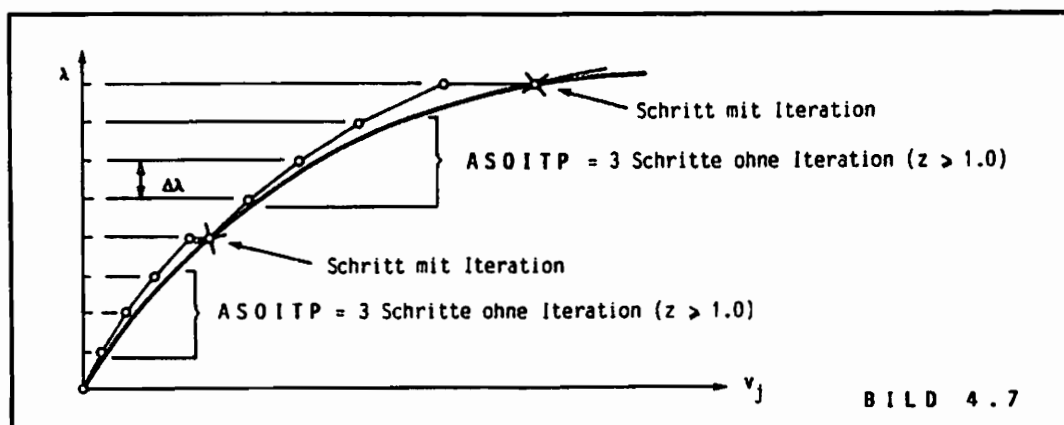
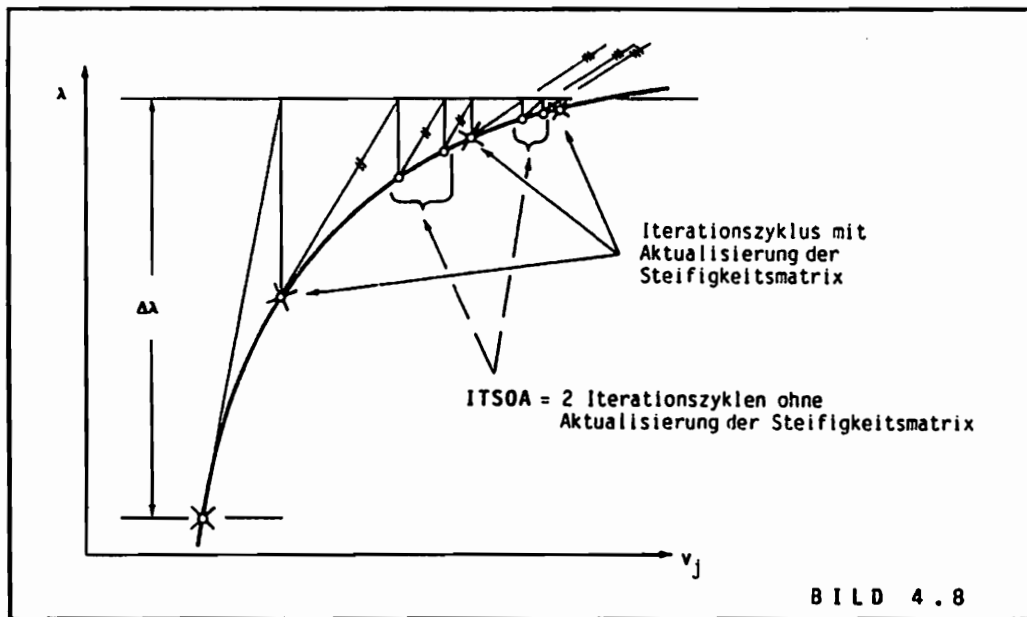


BILD 4.7



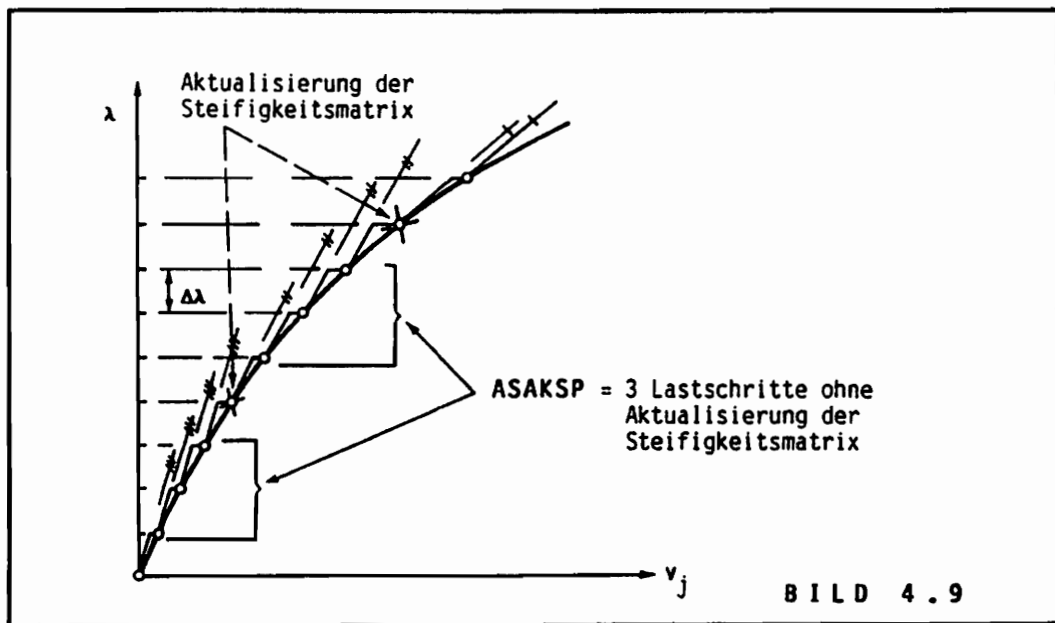
SNEWRA löst die nichtlinearen Gleichgewichtsbedingungen nach dem Standard-Newton-Raphson Verfahren mit den folgenden Variationsmöglichkeiten:

- keine Gleichgewichtsiteration in allen Inkrementen mit und ohne selbstkorrigierender Methode I. Stufe,
- periodische Gleichgewichtsiteration mit der Möglichkeit periodischer Aktualisierung der Steifigkeitsmatrix in den Iterationszyklen.

Optional in SKT1,3,7 jedoch beim ersten Aufruf in SKT1. !!!: MKBL ist nur von Bedeutung für $ITSOA > 0$.

4.7 CALL SNEWRA (ASAKSP, MKBL)

Parameter	Typ	Bedeutung
ASAKSP	integer	Anzahl der Inkremente bis zur nächsten Aktualisierung der Steifigkeitsmatrix gemäß Bild 4.9.
MKBL	logical	=.TRUE. Konstante Bogenlängenmethode ohne Änderung der gesamten Steifigkeitsmatrix gemäß Bild 4.5 b. =.FALSE. Konstante Bogenlängenmethode mit Änderung der Orthogonalitätsbedingung (4.3.6) als aktivem Teil der Steifigkeitsmatrix gemäß Bild 4.4 und Bild 4.5 c.



MNEWRA löst die nichtlinearen Gleichgewichtsbedingungen nach dem Modifizierten-Newton-Raphson Verfahren mit der Möglichkeit einer periodischen Aktualisierung der Steifigkeitsmatrix (${}_{n-}^{(i)}\underline{K} \equiv {}_n^1\underline{K}$). Optional in SKT1,3,7 jedoch beim ersten Aufruf in SKT1.

4.8 CALL EXIST (EXISTC, *Label)

Parameter	Typ	Bedeutung
EXISTC	logical	Antwort des Programmsystems ob: →.TRUE. Steifigkeitsmatrix existiert, →.FALSE. Steifigkeitsmatrix existiert nicht und muß aufgebaut werden.
*Label		Bedingter Sprung direkt zum Label CALL LOESE (...), wenn Gesamtsteifigkeitsmatrix (${}_{n-}^{(k)}\underline{K}$) und Restkraftvektor (${}_{n-}^{(k)}\underline{F}$) existieren und gültig sind (beim Standard-N-R Verfahren).

EXIST ist zum einen in Verbindung mit den Routinen MNEWRA und SNEWRA zu sehen und gibt die Rückmeldung, ob die Steifigkeits-

matrix existiert oder aufgebaut werden muß. Zum anderen ermöglicht EXIST den direkten Sprung zum Label CALL LOESE nach erfüllter Konvergenzabfrage über die Ungleichgewichtskräfte als letztes Kriterium. In Verbindung mit MNEWRA oder SNEWRA wenn entweder $ITSOA > 0$ oder gleichzeitig UNGLKA aufgerufen wird obligat in SKT4.

4.9 CALL KAWERB /ohne Parameter/

KAWERB ermöglicht in Verbindung mit der Routine AWPZKL das Überspringen der Belegungsrountinen für die Systemrandbedingungen, falls diese nicht geändert werden sollen. Optional in SKT1. !!! : Das Label wird in der Routine CALL AWPZKL (PARA, *Label) gesetzt.

4.10 CALL AKSTPR /ohne Parameter/

AKSTPR informiert das Programmsystem, daß mit dem aktuellen Steifigkeitsparameter weitergerechnet werden soll. Optional in SKT1,3,7.

4.11 CALL KOBSEL (VPE)

Parameter	Typ	Bedeutung
VPE	integer	a) Anzahl der Freiheitsgrade pro Knoten b) Anzahl der Systemfreiheitsgrade

KOBSEL reserviert Speicherplätze im Blank Common, Bereich CC, für die Diagonalglieder S_i gemäß (4.3.8). KOBSEL ermöglicht die Anwendung der selektiven konstanten Bogenlängenmethode gemäß Abschnitt 4.3 auf zwei verschiedenen Ebenen. Zum einen auf Knotenebene mit gleichförmiger selektiver Vorwahl der Komponenten in jedem Knoten (VPE gemäß a)) und zum anderen auf Systemebene mit spezieller selektiver Vorwahl der Komponenten (VPE gemäß b)). Im ersten Fall wird die Routine SELWER nur einmal für entsprechende Knotengrößen aufgerufen, während im zweiten Fall jede zu selektierende Systemgröße einen gesonderten Aufruf von SELWER erfordert. Optional in SKT1, jedoch in Verbindung mit SELWER obligat in SKT1.

4.12 CALL SELWER (IALF)

Parameter	Typ	Bedeutung
IALF	integer	Index des zu selektierenden Knoten- oder Systemfreiheitsgrades, $1 < IALF < VPE$.

SELWER bestimmt die zu selektierenden Knoten- oder Systemfreiheitsgrade ($S_i=0$ gemäß (4.3.8)). Optional in SKT2. !!!: Bei einem Restart muß SELWER noch einmal aufgerufen werden.

5. KONVERGENZKRITERIEN

Übersicht:

UNGLKA	-	Ungleichgewichtskraft - Kriterium
WUNGLK	-	Wichtung - Ungleichgewichtskräfte
TOTLAST	-	Totlast ("dead loads")
VZUWA	-	Verschiebungszuwachskriterium

Bei der Verwendung inkrementell-iterativer Lösungsverfahren kann je nach Problem, Art der Diskretisierung und Größe der gewählten Schrittweite ein sehr unterschiedliches Konvergenzverhalten vorliegen. Ein Iterationsverfahren erfordert daher ein sicheres Kriterium zur Feststellung, ob die Lösung im Rahmen der gewünschten Genauigkeit konvergiert hat oder möglicherweise divergiert ist.

Bei den Konvergenzkriterien unterscheidet man zwischen solchen, die auf die Ungleichgewichtskräfte angewendet werden, also ein Maß für den zu erreichenden Gleichgewichtszustand bilden, und solchen, die die Verschiebungszuwächse berücksichtigen. Weiter lassen sich Konvergenzkriterien in sogenannte "lokale" und "globale" Kriterien unterteilen. "Lokale" Konvergenzkriterien basieren auf einer Norm, die aus dem Zuwachs einiger Ungleichgewichtskraft- oder Verschiebungskomponenten im Iterationszyklus gebildet wird. Im Gegensatz dazu finden für "globale" Konvergenzkriterien Normen Verwendung, welche über alle Ungleichgewichtskraft- oder Verschiebungskomponenten definiert sind. Als Ungleichgewichts- und Verschiebungszuwächse können sowohl die absoluten als auch die relativen Größen gewählt werden.

Eine Diskussion zu diesen Fragestellungen insbesondere abhebend auf mechanische Probleme findet sich in der Arbeit von Bergan und Gough [25].

5.1 KONVERGENZKRITERIEN FÜR DIE UNGLEICHGEWICHTSKRÄFTE

a) Lokale Konvergenzkriterien

$$\varepsilon_1 > \max_{i=1,N} \left| \frac{n_i^{(k)}}{\Lambda_{i,ref}} \right|, \quad (5.1.1)$$

wobei als Bezugsgröße $\Lambda_{i,ref}$ wahlweise $\Delta R_i = \Delta \lambda \frac{(k)}{n_{ref,i}^R}$, $\frac{(k)}{n_{ref,i}^R}$ oder $\frac{(k)}{n_i^R}$ verwendet werden kann. Für Systeme mit nur einer Lastkomponente ($\frac{(k)}{n_{ref}^R}$) kann man die Bezugsgröße z.B. definieren als

$$\Lambda_{i,ref} = \frac{(k)}{n_{ref}^R} / a_i \quad (5.1.2)$$

mit a_i als Skalierungsfaktor, der z.B. proportional zu den inneren Knotenkräften sein kann

$$\epsilon_2 > \left[\frac{1}{N} \sum_{i=1}^N \left| \frac{n_i^J}{\Lambda_{i,ref}} \right|^2 \right]^{1/2} \quad (5.1.3)$$

b) Globales Konvergenzkriterium

$$\epsilon_3 > \frac{\| n^J \|}{\| \Lambda_{ref} \|} \quad (5.1.4)$$

mit $\| \dots \|$ als euklidische Norm. Für die Bezugsgröße Λ_{ref} kann wahlweise $\Delta \underline{R} = \Delta \lambda \frac{(k)}{n_{ref}^R}$, $\frac{(k)}{n_{ref}^R}$ oder $\frac{(k)}{n^R}$ benutzt werden.

!!!: Es erscheint nicht zweckmäßig, dieses globale Kriterium anzuwenden, falls das zu untersuchende Problem nur einen oder einige wenige belastete Freiheitsgrade aufweist. Der Grund dafür liegt in der großen Diskrepanz zwischen der Anzahl an Ungleichgewichtskräften und Lastkomponenten.

5.1.1 CALL UNGLKA (EPSG, EPSL, EPSLMX, VPK, INC, REF)

Parameter	Typ	Bedeutung
EPSG	real	ϵ_3 gemäß (5.1.4),
EPSL	real	ϵ_2 gemäß (5.1.3),
EPSLMX	real	ϵ_1 gemäß (5.1.1),
VPK	integer	Anzahl der Freiheitsgrade pro Knoten,
INC	logical	Parameter zur Steuerung der Bezugsgrößen
REF	logical	in (5.1.1), (5.1.3) und (5.1.4) mit

$$\Lambda_{i,ref} = \begin{cases} \Delta R_i & \text{wenn INC=.TRUE. u. REF=.FALSE.} \\ \frac{(k)}{n_{ref,i}^R} & \text{wenn INC=.TRUE. u. REF=.TRUE.} \\ \frac{(k)}{n_i^R} & \text{wenn INC=.FALSE. u. REF=.FALSE.} \end{cases}, \quad \Lambda_{ref} = \begin{cases} \Delta \underline{R} & \text{wenn INC=.TRUE. u. REF=.FALSE.} \\ \frac{(k)}{n_{ref}^R} & \text{wenn INC=.TRUE. u. REF=.TRUE.} \\ \frac{(k)}{n^R} & \text{wenn INC=.FALSE. u. REF=.FALSE.} \end{cases}$$

UNGLKA stellt fest, ob die Lösung im Rahmen der gewünschten Genauigkeit (ϵ_i , $i=1,2,3$) gemäß (5.1.1-4) konvergiert. Im Fall $\epsilon_i = 0.0$ wird das "i"-te Konvergenzkriterium und im Fall $VPK = 0$ werden die lokalen Konvergenzkriterien (5.1.1) und (5.1.3) nicht berücksichtigt. Optional in SKT1,3,7, jedoch beim ersten Aufruf in SKT1. In Verbindung mit WUNGLK obligat in SKT1.

5.1.2 CALL WUNGLK (IALF, WERT)

Parameter	Typ	Bedeutung
IALF	integer	"i"-te Position der Referenzkomponente $\Delta_{i,ref}$ (5.1.2) des Knotenkraftvektors $0 < IALF < VPK$,
WERT	real	Wert des zugehörigen Skalierungsfaktors "a _i " gemäß (5.1.2).

WUNGLK bestimmt den Skalierungsfaktor a_i gemäß (5.1.2). Voreinstellung $a_i = 1.0$. In Verbindung mit UNGLKA optional in SKT2,3,7.

5.1.3 CALL TOTLAST /ohne Parameter/

TOTLAST kann als "Spar"-Routine eingesetzt werden, wenn sogenannte "Totlasten" ("dead loads", $\overset{(k)}{R}_{n-ref} \equiv \text{const.}$) vorliegen, da sie nur einmalig die Bezugsgröße gemäß (5.1.1-4) bestimmt. Optional in SKT1. !!!: Bei einem Restart muß nach der Routine TOTLAST die Routine EMLAST noch einmal aufgerufen werden.

5.2 KONVERGENZKRITERIEN FÜR DIE VERSCHIEBUNGSZUWÄCHSE

a) Lokale Konvergenzkriterien

$$\epsilon_1 > \max_{i=1,N} \left| \frac{\overset{(k)}{\Delta v_i}}{\Delta v_{i,ref}} \right| \quad (5.2.1)$$

$$\epsilon_2 > \left[\frac{1}{N} \sum_{i=1}^N \left| \frac{\overset{(k)}{\Delta v_i}}{\Delta v_{i,ref}} \right|^2 \right]^{1/2} \quad (5.2.2)$$

b) Globales Konvergenzkriterium

$$\epsilon_3 > \frac{\|\Delta \underline{v}^{(k)}\|}{\|\Delta \underline{v}\|} . \quad (5.2.3)$$

5.2.1 CALL VZUWA (EPSG, EPSL, EPSLMX, VPK)

Parameter	Typ	Bedeutung
EPSG	real	ϵ_3 nach Gl. (5.2.3),
EPSL	real	ϵ_2 nach Gl. (5.2.2),
EPSLMX	real	ϵ_1 nach Gl. (5.2.1),
VPK	integer	Anzahl der Freiheitsgrade pro Knoten.

VZUWA ist analog zu UNGLKA (Abschnitt 5.1.1) aufgebaut, basiert jedoch auf den Konvergenzkriterien für die Verschiebungszuwächse (5.2.1-3). Optional in SKT1,3,7, jedoch beim ersten Aufruf in SKT1. !!!: Bei gleichzeitiger Anwendung von UNGLKA und VZUWA müssen die VPK übereinstimmen.

6. RESTART- und ARCHIVIERUNGSROUTINEN

Übersicht:

ARCHPA	-	Archivierungsparameter
RSTART	-	Restart

6.1 CALL ARCHPA (TAPEA, ASOAR)

Parameter	Typ	Bedeutung
TAPEA	character*6	Dateiname unter dem die durchgeführten Berechnungsschritte abgelegt werden,
ASOAR	integer	Anzahl der Schritte ohne Archivierung.

ARCHPA legt periodisch unter dem spezifizierten Dateinamen TAPEA eine zur Restartbearbeitung erforderliche Sicherungsdatei mit den Ergebnissen aller gewünschten Berechnungsschritte (ASOAR) an. Optional in SKT1 vor der Routine SICHERN.

6.2 CALL RSTART (TAPER, ASRES, ONEAN, *Label)

Parameter	Typ	Bedeutung
TAPER	character*6	Dateiname (entspricht dem Namen der Archivierungsdatei),
ASRES	integer	Aktueller Berechnungsschritt für den Restartbeginn,
ONEAN	logical	=.TRUE. Restart ohne Änderung der Steuerparameter, d.h. es muß ein direkter Sprung auf das spezifizierte Label (z.B. IDIAG) erfolgen, =.FALSE. Restart mit Änderung der Steuerparameter, d.h. es muß ein direkter Sprung zum spezifizierten Label (z.B. STEUPA) erfolgen,
*Label		s. ONEAN .

RSTART führt unter Verwendung der Archivierungsdatei einen Restart

durch. Dabei ist es nicht möglich, dimensionierte Parameter zu ändern, d.h. Routinen wie z.B. STATIK, PROBLEM etc. dürfen nicht aufgerufen werden und RSTART muß der erste MESY-Aufruf sein. Wenn TAPEA = TAPER ist, sucht RSTART den Speicherbereich für ASRES, führt dann den Restart durch und ARCHPA schreibt weiterhin in die bestehende Archivierungsdatei. Selbstverständlich wird bei unterschiedlichen Namen der Archivierungsvorgang in TAPEA neu begonnen. Optional in SKT1. !!!: Auf die I/O Kanäle 3 und 4 darf nicht zugegriffen werden.

7. VERWALTUNGSFUNKTIONEN

Unter dem Begriff Verwaltungsfunktion ist die Möglichkeit zu verstehen, Speicherplätze direkt zu adressieren. Dies führt zu Zeit- und Platzersparnis, da keine Routinen benötigt werden um Daten von Individualspeicherplätzen auf MESY-Speicherplätze zu transferieren.

7.1 **IRCEAD (I)** - Adresse der Elementsteifigkeitsmatrix, I - Anzahl der Elementfreiheitsgrade.

7.2 **IRDEAD ()** - Adresse der Elementdämpfungsmatrix.

7.3 **IRMEAD ()** - Adresse der Elementmassenmatrix.

7.4 **IRHEAD ()** - Adresse der Elementhilfsmatrix.

7.5 **IRPEAD ()** - Adresse des Elementlastvektors.

7.6 **IRSOAD ()** - Adresse des Elementrestkraftvektors.

7.7 **IRVOAD ()** - Adresse des Elementverschiebungsvektors.

7.8 **IRGSAD ()** - Adresse des Elementverformungsgeschichtenvektors.

!!!: Die mit diesen Funktionen ermittelten Adressen gelten für auf Elementebene bezogene Felder, die Werte in Elementkoordinaten enthalten. Bei den Verwaltungsfunktionen 7.2-8 handelt es sich um Funktionen ohne Parameter. Optional in SKT2-7.

Die ersten drei Funktionen entsprechen der Routine **ELORG**, jedoch muß die Funktion **IRCEAD** zuerst aufgerufen werden.

Die Funktion 7.5 entspricht der Routine **ELORGP**.

Die Funktion 7.6 entspricht der Routine **RSL**.

8. FUNKTIONEN ZUR BESTIMMUNG CHARAKTERISTISCHER SYSTEMGRÖSSEN

Die im folgenden aufgeführten Funktionen ermöglichen den direkten Zugriff auf im Blank Common, Bereich CC, abgelegte charakteristische Systemgrößen.

- 8.1 **SLAMB** () - $\lambda^{(k)}$ - Proportionalitätsfaktor für den Lastvektor gemäß (1.2.4).
- 8.2 **DLAMB** () - $\Delta\lambda$ - Inkrement des Proportionalitätsfaktors.
- 8.3 **UKN1** () - Aktueller Wert der Norm gemäß (5.1.1).
- 8.4 **UKN2** () - Aktueller Wert der Norm gemäß (5.1.3).
- 8.5 **UKN3** () - Aktueller Wert der Norm gemäß (5.1.4).
- 8.6 **VZWN1** () - Aktueller Wert der Norm gemäß (5.2.1).
- 8.7 **VZWN2** () - Aktueller Wert der Norm gemäß (5.2.2).
- 8.8 **VZWN3** () - Aktueller Wert der Norm gemäß (5.2.3).
- 8.9 **EUNDV** () - $\|\Delta v\|$ - Euklidische Norm des Verschiebungsincrements.
- 8.10 **DETN** () - Determinante der Steifigkeitsmatrix im aktuellen Inkrementzyklus.
- 8.11 **IVSTEU** () - Index des aktuellen Steuer-Verschiebungsincrements.
- 8.12 **DVSTEU** () - Aktuelles Steuer-Verschiebungsincrement.
- 8.13 **IMAXDV** () - Index des maximalen Verschiebungsincrements im aktuellen Inkrementzyklus.
- 8.14 **DVMAX** () - Maximales Verschiebungsincrement im aktuellen Inkrementzyklus.
- 8.15 **AKST** () - Aktueller Steifigkeitsparameter gemäß (4.9.1).
- 8.16 **IAASCH** () - Gesamtzahl aller Inkremente.

8.17 **DXO (I)** - Analog zur Funktion X0, jedoch für die inkrementellen Verschiebungen, I-Index des Systemfreiheitsgrades.

!!!: Mit Ausnahme der Funktion 8.17 handelt es sich um Funktionen ohne Parameter. Optional in SKT3-7.

9. PSEUDO-VOLLAUTOMATISCHE UND INTERAKTIVE STEUERUNG

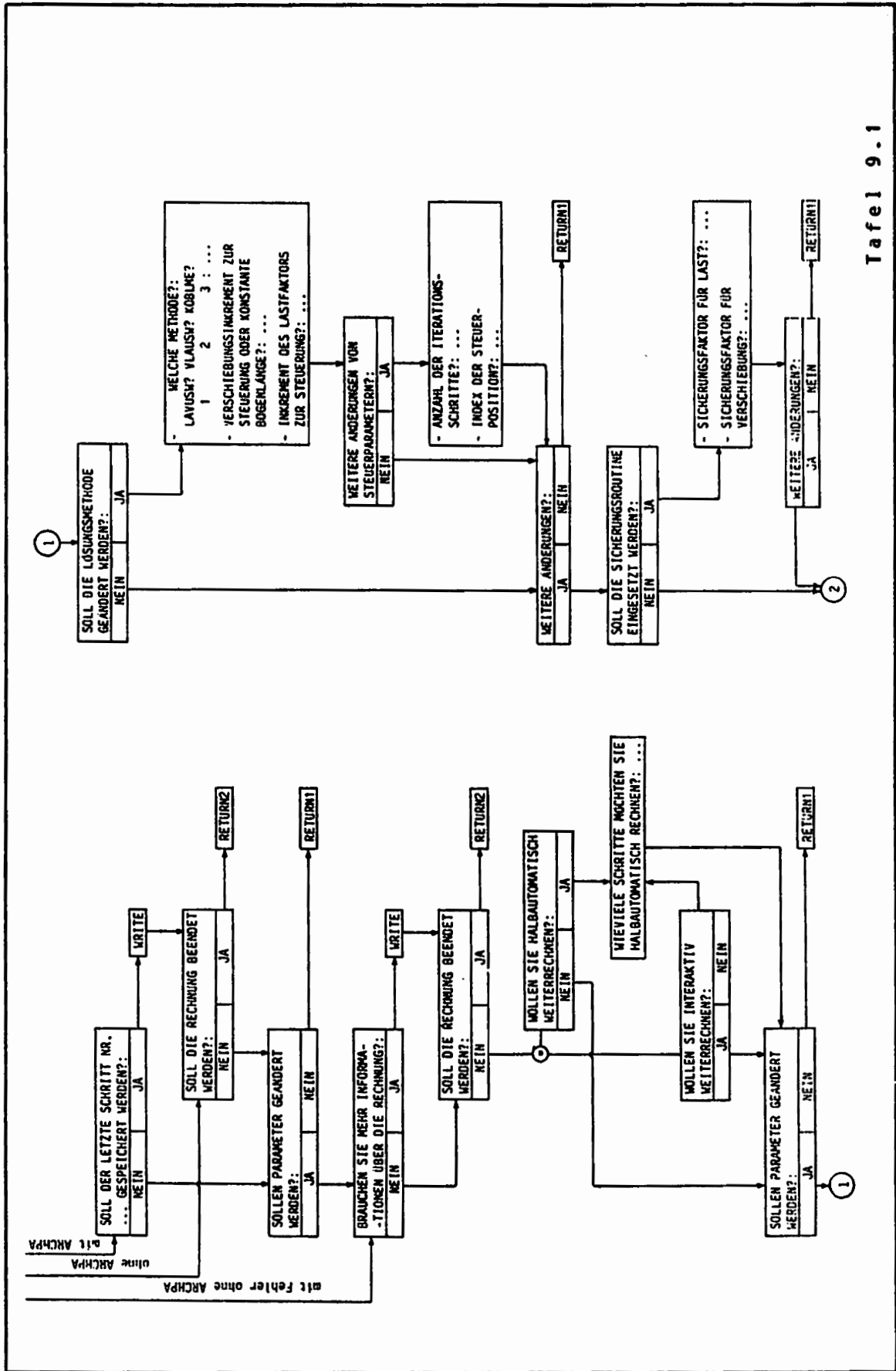
Übersicht:

AENSCHR	-	Änderung der Schrittweite
INTERA	-	Vorbereitung interaktive Steuerung
INTSTU	-	Interaktive Steuerung

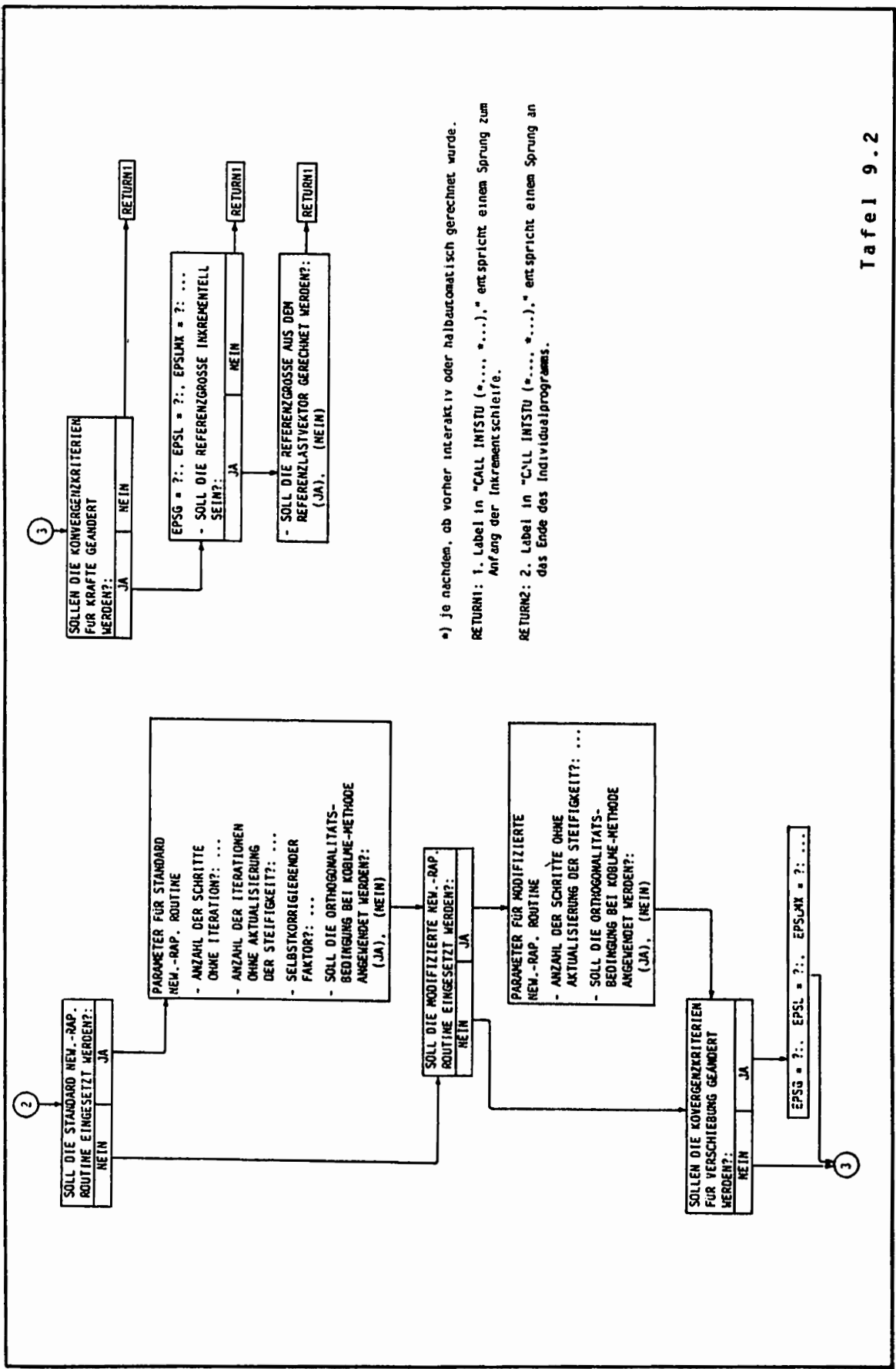
Im Kapitel 4 haben wir verschiedene Strategien und zugehörige FORTRAN-Routinen zur Lösung der diskretisierten nichtlinearen Grundgleichungen (1.2.8) vorgestellt. Die Effizienz dieser Verfahren hängt jedoch stark von der Wahl geeigneter Zuwächse der Steuergrößen, d. h. Lasten, Verschiebungen und Bogenlängen ab. Zu groß voreingestellte Zuwächse können dazu führen, daß keine Konvergenz erreicht wird oder die Lösung in einen nicht gewünschten Anziehungsbereich der Last-Verschiebungs-Kurve abdriftet. Zu klein voreingestellte Zuwächse erhöhen auf der anderen Seite die Rechenzeit und damit die Kosten einer nichtlinearen Analyse in einem nicht vertretbaren Maß. Ziel ist es daher, variable, d.h. möglichst optimal an die charakteristischen Phänomene der Antwortspektren angepaßte Zuwächse einprägen zu können.

Darüberhinaus hat die Analyse einer Vielzahl meist a priori unbekannter nichtlinearer Probleme gezeigt, daß trotz ausgefeilter Steuerroutinen in einigen Bereichen der Lösung nur eine interaktive Steuerung, bei der die Erfahrung des Bearbeiters direkt umgesetzt werden kann, zu brauchbaren Ergebnissen führt. Als Beispiele sind sprunghafte Wechsel der Deformationsmuster beim "looping-Effekt" oder eng zusammenliegende Gleichgewichtspfade mit unterschiedlichen und oftmals wechselnden Konvergenzradien zu nennen.

In diesem Kapitel werden daher drei Routinen zum einen für die pseudo-vollautomatische und zum anderen für die interaktive Steuerung vorgestellt. Ein detailliertes Flußdiagramm des gesamten interaktiven Dialogs findet sich in den Tafeln 9.1-2.



Tafel 9.1



*) je nachdem, ob vorher interaktiv oder halbautomatisch gerechnet wurde.
 RETURN1: 1. Label in "CALL INTSTU (*...*,*...*,*...*)" entspricht einem Sprung zum Anfang der Inkrementenschleife.
 RETURN2: 2. Label in "CALL INTSTU (*...*,*...*,*...*)" entspricht einem Sprung an das Ende des Individualprogramms.

Tafel 9.2

9.1 CALL AENSCHR (MDV, MDLAM, MINF, MINIT, OUTACT1, *Label)

Parameter	Typ	Bedeutung
MDV	real	Absoluter Verschiebungszuwachs = DELTV in STEUPA,
MDLAM	real	Absoluter Lastzuwachs = DELTP in STEUPA,
MINF	real	kleinster zulässiger Verkleinerungsfaktor für die Schrittweite; muß positiv sein,
MINIT	integer	Anzahl der Iterationszyklen, bis zu der eine stufenweise Vergrößerung der Schrittweite bis auf die zulässigen Maxima (MDV,MDLAM) erfolgt,
OUTACT1	logical	=.TRUE.: Die Aktivitäten von AENSCHR und deren Ursachen werden auf Kanal 6 ausgegeben,
*Label		Rücksprunglabel für den Fall, daß in LOESE keine Konvergenz erfolgt; bezeichnet das Ende der Inkrement-Schleife.

AENSCHR ändert automatisch die Schrittweite bzw. die Bogenlänge bei Konvergenzschwierigkeiten und unterstützt bei LAVUSW und VLAUSW die Umschaltung zwischen Last- und Verschiebungssteuerung in kritischen Bereichen des Gleichgewichtspfades. Optional in Sektion 6. !!!: CALL AENSCHR muß direkt auf CALL LOESE folgen und die Labelnummer des zweiten Rücksprungparameters (hier 650) von CALL LOESE tragen.

Beispiel: CALL LOESE (..., *400, *650)
 650 CALL AENSCHR (..., *799)

9.2 CALL INTERA (I, O, Full)

Parameter	Typ	Bedeutung
I	integer	Input-Kanalnummer,
O	integer	Output-Kanalnummer,

FULL logical =.TRUE. voll interaktive Rechnung,
 =.FALSE. halbautomatische interaktive
 Rechnung mit ALAZY Inkrementen (s. Abschn.
 4.5).

INTERA bereitet die interaktive Steuerung der nichtlinearen Berechnung vor. Optional in SKT1, jedoch in Verbindung mit INTSTU obligat in SKT1. !!!: Auf die Kanalnummern 3,4,6 darf nicht zugegriffen werden.

9.3 CALL INTSTU (*Label1, *Label2)

Parameter	Typ	Bedeutung
*Label1		Beginn der Inkrement-Schleife (Anfang SKT3),
*Label2		Ende des Individualprogramms (SKT8).

INTSTU ermöglicht die interaktive Berechnung gemäß Flußdiagramm (s. Tafeln 9.1-2) !!!: Man beachte, daß im Gegensatz zur Standard MESY3 Quelle [19] immer eine modifizierte LOESE-Routine mit einem zusätzlichen Label auf Parameterposition zu verwenden ist:

```
CALL LOESE (PARA, *Label3, *Label4)
```

*Label3 markiert analog zum alten LOESE-Aufruf den Iterationsanfang. Mit *Label4 kann zusätzlich die Sektion 7, "Ergebnisse nachbereiten", übersprungen werden, wenn die Berechnung nicht konvergiert hat, z.B. mit einem direkten Sprung zur Sektion 7A. Optional in Sektion 7A.

10. ANWENDUNGSBEISPIELE

Die in diesem Kapitel vorgestellten Beispiele sollen zur Überprüfung der in den vorangegangenen Kapiteln beschriebenen Steueralgorithmien und -routinen dienen. Sie wurden als hochgradig nichtlineare Probleme aus dem Anwendungsbereich dünner elastischer Strukturen ausgewählt. Zu einer genauen Erfassung der komplizierten Systemantworten sind komplexe Modelltheorien und leistungsstarke Näherungsverfahren unumgänglich, deren Anwendung erst durch die sprunghafte Entwicklung elektronischer Rechenanlagen und computerorientierter Algorithmen möglich wurde. Im folgenden soll zunächst für den interessierten Leser ein kurzer Abriss theoretischer Grundlagen der zur Berechnung herangezogenen Elementmodelle gegeben werden.

10.1 MODELLTHEORETISCHE BETRACHTUNGEN

Zu einer wegweisenden Entwicklung auf dem Gebiet der Schalentheorie führte die Anwendung des polaren Zerlegungssatzes der Kontinuumsmechanik [26]. Mit seiner Hilfe wurde es möglich, unabhängig von den Materialdehnungen die Größe der Rotationen einzuschränken. Unter der Annahme kleiner Dehnungen von der Ordnung $O(\eta)$, $\eta \ll 1$, schlägt Pietraszkiewicz [26,27] eine Klassifizierung von Schalentheorien nach der Größenordnung der Rotationswinkel vor: kleine Rotationen $\omega \leq O(\eta)$, moderate Rotationen $\omega = O(\eta^{1/2})$, große Rotationen $\omega = O(\eta^{1/4})$ und finite Rotationen $\omega \geq O(1)$. Darauf aufbauend sind in der Literatur eine Vielzahl geometrisch nichtlinearer Schalentheorien bei Auftreten finiter, großer und moderater Rotationen in reiner Verschiebungsformulierung entwickelt worden und zur Anwendung gelangt (s. [28,29] und dort zitierte Arbeiten). Dabei ist zu beobachten, daß Schalenrotationen um Tangenten zur Mittelfläche bis in den Bereich finiter Rotationen hinein auftreten können. Gleichzeitig bleiben die Rotationen um Normalen in der Regel klein. Eine Verfeinerung des in [26,27] vorgeschlagenen Klassifikationsschemas führte in der Arbeit [29] zu leistungsfähigen Schalenvarianten bei Auftreten großer Rotationen mit hoher Praktikabilität hinsichtlich der numerischen Anwendung. Auf der Basis umfangreicher vergleichender Untersuchungen [28,29] wird daher zur Erfassung

von Schalendeformationen mit groß/kleinen Rotationen die in Tafel 10.1 dargestellte Theorie vorgeschlagen. Sämtliche restriktive Annahmen sollten für die Mehrzahl aller Ingenieur-Anwendungen erfüllt sein. Zusätzlich sind in den Tafeln 10.2-3 zwei exakte Theorien finiter Rotationen angegeben, die in [28,30,31] für die Sonderfälle der axial-symmetrischen Deformationen von Rotationsschalen und der ebenen Balkenbiegung hergeleitet wurden. Im Hinblick auf modelltheoretische Grundlagen zur Erfassung physikalischer Nichtlinearitäten unter Verwendung eines Schichten- und mehrerer vereinfachter Modelle sei auf [32] verwiesen. Der Einfluß gleichförmiger richtungstreuer Drucklasten auf das Tragverhalten von Balken und Schalen unterschiedlicher Nichtlinearität wurde in [33] untersucht.

10.2 Finite-Element-Formulierung

Auf der Basis der in den Tafeln 10.1-3 angegebenen Schalen- und Balkentheorien wurden in [28,34] die FE-Algorithmen für ein doppelt gekrümmtes Schalenelement, ein Ringelement zur Berechnung axialsymmetrischer Deformationen von Rotationsschalen, und ein Balkenelement erstellt. Alle drei Elemente entstammen der Klasse der sogenannten "High-Precision" Elemente. Besondere Bedeutung kommt dabei der exakten Geometriebeschreibung sowie der Wahl geeigneter bereichsweiser Verschiebungsansätze zu, die gewissen Vollständigkeits- und Kontinuitätsanforderungen längs der Elementränder genügen müssen.

Der FE-Algorithmus für das doppelt gekrümmte Dreieckelement mit 54 Freiheitsgraden basiert auf den Arbeiten [35,36]. Es wird eine exakte Geometriebeschreibung in den 21 Integrationspunkten einer numerischen Integration nach Gauß durchgeführt. Als Ansatzfunktionen finden quintische Polynome sowohl für die Tangential- als auch die Normalverschiebung Verwendung. Eine Besonderheit dieses Elementes liegt in der speziellen vektorisierten Aufbereitung der Algorithmen für die Steifigkeitsmatrix und den Restkraftvektor. In Verbindung mit einer Programmierung in Vektor-FORTRAN 200 und der Realisierung auf dem Bochumer Vektorrechner CDC Cyber 205 können dadurch deutliche Rechenzeitbeschleunigungen gegenüber üblichen Skalarrechnern erreicht werden [34].

Beim Schalenring- und Balkenelement mit jeweils 8 Freiheitsgraden

SCHALENTHEORIE GROSS/KLEINER ROTATIONEN

FUNKTIONAL VOM GESAMTPOTENTIAL

$$\begin{aligned}
 J_p(\underline{u}) = & 1/2 \iint_M [H^{\alpha\beta\lambda\mu} (\gamma_{\alpha\beta} \gamma_{\lambda\mu} + h^2/12 x_{\alpha\beta} x_{\lambda\mu}) - 2p \cdot \underline{u}] dA \\
 & - \int_{C_f} [T_{vv}^* u_v + T_{tv}^* u_t + T_{nv}^* w + H_{vv}^* \eta_v + H_{tv}^* \eta_t + H_{nv}^* (n - 1)] ds \\
 & + \iint_M q[-1/2 u^\lambda \varphi_\lambda + w(1 + 1/2 \theta_x^*)] dA
 \end{aligned}$$

KINEMATISCHE BEZIEHUNGEN

$$\begin{aligned}
 \gamma_{\alpha\beta} &= \theta_{\alpha\beta} + 1/2 \varphi_\alpha \varphi_\beta + 1/2 \theta_\alpha^\lambda \theta_{\lambda\beta} \\
 x_{\alpha\beta} &= -1/2 [(\delta_\alpha^\lambda + \theta_\alpha^\lambda) \varphi_{\lambda|\beta} + (\delta_\beta^\lambda + \theta_\beta^\lambda) \varphi_{\lambda|\alpha} - \varphi_\alpha \theta_x^* |_\beta - \varphi_\beta \theta_x^* |_\alpha] \\
 n_v &= -\varphi_v, \quad n_t = -\varphi_t, \quad n = 1 - 1/2 \varphi_t^2 - 1/2 n_v^2
 \end{aligned}$$

EINGEPRÄGTE LASTEN

p - Flächenlast

$$\left. \begin{aligned}
 T_v^* &= T_{vv}^* \underline{v} + T_{tv}^* \underline{t} + T_{nv}^* \underline{n} - \text{Randkräfte} \\
 H_v^* &= H_{vv}^* \underline{v} + H_{tv}^* \underline{t} + H_{nv}^* \underline{n} - \text{statisches Randmoment}
 \end{aligned} \right\} \text{"Totlasten"}$$

q = Intensität einer verformungsabhängigen gleichförmigen Drucklast

NOTATION s. [29]

AXIALSYMMETRISCHE DEFORMATIONEN VON ROTATIONSSCHALEN
THEORIE FINITER ROTATIONEN

FUNKTIONAL VOM GESAMTPOTENTIAL

$$\begin{aligned}
 J_p(\underline{u}) = & \pi \int_{S_1}^{S_2} \{ Eh/1-\nu^2 [(\gamma_{SS} + \nu\gamma_{\theta\theta})^2 + (\gamma_{\theta\theta} + \nu\gamma_{SS})^2 \\
 & + Eh^3/12(1-\nu^2) [x_{SS} + \nu x_{\theta\theta})^2 + (x_{\theta\theta} + \nu x_{SS})^2] - 2p \cdot \underline{u} \} r ds \\
 & - 2\pi r [F^* \cdot \underline{u} + H^* (\bar{\eta} - \eta)]_{S_1}^{S_2} \\
 & + 2\pi \int_{S_1}^{S_2} q [-1/2 \varphi_S u + (1 + 1/2 \theta_{SS} + 1/2 \theta_{\theta\theta}) w] r ds
 \end{aligned}$$

KINEMATISCHE BEZIEHUNGEN

$$\begin{aligned}
 \gamma_{SS} &= \theta_{SS} + 1/2 \theta_{SS}^2 + 1/2 \varphi_S^2, & \gamma_{\theta\theta} &= \theta_{\theta\theta} \\
 x_{SS} &= -\varphi_S'(1 + \theta_{SS}) + \varphi_S \theta_{SS}', & x_{\theta\theta} &= -(\sin\theta/r)\varphi_S - \sigma_\theta \theta_{SS} \\
 \underline{u}(s, \theta) &= u(s) \underline{a}_S + w(s) \underline{n} \\
 \bar{\eta} &= \lambda_S^{-1} (-\varphi_S \underline{a}_S + (1 + \theta_{SS}) \underline{n}), & \lambda_S &= [(1 + \theta_{SS})^2 + \varphi_S^2]^{1/2}
 \end{aligned}$$

EINGEPRÄGTE LASTEN

\underline{p} - Flächenlast
 \underline{F}^* - Randkraft
 \underline{H}^* - statisches Randmoment
 q = Intensität einer verformungsabhängigen gleichförmigen Drucklast
 } "Totlasten"

NOTATION s. [30]

**EBENE BALKENBIEGUNG
THEORIE FINITER ROTATIONEN**

FUNKTIONAL VOM GESAMTPOTENTIAL

$$J_p(\underline{u}) = 1/2 \int_{S_1}^{S_2} (C_1 \gamma^2 + C_2 \kappa^2 - 2\underline{p} \cdot \underline{u}) ds - [\underline{F}^* \cdot \underline{u} + \underline{H}^* \cdot (\underline{\bar{n}} - \underline{n})]_{S_1}^{S_2} + \int_{S_1}^{S_2} q[-1/2 \varphi u + (1 + 1/2 \vartheta)w]$$

C_1 = Dehnsteifigkeit, C_2 = Biegesteifigkeit

KINEMATISCHE BEZIEHUNGEN

$$\gamma = \vartheta + 1/2 \vartheta^2 + 1/2 \varphi^2, \quad \kappa = -\varphi' - \varphi' \vartheta + \varphi \vartheta'$$

$$\underline{u} = u(s)\underline{e} + w(s)\underline{n}$$

$$\underline{\bar{n}} = \lambda^{-1} (-\varphi \underline{e} + (1 + \vartheta)\underline{n}), \quad \lambda = [(1 + \vartheta)^2 + \varphi^2]^{1/2}$$

EINGEPRÄGTE LASTEN

\underline{p} - Flächenlast

\underline{F}^* - Randkraft

\underline{H}^* - statisches Randmoment

} "Totlasten"

q = Intensität einer verformungsabhängigen gleichförmigen Drucklast

NOTATION s. [31]

Tafel 10.3

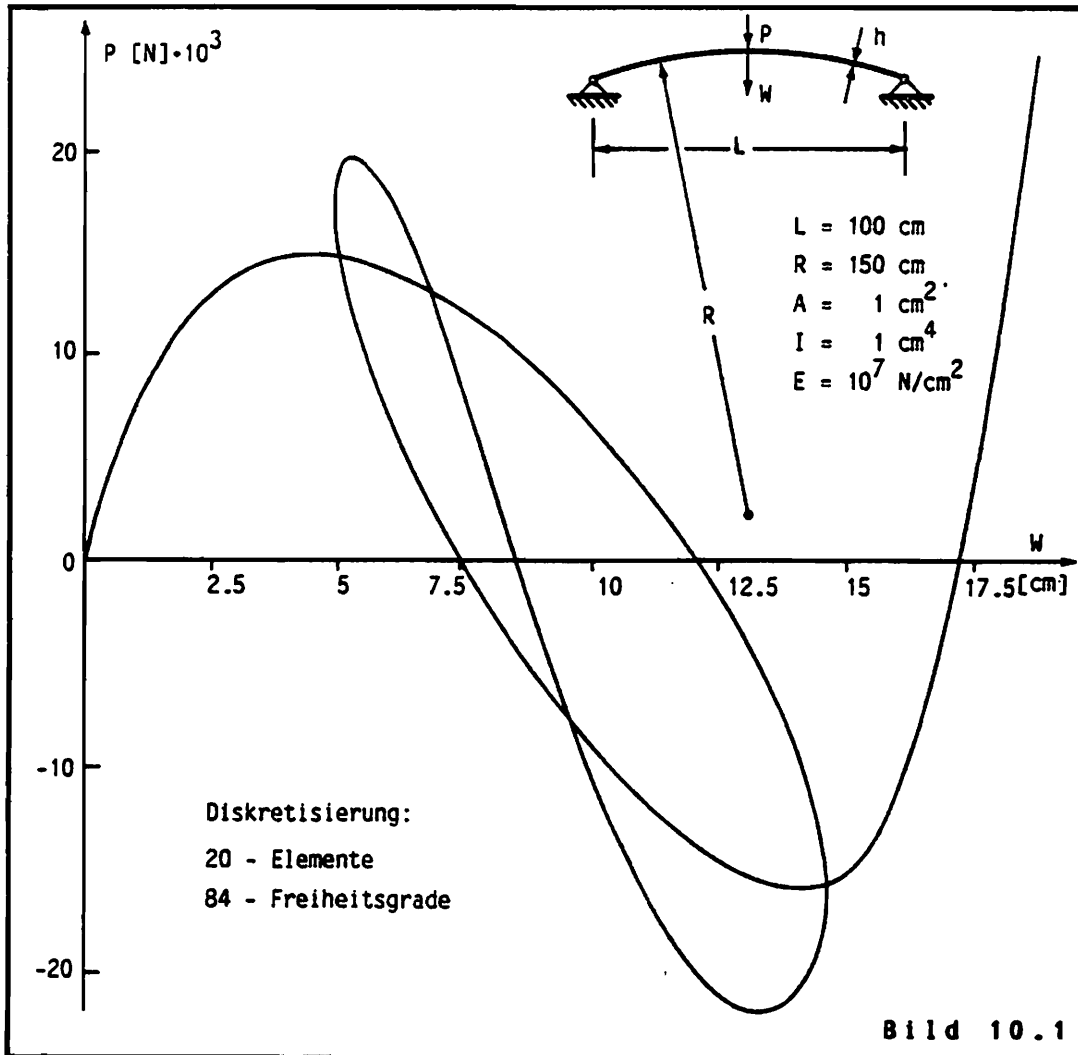
finden als Ansatzfunktionen kubische Polynome für die Tangential- und Normalverschiebung Verwendung. Die Geometrie wird exakt in den Integrationspunkten einer 3-Punkt Gauß Integration beschrieben. Die Einfachheit der reduzierten Kinematiken (s. Tafeln 10.2-3) ermöglichte es mit Hilfe dieser Elemente, eine umfangreiche vergleichende Untersuchung von 14 Schalentheorien durchzuführen [29]. Alle Elementroutinen sind als vollparameterversorgte Routinen mit zugehörigen Kinematikroutinen im Programmsystem MESY3 implementiert. Sie werden durch entsprechende Aufrufe im Individualprogramm aktiviert.

10.3 BALKENBEISPIELE

10.3.1 KREISBÖGEN UNTER EINZELLAST

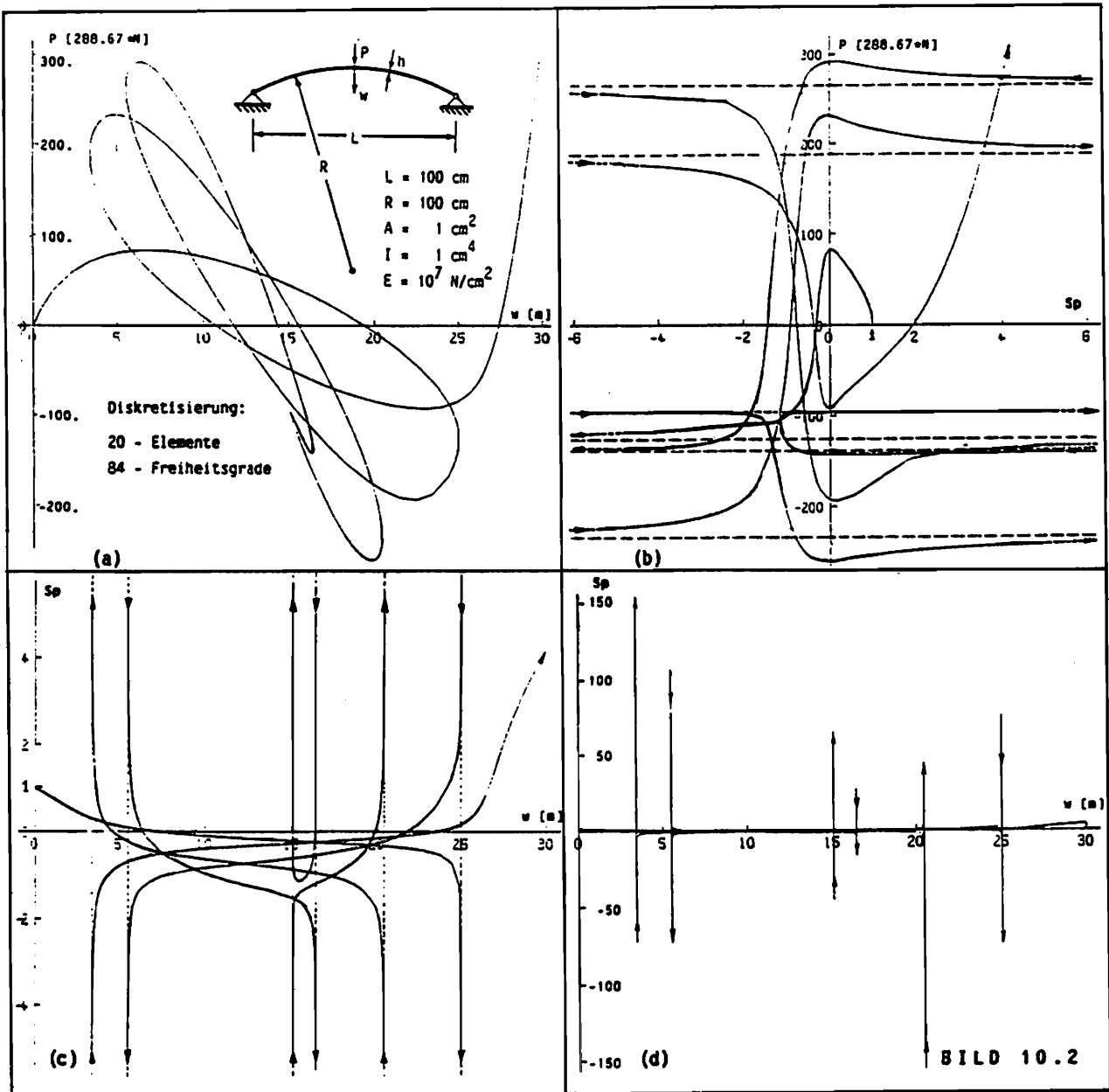
In diesem ersten Anwendungsbeispiel soll das hochgradig nichtlineare Tragverhalten von elastischen Kreisbögen unter Einzellasten analysiert werden. Durch Änderung des Krümmungsradius bei sonst konstant gehaltenen Problem-, d.h. Geometrie- und Materialdaten, lassen sich Strukturantworten mit unterschiedlichsten Phänomenen erzeugen. Das anfangs reine Spannungsproblem (großer Krümmungsradius) verändert sich auf diese Weise zunächst in ein Durchschlagsproblem ohne Verzweigung und später mit Verzweigung in einen asymmetrischen Lösungspfad [28,37]. Weitere Verkleinerungen des Krümmungsradius führen dann zum sogenannten "Looping"-Effekt mit ansteigender Zahl der relativen Extrema und Durchdringungen der Lösungen. Genauere Untersuchungen zeigen, daß dabei in Punkten vertikaler Tangenten des Lösungspfades die Anzahl der Halbwellen des Deformationsmusters um jeweils zwei entweder ansteigt oder abnimmt. Diese Erscheinung läßt sich physikalisch anschaulich deuten: Längs der Last-Verschiebungskurve gibt es Konfigurationen mit hoher globaler Steifigkeit der Struktur (vertikale Tangenten). Lastzuwächse führen zunächst zu einer lokalen Störung, die ab einer gewissen Größenordnung in ein gleichmäßiges Deformationsmuster mit zwei weiteren Halbwellen übergeht. Dieses Verhalten ist über die Biegesteifigkeit und die Geometrie parametrisiert. Insbesondere tiefe und biegeeweiche Bögen liefern unter der Wirkung von Einzellasten höherwellige Deformationsmuster instabiler Natur.

Die Bilder 10.1-3 zeigen das symmetrische Tragverhalten mit fortschreitendem "Looping" der Last-Verschiebungskurven für eine Serie von Kreisbögen mit variablen Krümmungsradien ($R_i = 150., 100., 75.$). Die FE-Diskretisierung erfolgte jeweils mit 20 Balkenelementen, die auf der Theorie finiter Rotationen gemäß Tafel 10.3 aufgebaut sind (s. auch [28,38]). Die zugehörige Makro Elementroutine ist HPBMAK. Sie wird durch die Integrationsroutine HPBINT und die Problemdatei gemäß DATA-Anweisungen versorgt. Systemrandbedingungen werden in allen Individualprogrammen (INDPROi) unabhängig von den Problemdatei durch EINGSV-Aufrufe in SKT3 realisiert. Die Parameter TYPEL, THEORY, und

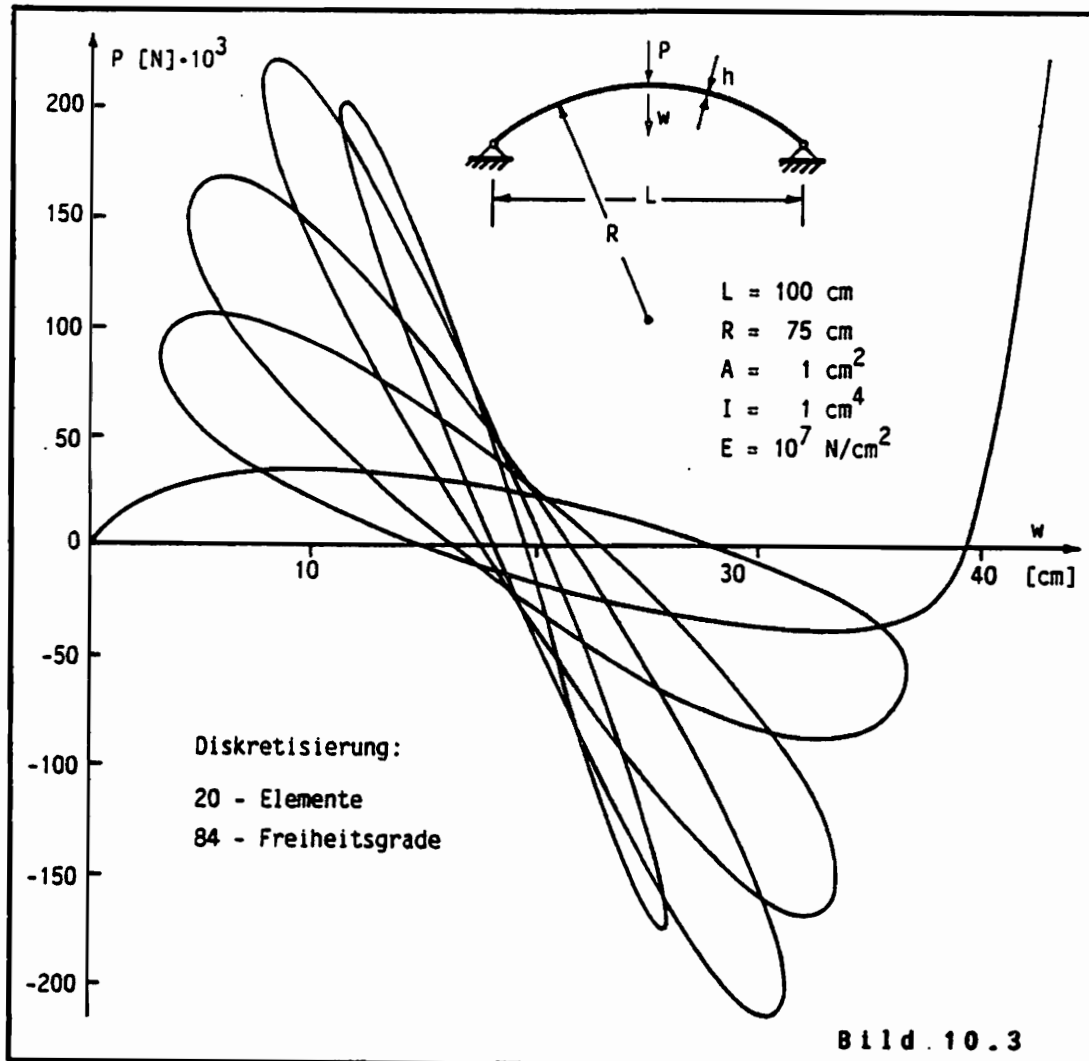


PRESS geben an, ob Elementlasten vorliegen (TYPEL=1) oder nicht (TYPEL=0), ob mit einer Theorie moderater Rotationen (THEORY=0) [27], der Theorie finiter Rotationen (THEORY=1), verformungsunabhängigen Elementlasten (PRESS=0) oder verformungsabhängigen Elementlasten (PRESS=1) gerechnet werden soll.

In Bild 10.4 ist das symmetrische Tragverhalten eines bekannten Halbkreisbogen-Beispiels (s. [19,28] und dort zitierte Arbeiten) wiedergegeben. Das fortschreitende "Looping" ist hier so ausgeprägt, daß die Berechnung in dem durch eine gestrichelte Linie markierten Bereich abgebrochen wurde. Die zugehörigen asymmetrischen Lösungen sind in Bild 10.5 dargestellt. Interessanterweise ist der asymmetrische Sekundärpfad dieses Balkenproblems bis zum asymmetrischen Durchschlagspunkt stabil. Die FE-Diskretisierung erfolgte mit 40 EE-Stab Balkenelementen (Routine EEST in INDPRO3 und INDPRO4) [19,38]. Testrechnungen mit dem oben



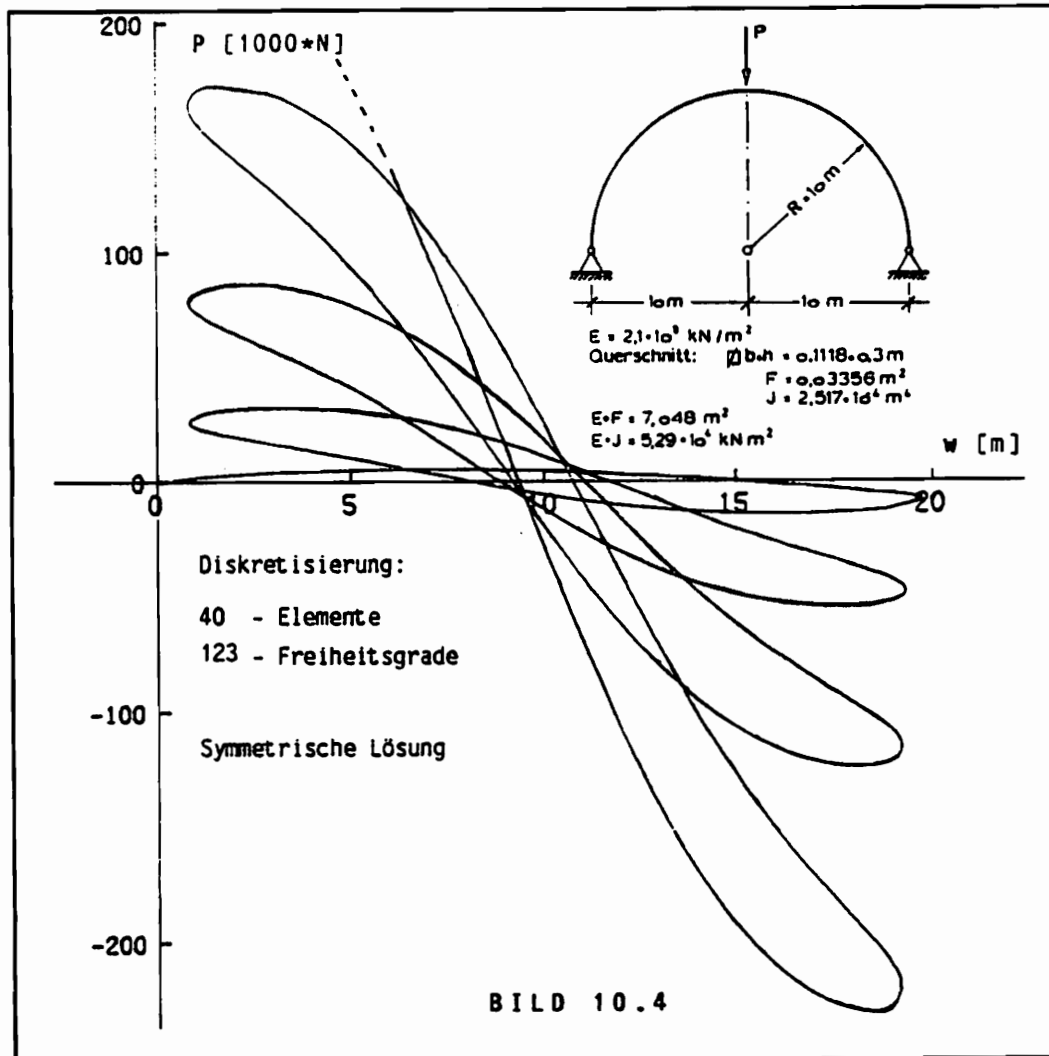
genannten "High Precision" Balkenelement wurden durchgeführt. Wie den Bildern 10.2b,c,d und 10.5b zu entnehmen ist, spielt der aktuelle Steifigkeitsparameter (s. Abschnitt 4.4) nicht nur für die Überwindung lokaler Extrema bei gleichzeitiger Unterdrückung der Gleichgewichtsiteration sondern für die vollständige Kontrolle der Strukturantworten eine bedeutende Rolle. Man beachte besonders Punkte der Last-Verschiebungskurven mit horizontalen und vertikalen Tangenten, d.h. verschwindender bzw. unendlicher Steifigkeit der Struktur. Tafel 10.4 zeigt schließlich die Leistungsfähigkeit der von den Autoren vorge-



schlagenen modifizierten konstanten Bogenlängenmethode mit Aktualisierung der Orthogonalitätsbedingung gemäß Bild 4.5c zur Überwindung relativer Extrema (s. auch Abschn. 4.3). Die dort gelisteten Rechenergebnisse gelten für den in Bild 10.2 dargestellten Kreisbogen.

Im folgenden werden wir uns mit der programmtechnischen Realisierung der Balkenberechnungen gemäß Bild 10.1-5 befassen. Dabei konzentrieren wir uns auf die Routinen des in dieser Arbeit vorgestellten Programmpakets. Im Hinblick auf die Erläuterung der verwendeten Standard-MESY3-Routinen, z.B. BANDMT, BANDBR, SYSDIM, MAXELD, ... sei auf [19] verwiesen. Bei Unklarheiten sollte der Nutzer auf das alphabetische Routinen-Listing im Anhang C mit Bezug zur Algorithmen- und Routinenbeschreibung in den Kapiteln 1-9 zurückgreifen.

INDPRO1 ist das Individualprogramm zur Berechnung des in Bild 10.2 dargestellten Bogens ($R=100$.) mit Verwendung des Steuer-Modus LAVUSW



(STEUPA) bei gleichzeitiger Kontrolle über den aktuellen Steifigkeitsparameter (AKSTPR) und pseudo-vollautomatischer Steuerung (AENSCHR). Es wird sowohl ein Konvergenzkriterium für die Verschiebungszuwächse als auch für die Ungleichgewichtskräfte mit entsprechender Wichtung benutzt (VZUWA, UNGLKA, WUNGLK). Die Aufrufe von KOBSEL und ARCHPA sind nur für den im INDPRO2 beschriebenen Restart (RSTART) notwendig. KOBSEL dient dabei der Speicherplatzvorbelegung für den dort verwendeten Steuer-Modus Interaktiv. Die Stellung und Wirkung der Routinen SYMELM, NOTHEM, KAWERB in Verbindung mit AWPZKL, EMLAST in Verbindung mit IRPEAD, TOTLAST, SICHERN und EXIST ist den zugehörigen Beschreibungen direkt zu entnehmen. Die in Kommentarzeilen eingefügten Routinen LAAEPA, SNEWRA und MNEWRA sind als mögliche Alternativen bzw. Erweiterungen zu betrachten. Nach Aufruf der Routine TOTLAST kann der LAST-Aufruf außerhalb der Iterationsschleife erfolgen, z.B. in SKT3.

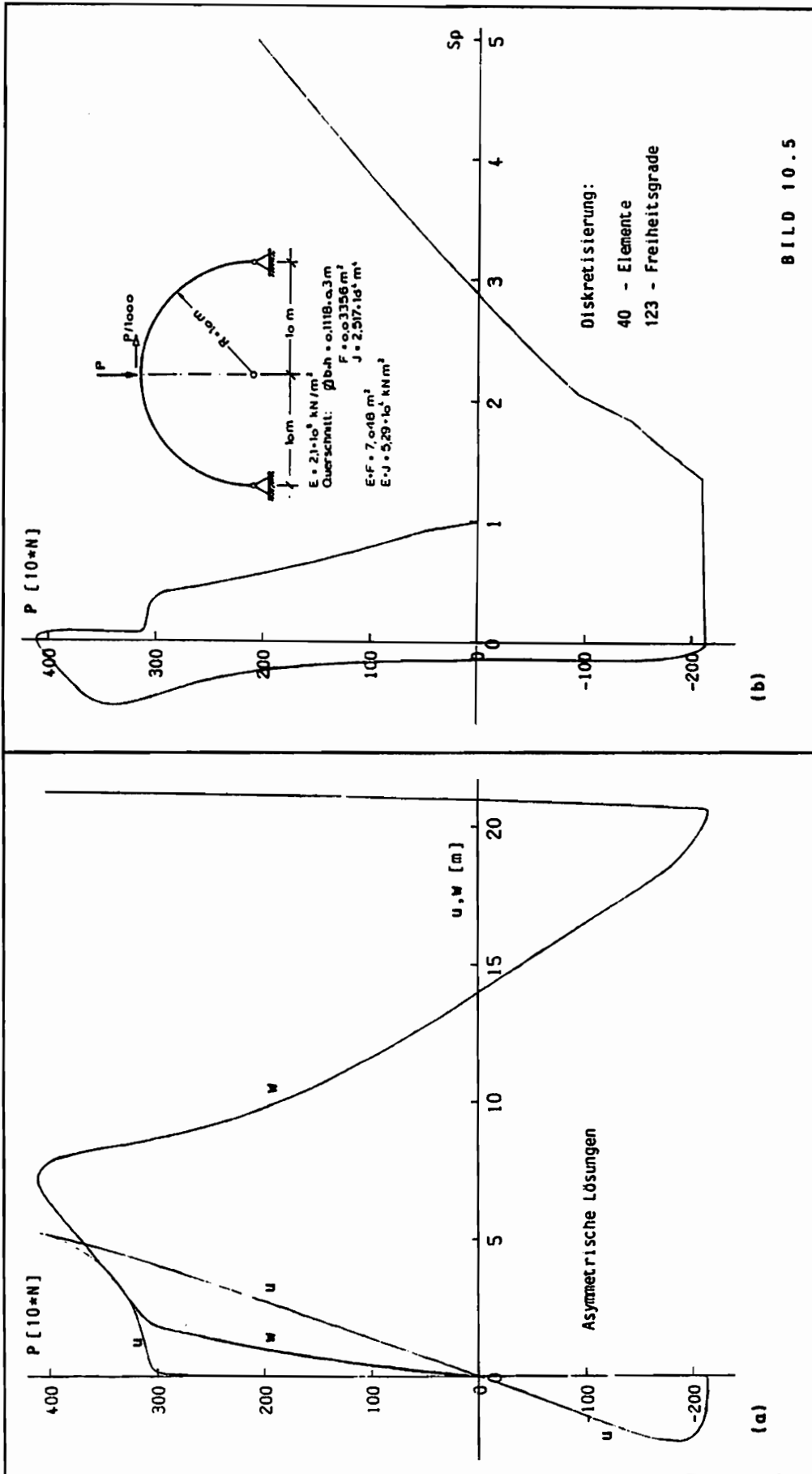


BILD 10.5

KOBLE mit MNEIRA gemäß Bild 4.5b										KOBLE mit MNEIRA gemäß Bild 4.5c				
Lösungsschritt	Lastschritt	Bogenlängenzinkrement	Anzahl der Iterationen	Lastfaktorinkrement	Δw [m]	Lastfaktor	Verschiebung w [m]	Lastschritt	Bogenlängenzinkrement	Anzahl der Iterationen	Lastfaktorinkrement	Verschiebungszinkrement Δw [m]	Lastfaktor	Verschiebung w [m]
1	1	10.	5	9.99	0.247	9.988	0.247	1	10.	5	9.99	0.247	9.986	0.247
2	2	10.	5	9.98	0.282	19.972	0.530	2	10.	5	9.98	0.282	19.969	0.530
3	3	10.	5	9.98	0.328	29.951	0.857	3	10.	5	9.98	0.328	29.944	0.857
4	4	10.	6	9.97	0.389	39.922	1.247	4	10.	6	9.97	0.389	39.901	1.246
5	5	10.	7	9.96	0.477	49.879	1.724	5	10.	7	9.95	0.477	49.857	1.723
6	6	10.	8	9.93	0.613	59.810	2.337	6	10.	8	9.91	0.611	59.768	2.334
7	7	10.	11	9.88	0.851	69.686	3.188	7	10.	11	9.82	0.844	69.593	3.178
8	8	10.	25	9.72	1.419	keine Konvergenz		8	10.	25	9.50	1.386	keine Konvergenz	
9	8	5.	9	4.86	0.592	74.546	3.780	8	5.	8	4.83	0.584	74.419	3.762
10	9	5.	13	4.74	0.830	79.283	4.610	9	5.	11	4.65	0.801	79.069	4.563
11	10	5.	25	4.52	0.851	keine Konvergenz		10	5.	25	3.99	1.326	keine Konvergenz	
12	10	2.5	10	2.18	0.590	81.462	5.200	10	2.5	9	2.14	0.555	81.211	5.118
13	11	2.5	15	1.77	0.866	83.228	6.066	11	2.5	11	1.75	0.752	82.962	5.870
14	12	2.5	4	2.88	11.431			12	2.5	12	0.625	1.030	83.587	6.904
15	12	1.25	10	7.28	22.169			13	2.5	10	-1.07	1.040	82.514	7.942
16	12	0.625	5	4.80	4.917			14	2.5	10	-1.68	0.754	80.833	8.695
17	12	0.3125	3	17.81	65.204			15	2.5	8	-1.90	0.624	78.934	9.319
18	12	0.01563	8	9.03	27.960			16	2.5	6	-2.03	0.550	76.909	9.869
19	12	0.007813	10	13.75	47.312			17	2.5	6	-2.10	0.498	74.804	10.37
20	12	0.003606	4	16.34	13.334			18	2.5	6	-2.16	0.460	72.646	10.83
21	12	0.001953	8	17.84	47.695			19	2.5	5	-2.20	0.430	70.450	11.26
22	12	0.0009766	7	9.67	30.237			20	5.0	7	-4.48	0.792	65.973	12.05
23	12	0.0004883	7	19.30	74.853			21	5.0	7	-4.55	0.723	61.421	12.77
24	12	0.0002441	6	-177.37	-175.929			22	5.0	6	-4.60	0.670	56.817	13.44
25	12	0.0001221	2	2.82	6.354			23	5.0	5	-4.64	0.628	52.176	14.07
26								24	10.0	9	-9.37	1.160	42.805	15.23

Tafel 10.4

```
PROGRAM INDPRO1
C
C   BEISPIEL 1-C-1 / JCHN085
C
C   BOGEN UNTER EINZELLAST / R = 100.
C
PARAMETER(MAXC=2100)
PARAMETER(DELV=-2.0)
PARAMETER(DELP=+20.0)
PARAMETER(NZYK=177)
PARAMETER(IIII=20)
C
EXTERNAL SYSTV,SYM,HALTEN,GAUSS,LAVUSW,VLAUSW
COMMON LC(300),IC(300),CC(MAXC)
INTEGER TYPEL,THEORY,PRESS
LOGICAL LC,EXISTC
REAL L,KRAFT,NX,LAM
DIMENSION HX(3),NX(12,3),X(3)
C
DATA E1 , H1 , R , L , Q1 , P1 , KRAFT , NE
: / 1.OE7 , 3.4641016 , 100. , 5.235987756 , 0. , 0. , 1000. , 20/
DATA TYPEL,THEORY,PRESS
: / 0, 1, 0 /
CU = 1.0/R
III = IIII
I = 0
CALL HPBINT(X,NX,HX)
OPEN(5,FILE='BILD1')
OPEN(7,FILE='BILD2')
C
CSKT1 MESY - PROGRAMMANGABEN *****
C
CALL STATIK(SYSTV)
CALL BANDMT(SYM)
CALL BANDBR(7)
CALL SYSDIM(4*(NE+1),NE)
CALL MAXELD(8)
CALL MAXNKO(8)
CALL MAXCOM(MAXC)
C-----CALL LAAEPA-----
CALL SYMELM
CALL NOTHEM
CALL KOBSEL( 4 )
C
100 CALL STEUPA(LAVUSW,DELV,DELP,NZYK,25,NE*2+3,.TRUE.)
CALL KAWERB
CALL EMLAST
CALL TOTLAST
CALL AKSTPR
CALL VZUWA(.0001 ,.00001 ,.0001, 4)
CALL UNGLKA(.005, .005, .005, 4,.TRUE.,.TRUE.)
C-----CALL SNEWRA(2,2,1.2,.TRUE.)-----
C-----CALL MNEWRA(0,.FALSE.)-----
CALL ARCHPA('TAPE01',10)
CALL SICHERN(1.1,1.1)
```

```
3
CSKT2 PROBLEMLAUF VORBEREITEN *****
3
  CALL PROBLM(01,' BEISPIEL 1-C-1 / JCHN085')
  CALL IDIAG
  CALL WUNGLK(2,0.1)
  CALL WUNGLK(4,0.1)
  WRITE(6,1000) NE,TYPEL,THEORY,PRESS
  WRITE(6,2000) E1,H1,CU,L
C
CSKT3 ANFANG PROBLEMLAUF *****
C
  CALL LAST(NE*2+3,-KRAFT)
  DO 799 I = 1,NZYK
  CALL AWPZKL(HALTEN,*400)
C
  CALL EINGSV(1,0.)
  CALL EINGSV(3,0.)
  CALL EINGSV(NE*4+1,0.)
  CALL EINGSV(NE*4+3,0.)
C
CSKT4 SYSTEMEINSTELLUNGEN *****
C
  400 CALL EXIST(EXISTC,*600)
C
CSKT5 ELEMENTSEKTIONEN *****
C
  DO 599 II = 1,NE
  CALL EL
  I1 = 4*(II-1) + 1
  DO 510 J = 1,8
510 CALL KO(J,I1+J-1,1.0)
  CALL LRV
  NCE = IRCEAD(8)
  NVO = IRVOAD( )
  NQE = IRPEAD( )
  NRFV = IRSOAD( )
  LAM = SLAMB( )
  CALL HPBMAK(TYPEL,I1,E1,H1,H1,L,CU,HX,NX,X
:           ,P1,P1,Q1,Q1,THEORY,PRESS,EXISTC,LAM
:           ,CC(NCE),CC(NQE),CC(NRFV),CC(NVO) )
599 CALL SYGL
C
CSKT6 SYSTEMGLEICHUNGEN *****
C
  600 CALL LOESE(GAUSS,*400,*650)
C
  650 CALL AENSCHR(ABS(DELV),ABS(DELP),.0001,5,.FALSE.,*799)
C
CSKT7 ERGEBNISSE NACHBEREITEN *****
C
  DET = DETN( )
  NDGL = NEGDGL( )
  AKSTP = AKST( )
  NN = NZKL( )
  PS = SLAMB( )
  IF( KRAFT.NE.O.O )
:PS = SLAMB( )*KRAFT
  MAXDV = IMAXDV( )
  DVOMAX = DVMAX( )
```

```
DVST = DVSTEU( )
IAA = IAASCH( )
IVST = IVSTEU( )
DLAMP = DLAMB( )
DK = EUNDV( )
UK1 = UKN1( )
UK2 = UKN2( )
UK3 = UKN3( )
VZ1 = VZWN1( )
VZ2 = VZWN2( )
VZ3 = VZWN3( )
C
AKSTP1 = AKSTP
IF( ABS(AKSTP).GT.6.0 ) AKSTP1 = SIGN( 6.0,AKSTP )
WRITE(5,*) PS, ', ', -XO(NE*2+3), ', ', AKSTP1
WRITE(7,*) AKSTP1, ', ', -XO(NE*2+3)
WRITE(6,3000) PS,XO(NE*2+3),DET,NDGL,NN,AKSTP
IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
GO TO 730
710 III = III + IIII
DO 720 IS = 1,NE+1
LS = (IS-1)*4 + 1
720 WRITE(6,4000) IS,XO(LS),XO(LS+1),XO(LS+2),XO(LS+3)
730 WRITE(6,5000) UK1,UK2,UK3,DLAMP,IAA,DK
: ,VZ1,VZ2,VZ3,DVST,IVST
C
799 CONTINUE
C
CSKT8 ENDE INDIVIDUELLES PROGRAMM *****
C
801 CALL ENDE
STOP
C
1000 FORMAT(10X,'NE=',I2,' TYP1=',I2,' THEORY=',I2,' PRESS=',I2//
: 10X,'BEISPIEL 1-C-1 / BOGEN UNTER EINZELLAST / R = 100.'//)
2000 FORMAT(10X,'E-MODUL=',E12.7,3X,'HOEHE=',E12.7//
: 10X,'KRUEMMUNG=',E12.7,3X,'EL-LAENGE=',E12.7///)
3000 FORMAT( 1X,'--- P=',E12.6,' W=',E10.4,' DET=',E10.4
: , ' ND=',I2,' IT=',I2,' AS=',E10.4 )
4000 FORMAT( 1X,'NR=',I3,' ',4(E12.4,' '))
5000 FORMAT( 1X,'UK1=',E10.4,' UK2=',E10.4,' UK3=',E10.4
: , ' DP=',E9.3,' I=',I3,' KB=',E9.3/1X
: , ' VZ1=',E10.4,' VZ2=',E10.4,' VZ3=',E10.4
: , ' DV=',E9.3 , ' ISTU(',I2,')' )
END
```

Anzumerken ist, daß in SKT5 über verschiedene Verwaltungsfunktionen, z.B. IRCEAD, IRVOAD, ... direkt Speicherplätze adressiert werden, wodurch ein unnötiger Datentransfer von Individualspeicherplätzen auf MESY-Speicherplätze eingespart wird. In SKT7 werden einige der in Kapitel 8 beschriebenen Funktionen zur Bestimmung charakteristischer Systemgrößen für eine komfortable Ergebnisaufbereitung aufgerufen. Man beachte, daß die gegenüber der MESY3-Quelle [19] modifizierte LOESE-Routine einen zusätzlichen Label auf Parameterposition aufweist. (s. Abschnitt 9.3).

INDPRO2 unterscheidet sich von INDPRO1 zunächst durch den RSTART-Aufruf, womit eine z.B. durch Konvergenzprobleme abgebrochene Rechnung wieder aufgenommen werden kann, hier im Modus Interaktiv unter Verwendung des Standard-Newton-Raphson-Verfahrens mit entsprechenden Spezifikationen (INTERA, INTSTU, SNEWRA). In SKT5 wird im Gegensatz zum INDPRO1 nur die Verwaltungsfunktion IRVOAD zur Adressierung der Elementrandverschiebungen eingesetzt. Daher ist ein vollständiger Datentransfer von Individualspeicherplätzen auf MESY-Speicherplätze für die Elementroutinen notwendig. Für den weniger erfahrenen MESY-Nutzer ist dies jedoch die einfachere Möglichkeit der Implementierung eigener vollparameterversorgter Elementroutinen. Der Aufruf EMLAST in SKT1 ist obligat bei TOTLAST nach RSTART und in Verbindung mit ELORGP in SKT5 zu sehen. Die Aufrufe SELWER ermöglichen im Modus Interaktiv mit Hilfe der Routine KOBSEL in INDPRO1 die Anwendung der selektiven konstanten Bogenlängenmethode. Man beachte, daß Dimensionierungsroutinen wie STATIK, PROBLEM, ... nach RSTART nicht aufgerufen werden dürfen oder durch Setzen eines Labels in RSTART übersprungen werden müssen (s. Abschnitt 6.2). Eine solche interaktive Steuerung kann in extremen Bereichen der Last-Verschiebungskurven hilfreich sein. Ein Beispiel ist der durch einen "Haken" gekennzeichnete Beginn eines "Loop" im Bild 10.2.

```
PROGRAM INDPRO2
C
C   BEISPIEL 1-C-2 / JCHN085
C
C   BOGEN UNTER EINZELLAST / R = 100.
C
PARAMETER(MAXC=2100)
PARAMETER(DELV=+0.1)
PARAMETER(DELP=-1.0)
PARAMETER(NZYK=30)
PARAMETER(IIII=10)
C
EXTERNAL SYSTV,SYM,HALTEN,GAUSS,LAVUSW,VLAUSW,KOBLME
COMMON LC(300),IC(300),CC(MAXC)
INTEGER TYPEL,THEORY,PRESS
LOGICAL LC,EXISTC
REAL L,KRAFT,NX,LAM
DIMENSION HX(3),NX(12,3),X(3),CE(36),QE(8),RFV(8),VO(8)
C
DATA E1 , H1 , R , L , Q1 , P1 , KRAFT , NE
: / 1.0E7 , 3.4641016 , 100. , 5.235987756 , 0. , 0. , 1000. , 20/
DATA TYPEL, THEORY, PRESS
: / 0, 1, 0 /
CU = 1.0/R
III = IIII
I = 0
CALL HPBINT(X,NX,HX)
OPEN(5,FILE='BILD')
```

```
C
CSKT1 MESY - PROGRAMMANGABEN *****
C
    CALL RSTART('TAPE01',70,.FALSE.,*100)
C
100 CALL STEUPA(LAVUSW,DELV,DELP,NZYK,25,NE*2+3,.TRUE.)
    CALL KAWERB
    CALL EMLAST
    CALL AKSTPR
    CALL VZUWA(.0001 ,.00001 ,.0001, 4)
    CALL UNGLKA(.005,.005,.005, 4,.TRUE.,.TRUE.)
    CALL INTERA(1,2,.FALSE.)
    CALL SNEWRA(0,2,1.2,.FALSE.)
    CALL ARCHPA('TAPE02',10)
    CALL SICHERN(2.0,5.0)
C
CSKT2 PROBLEMLAUF VORBEREITEN *****
C
    CALL IDIAG
    CALL SELWER( 1 )
    CALL SELWER( 2 )
    CALL SELWER( 4 )
    CALL WUNGLK(2,0.1)
    CALL WUNGLK(4,0.1)
    WRITE(6,1000) NE,TYP,EL,THEORY,PRESS
    WRITE(6,2000) E1,H1,CU,L
C
CSKT3 ANFANG PROBLEMLAUF *****
C
301 I = I + 1
    CALL AWPZKL(HALTEN,*400)
C
    CALL EINGSV(1,0.)
    CALL EINGSV(3,0.)
    CALL EINGSV(NE*4+1,0.)
    CALL EINGSV(NE*4+3,0.)
C
CSKT4 SYSTEMEINSTELLUNGEN *****
C
400 CALL EXIST(EXISTC,*600)
    CALL LAST(NE*2+3,-KRAFT)
C
CSKT5 ELEMENTSEKTIONEN *****
C
    DO 599 II = 1,NE
    CALL EL
    I1 = 4*(II-1) + 1
    DO 510 J = 1,8
510 CALL KO(J,I1+J-1,1.0)
    CALL LRV
    JJ = IRVOAD( )
    DO 520 J = 1,8
    VO(J) = CC(JJ)
520 JJ = JJ + 1
    LAM = SLAMB( )
    CALL HPMMAK(TYP,EL,I1,E1,H1,H1,L,CU,HX,NX,X,P1,P1,Q1,Q1
:           ,THEORY,PRESS,EXISTC,LAM,CE,QE,RFV,VO )
```

```
      IF( .NOT.EXISTC )
:CALL ELORG (CE,8,CE,0,CE,0)
      IF( TYPEL.GT.0 )
:CALL ELORGP(QE,8)
      DO 580 J = 1,8
580 CALL ASL(J,RFV(J))
599 CALL SYGL
C
CSKT6 SYSTEMGLEICHUNGEN *****
C
      600 CALL LOESE(GAUSS,*400,*799)
C
CSKT7 ERGEBNISSE NACHBEREITEN *****
C
      DET      = DETN( )
      NDGL     = NEGDGL( )
      AKSTP    = AKST( )
      NN       = NZKL( )
      PS       = SLAMB( ) *KRAFT
      DVST     = DVSTEU( )
      IAA      = IAASCH( )
      IVST     = IVSTEU( )
      DLAMP    = DLAMB( )
      DK       = EUNDV( )
      UK1      = UKN1( )
      UK2      = UKN2( )
      UK3      = UKN3( )
      VZ1      = VZWN1( )
      VZ2      = VZWN2( )
      VZ3      = VZWN3( )
C
      AKSTP1 = AKSTP
      IF( ABS(AKSTP).GT.6.0 ) AKSTP1 = SIGN( 6.0,AKSTP )
      WRITE(5,*) PS,',','-XO(NE*2+3)',',',AKSTP1
      WRITE(6,3000) PS,XO(NE*2+3),DET,NDGL,NN,AKSTP
      IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
      GO TO 730
710 III = III + III
      DO 720 IS = 1,NE+1
      LS = (IS-1)*4 + 1
720 WRITE(6,4000) IS,XO(LS),XO(LS+1),XO(LS+2),XO(LS+3)
730 WRITE(6,5000) UK1,UK2,UK3,DLAMP,IAA,DK
      :           ,VZ1,VZ2,VZ3,DVST,IVST
C
CSKT7A INTERAKTIVE STEUERUNG *****
C
      799 CALL INTSTU(*301,*801)
C
CSKT8 ENDE INDIVIDUELLES PROGRAMM *****
C
      801 CALL ENDE
      STOP
C
1000 FORMAT(10X,'NE=',I2,' TYPEL=',I2,' THEORY=',I2,' PRESS=',I2//
:          10X,'BEISPIEL 1-C-2 / BOGEN UNTER EINZELLAST / R = 100. '//)
2000 FORMAT(10X,'E-MODUL=',E12.7,3X,'HOEHE=',E12.7//
:          10X,'KRUEMMUNG=',E12.7,3X,'EL-LAENGE=',E12.7///)
```

```
3000 FORMAT( 1X, '--- P=',E12.6, ' W=',E10.4, ' DET=',E10.4
:           , ' ND=',I2, ' IT=',I2, ' AS=',E10.4
4000 FORMAT( 1X, 'NR=',I3, ' ',4(E12.4, ' '))
5000 FORMAT( 1X, 'UK1=',E10.4, ' UK2=',E10.4, ' UK3=',E10.4
:           , ' DP=',E9.3, ' I=',I3, ' KB=',E9.3/1X
:           , 'VZ1=',E10.4, ' VZ2=',E10.4, ' VZ3=',E10.4
:           , ' DV=',E9.3 , ' ISTU(',I2,')'
END
```

INDPRO3 ist das Individualprogramm zur Berechnung der symmetrischen Strukturantworten des in Bild 10.4 dargestellten Halbkreisbogens im Job-Betrieb an der CDC Cyber 855. Als Besonderheit sind hier in SKT5 die Element- und Kinematikroutinen für den EE-Stab EEST, EEKIN sowie EECS [19] zu nennen. Man beachte, daß durch Weglassen des Aufrufs RSTART und Aktivierung des Aufrufs PROBLM der normale Beginn einer Problemanalyse im Job-Betrieb erreicht wird.

```
/JOB
jobname.
/USER,.....,-----
/NOSEQ
GET,TAPE01.
SKIP,MARKE.
EXIT.
ENDIF,MARKE.
BEGIN,MESY3.
REPLACE,TAOPE02.
/EOR
PROGRAM INDPRO3
C
C BEISPIEL 1-D-1 / JCHN085
C
C HALBKREISBOGEN - BRINK
C
PARAMETER(MAXC=2500)
PARAMETER(DELV=+5.0)
PARAMETER(DELP=+5.0)
PARAMETER(NZYK=100)
PARAMETER(IIII=25)
C
EXTERNAL SYSTV,SYM,HALTEN,GAUSS,KOBLME
COMMON LC(300),IC(300),CC(MAXC)
LOGICAL LC,EXISTC
REAL J,LX,LY,KRAFT,LAM
C
DATA E , J , R , F , KRAFT , NE
: / 2.1E8 , 2.517E-4 , 10. , .03356 , 1000. , 40/
III = IIII
I = 0
OPEN(5,FILE='BILD')
```


CSKT1 MESY - PROGRAMMANGABEN *****

C

CALL RSTART('TAPE01',100,.FALSE.,*100)
CALL STATIK(SYSTV)
CALL BANDMT(SYM)
CALL BANDBR(5)
CALL SYSDIM(3*(NE+1),NE)
CALL MAXELD(6)
CALL MAXNKO(18)
CALL MAXCOM(MAXC)
CALL NOTHEM
CALL KOBSEL(3)

C

100 CALL STEUPA(KOBLME,DELV,DELP,NZYK,10,NE*3/2+1,.TRUE.)
CALL KAWERB
CALL TOTLAST
CALL AKSTPR
CALL VZUWA(.001 ,.0001 ,.001,4)
CALL UNGLKA(.01, .01, .05, 4,.TRUE.,.TRUE.)
C-----CALL INTERA(1,2,.FALSE.)-----
C-----CALL SNEWRA(0,2,1.2,.TRUE.)-----
C-----CALL MNEWRA(0,.FALSE.)-----
CALL ARCHPA('TAPE02',NZYK-1)
CALL SICHERN(1.5,0.5)

C

CSKT2 PROBLEMLAUF VORBEREITEN *****

C

C-----CALL PROBLM(01,' BEISPIEL 1-D-1 / JCHN085')-----
CALL IDIAG
DW = PI()/NE
CALL SELWER(3)
CALL WUNGLK(3,0.1)

C

CSKT3 ANFANG PROBLEMLAUF *****

C

CALL LAST(NE*3/2+1,-KRAFT)
DO 799 I = 1,NZYK
CALL AWPZKL(HALTEN,*400)

C

CALL EINGSV(1,0.)
CALL EINGSV(2,0.)
CALL EINGSV(NE*3+1,0.)
CALL EINGSV(NE*3+2,0.)

C

CSKT4 SYSTEMEINSTELLUNGEN *****

C

400 CALL EXIST(EXISTC,*600)

C

CSKT5 ELEMENTSEKTIONEN *****

C

DO 599 JE = 1,NE
CALL EL
W1 = DW*(JE-1)
W2 = W1 + DW
DX = R*(SIN(W2)-SIN(W1))
DY = R*(COS(W1)-COS(W2))
I1 = 3*JE - 2
I2 = 3*JE + 1

```
CALL EEST( E, J, F, DX, DY)
CALL EEKIN( I1, I2, DX, DY)
CALL RSL
IF( .NOT.EXISTC )
:CALL EECS( I1, I2, DX, DY)
599 CALL SYGL
C
CSKT6 SYSTEMGLEICHUNGEN *****
C
600 CALL LOESE(GAUSS,*400,*799)
C
CSKT7 ERGEBNISSE NACHBEREITEN *****
C
DET = DETN( )
NDGL = NEGDGL( )
AKSTP = AKST( )
NN = NZKL( )
PS = SLAMB( ) * KRAFT
DVST = DVSTEU( )
IAA = IAASCH( )
IVST = IVSTEU( )
DLAMP = DLAMB( )
DK = EUNDV( )
UK1 = UKN1( )
UK2 = UKN2( )
UK3 = UKN3( )
VZ1 = VZWN1( )
VZ2 = VZWN2( )
VZ3 = VZWN3( )
C
AKSTP1 = AKSTP
IF( ABS(AKSTP).GT.5.0 ) AKSTP1 = SIGN( 5.0,AKSTP )
WRITE(5,*) PS, ', ', -XO(NE*3/2+1), ', ', XO(NE*3/2+2), ', ', AKSTP1
WRITE(6,1000) PS,XO(NE*3/2+1),XO(NE*3/2+2),DET,NDGL,NN,AKSTP
IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
GO TO 730
710 III = III + III
DO 720 IS = 1,NE+1
LS = (IS-1)*3 + 1
720 WRITE(6,2000) IS,XO(LS),XO(LS+1),XO(LS+2)
730 WRITE(6,3000) UK1,UK2,UK3,DLAMP,IAA,DK
: ,VZ1,VZ2,VZ3,DVST,IVST
C
799 CONTINUE
C
CSKT8 ENDE INDIVIDUELLES PROGRAMM *****
C
801 CALL ENDE
STOP
C
1000 FORMAT( 1X,'--P',E12.6,' W',E10.4,' U',E10.4,' DET',E10.4
: , ' ND',I2,' IT',I2,' AS',E10.4 )
2000 FORMAT( 1X,'NR=',I3,' ',3(E13.5,' '))
3000 FORMAT( 1X,'UK1=',E10.4,' UK2=',E10.4,' UK3=',E10.4
: , ' DP=',E9.3,' I=',I3,' KB=',E9.3/1X
: , ' VZ1=',E10.4,' VZ2=',E10.4,' VZ3=',E10.4
: , ' DV=',E9.3 , ' ISTU(',I2,')' )
END
```

Im Gegensatz dazu zeigt INDPRO4 die klassische Restart-Bearbeitung ohne Aufruf von Dimensionierungsroutinen für das gleiche Problem. Jedoch wird eine pseudo-vollautomatische Steuerung gemäß Kapitel 9 (AENSCHR) gewählt.

```
PROGRAM INDPRO4
C
C   BEISPIEL 1-D-2 / JCHN085
C
C   HALBKREISBOGEN - BRINK
C
C   PARAMETER(MAXC=2500)
C   PARAMETER(DELV=+5.0)
C   PARAMETER(DELP=+5.0)
C   PARAMETER(NZYK=100)
C   PARAMETER(IIII=10)
C
C   EXTERNAL SYSTV,SYM,HALTEN,GAUSS,KOBLME
C   COMMON LC(300),IC(300),CC(MAXC)
C   LOGICAL LC,EXISTC
C   REAL J,LX,LY,KRAFT,LAM
C
C   DATA E , J , R , F , KRAFT , NE
: / 2.1E8 , 2.517E-4 , 10. , .03356 , 1000. , 40/
C   III = IIII
C   I = 0
C   OPEN(5,FILE='BILD')
C
C   CSKT1 MESY - PROGRAMMANGABEN *****
C
C   CALL RSTART('TAPE02',100,.FALSE.,*100)
C
C   100 CALL STEUPA(KOBLME,DELV,DELP,NZYK,10,NE*3/2+1,.TRUE.)
C   CALL KAWERB
C   CALL TOTLAST
C   CALL AKSTPR
C   CALL VZUWA(.001 ,.0001 ,.001,4)
C   CALL UNGLKA(.01, .01, .05, 4,.TRUE.,.TRUE.)
C   CALL SNEWRA(0,2,1.2,.FALSE.)
C   CALL SICHERN(1.5,0.5)
C
C   CSKT2 PROBLEMLAUF VORBEREITEN *****
C
C   CALL IDIAG
C   DW = PI( )/NE
C   CALL SELWER( 3 )
C   CALL WUNGLK(3,0.1)
C
C   CSKT3 ANFANG PROBLEMLAUF *****
C
C   CALL LAST(NE*3/2+1,-KRAFT)
C   DO 799 I = 1,NZYK
C   CALL AWPZKL(HALTEN,*400)
C
C   CALL EINGSV(1,0.)
C   CALL EINGSV(2,0.)
C   CALL EINGSV(NE*3+1,0.)
C   CALL EINGSV(NE*3+2,0.)
```

```
C
CSKT4 SYSTEMEINSTELLUNGEN *****
C
  400 CALL EXIST(EXISTC,*600)
C
CSKT5 ELEMENTSEKTIONEN *****
C
  DO 599 JE = 1,NE
  CALL EL
  W1 = DW*(JE-1)
  W2 = W1 + DW
  DX = R*(SIN(W2)-SIN(W1))
  DY = R*(COS(W1)-COS(W2))
  I1 = 3*JE - 2
  I2 = 3*JE + 1
  CALL EEST( E, J, F, DX, DY)
  CALL EEKIN( I1, I2, DX, DY)
  CALL RSL
  IF( .NOT.EXISTC )
:CALL EECS( I1, I2, DX, DY)
  599 CALL SYGL
C
CSKT6 SYSTEMGLEICHUNGEN *****
C
  600 CALL LOESE(GAUSS,*400,*650)
C
  650 CALL AENSCHR(ABS(DELV),ABS(DELP),.0001,3,.TRUE.,*799)
C
CSKT7 ERGEBNISSE NACHBEREITEN *****
C
  DET      = DETN( )
  NDGL     = NEGDGL( )
  AKSTP    = AKST( )
  NN       = NZKL( )
  PS       = SLAMB( ) *KRAFT
  DVST     = DVSTEU( )
  IAA      = IAASCH( )
  IVST     = IVSTEU( )
  DLAMP    = DLAMB( )
  DK       = EUNDV( )
  UK1      = UKN1( )
  UK2      = UKN2( )
  UK3      = UKN3( )
  VZ1      = VZWN1( )
  VZ2      = VZWN2( )
  VZ3      = VZWN3( )
C
  AKSTP1 = AKSTP
  IF( ABS(AKSTP).GT.5.0 ) AKSTP1 = SIGN( 5.0,AKSTP )
  WRITE(5,*) PS, ', ', -XO(NE*3/2+1), ', ', AKSTP1
  WRITE(6,1000) PS,XO(NE*3/2+1),DET,NDGL,NN,AKSTP
  IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
  GO TO 730
710 III = III + III
  DO 720 IS = 1,NE+1
  LS = (IS-1)*3 + 1
720 WRITE(6,2000) IS,XO(LS),XO(LS+1),XO(LS+2)
730 WRITE(6,3000) UK1,UK2,UK3,DLAMP,IAA,DK
:      ,VZ1,VZ2,VZ3,DVST,IVST
```

```
C
799 CONTINUE
C
CSKT8 ENDE INDIVIDUELLES PROGRAMM *****
C
801 CALL ENDE
STOP
C
1000 FORMAT( 1X,'--P',E12.6,' W',E10.4,' DET',E10.4
:          , ' ND',I2,' IT',I2,' AS',E10.4      )
2000 FORMAT( 1X,'NR=',I3,' ',3(E13.5,' '))
3000 FORMAT( 1X,'UK1=',E10.4,' UK2=',E10.4,' UK3=',E10.4
:          , ' DP=',E9.3,' I=',I3,' KB=',E9.3/1X
:          , ' VZ1=',E10.4,' VZ2=',E10.4,' VZ3=',E10.4
:          , ' DV=',E9.3 , ' ISTU(',I2,')'      )
END
```

INDPRO5 liefert schließlich die in Bild 10.5 dargestellten asymmetrischen Lösungen für den Halbkreisbogen unter Einzellast im Modus Interaktiv. Durch den zweiten LAST-Aufruf in SKT4 (hier kein TOTLAST-Aufruf) und die vorausgesetzte IF-Abfrage wird bereichsgesteuert eine horizontal wirkende Lastimperfection aufgebracht. Sie ermöglicht den rechnerischen Übergang von Primär- zum Sekundärpfad der Last-Verschiebungsbeziehungen.

```
PROGRAM INDPRO5
C
C BEISPIEL 1-D-3 / JCHN085
C
C HALBKREISBOGEN - BRINK
C
PARAMETER(MAXC=2500)
PARAMETER(DELV=+5.0)
PARAMETER(DELP=+5.0)
PARAMETER(NZYK=10)
PARAMETER(IIII=20)
C
EXTERNAL SYSTV,SYM,HALTEN,GAUSS,LAVUSW,VLAUSW,KOBLME
COMMON LC(300),IC(300),CC(MAXC)
LOGICAL LC,EXISTC
REAL J,LX,LY,KRAFT,LAM
C
DATA E , J , R , F , KRAFT , NE
: / 2.1E8 , 2.517E-4 , 10. , .03356 , 100.0 , 40/
III = IIII
I = 0
ALFA = 0.001
OPEN(5,FILE='BILD')
C
CSKT1 MESY - PROGRAMMANGABEN *****
C
CALL STATIK(SYSTV)
CALL BANDMT(SYM)
CALL BANDBR(5)
CALL SYSDIM(3*(NE+1),NE)
```

```
CALL MAXELD(6)
CALL MAXNKO(18)
CALL MAXCOM(MAXC)
CALL NOTHEM
CALL KOBSEL( 3 )
C
100 CALL STEUPA(KOBLME,DELV,DELP,NZYK,10,NE*3/2+1,.TRUE.)
CALL KAWERB
CALL AKSTPR
CALL VZUWA(.001 ,.0001 ,.001,4)
CALL UNGLKA(.01, .01, .05, 4,.TRUE.,.TRUE.)
CALL INTERA(1,2,.FALSE.)
CALL MNEWRA(0,.FALSE.)
CALL SICHERN(2.0,0.75)
C
CSKT2 PROBLEMLAUF VORBEREITEN *****
C
CALL PROBLM(01,' BEISPIEL 1-D-3 / JCHN085')
CALL IDIAG
DW = PI( )/NE
CALL SELWER( 3 )
CALL WUNGLK(3,0.1)
C
CSKT3 ANFANG PROBLEMLAUF *****
C
301 I = I + 1
CALL AWPZKL(HALTEN,*400)
C
CALL EINGSV(1,0.)
CALL EINGSV(2,0.)
CALL EINGSV(NE*3+1,0.)
CALL EINGSV(NE*3+2,0.)
C
CSKT4 SYSTEMEINSTELLUNGEN *****
C
IF( ABS(XO(NE*3/2+2)).GT.4.0 ) ALFA = 0.0
400 CALL EXIST(EXISTC,*600)
CALL LAST(NE*3/2+1,-KRAFT)
CALL LAST(NE*3/2+2,+KRAFT*ALFA)
C
CSKT5 ELEMENTSEKTIONEN *****
C
DO 599 JE = 1,NE
CALL EL
W1 = DW*(JE-1)
W2 = W1 + DW
DX = R*(SIN(W2)-SIN(W1))
DY = R*(COS(W1)-COS(W2))
I1 = 3*JE - 2
I2 = 3*JE + 1
CALL EEST( E, J, F, DX, DY)
CALL EEKIN( I1, I2, DX, DY)
CALL RSL
IF( .NOT.EXISTC )
:CALL EECS( I1, I2, DX, DY)
599 CALL SYGL
C
```

CSKT6 SYSTEMGLEICHUNGEN *****

C

600 CALL LCESE(GAUSS,*400,*799)

C

CSKT7 ERGEBNISSE NACHBEREITEN *****

C

```
DET      = DETN( )
NDGL     = NEGDGL( )
AKSTP    = AKST( )
NN       = NZKL( )
PS       = SLAMB( ) *KRAFT
DVST     = DVSTEU( )
IAA      = IAASCH( )
IVST     = IVSTEU( )
DLAMP    = DLAMB( )
DK       = EUNDV( )
UK1      = UKN1( )
UK2      = UKN2( )
UK3      = UKN3( )
VZ1      = VZWN1( )
VZ2      = VZWN2( )
VZ3      = VZWN3( )
```

C

```
AKSTP1 = AKSTP
IF( ABS(AKSTP).GT.5.0 ) AKSTP1 = SIGN( 5.0,AKSTP )
WRITE(5,*) PS, ', ', -XO(NE*3/2+1), ', ', XO(NE*3/2+2), ', ', AKSTP1
WRITE(6,1000) PS,XO(NE*3/2+1),XO(NE*3/2+2),DET,NDGL,NN,AKSTP
IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
```

GO TO 730

710 III = III + IIII

DO 720 IS = 1,NE+1

LS = (IS-1)*3 + 1

720 WRITE(6,2000) IS,XO(LS),XO(LS+1),XO(LS+2)

730 WRITE(6,3000) UK1,UK2,UK3,DLAMP,IAA,DK

: ,VZ1,VZ2,VZ3,DVST,IVST

C

CSKT7A INTERAKTIVE STEUERUNG *****

C

799 CALL INTSTU(*301,*801)

C

CSKT8 ENDE INDIVIDUELLES PROGRAMM *****

C

801 CALL ENDE

STOP

C

1000 FORMAT(1X, '--P',E12.6, ' W',E10.4, ' U',E10.4, ' DET',E10.4

: , ' ND',I2, ' IT',I2, ' AS',E10.4)

2000 FORMAT(1X, 'NR=',I3, ' ',3(E13.5, ' '))

3000 FORMAT(1X, 'UK1=',E10.4, ' UK2=',E10.4, ' UK3=',E10.4

: , ' DP=',E9.3, ' I=',I3, ' KB=',E9.3/1X

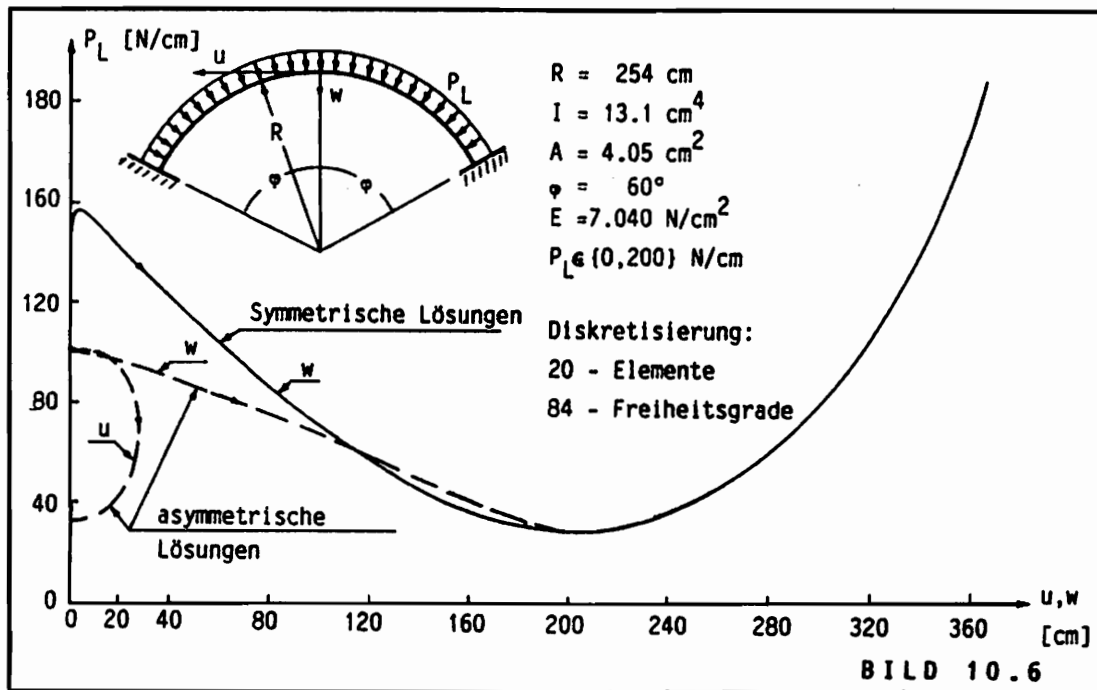
: , ' VZ1=',E10.4, ' VZ2=',E10.4, ' VZ3=',E10.4

: , ' DV=',E9.3 , ' ISTU(' ,I2, ')')

END

10.3.2 KREISBOGEN UNTER VERFORMUNGSABHÄNGIGER DRUCKLAST

Die symmetrischen und asymmetrischen Lösungen eines beidseitig eingespannten Kreisbogens unter verformungsabhängiger Drucklast sind in Bild 10.6 dargestellt. Die FE-Diskretisierung erfolgte mit 20 Balken-



elementen, die auf der Theorie finiter Rotationen gemäß Tafel 10.3 aufgebaut sind (HPBMAK). Wie in [33] nachgewiesen wurde, gilt die Existenz des dort angegebenen Drucklastpotentials nur für bestimmte geometrische Randbedingungen. Vergleichende Untersuchungen dieses Problems für verformungsabhängige und richtungstreue Drucklasten finden sich in [32, 33, 39]. Sie zeigen die Notwendigkeit für die Berücksichtigung der Verformungsabhängigkeit, beispielsweise zur korrekten Bestimmung von Beuldrücken dünner Kreisbögen. In Einzelfällen können anderenfalls Abweichungen bis zu 33% auftreten.

Wie schon in 10.3.1 erwähnt, sind im INDPRO für verformungsabhängige Elementlasten die Parameter TYPEL und THEORY auf 1 voreinzustellen. INDPRO6 zeigt eine interaktive Steuerung (INTERA, INTSTU) unter Verwendung der selektiven konstanten Bogenlängenmethode (KOBSEL, STEUPA, SELWER) und gleichzeitiger Kontrolle über den aktuellen Steifigkeitsparameter (AKSTPR). Erwähnenswert ist die durch SELWER-


```
PROGRAM INDPRO6
C
C   BEISPIEL 2-H / JCHN085
C
C   BOGEN UNTER DRUCKLAST
C
PARAMETER(MAXC=2100)
PARAMETER(DELV=+5.0000)
PARAMETER(DELP=+5.0000)
PARAMETER(NZYK=70)
PARAMETER(IIII=10)
C
EXTERNAL SYSTV,SYM,HALTEN,GAUSS,LAVUSW,VLAUSW,KOBLME
COMMON LC(300),IC(300),CC(MAXC)
INTEGER TYPEL,THEORY,PRESS
LOGICAL LC,EXISTC
REAL L,NX,LAM
DIMENSION HX(3),NX(12,3),X(3)
C
DATA          E1  ,  H1  ,  R  ,  Q1  ,  P1  ,  NE
:             / 7.040E6 , 6.23 , 254.0 , 1.538462 , 0.0 , 20 /
DATA          TYPEL, THEORY, PRESS
:             / 1, 1, 1 /
L  = (8.*ATAN(1.)/3.)*R/NE
CU  = 1.0/R
III = IIII
I   = 0
CALL HPBINT(X,NX,HX)
OPEN(5,FILE='BILD')
C
CSKT1 MESY - PROGRAMMANGABEN *****
C
CALL STATIK(SYSTV)
CALL BANDMT(SYM)
CALL BANDER(7)
CALL SYSDIM(4*(NE+1),NE)
CALL MAXELD(8)
CALL MAXNKO(8)
CALL MAXCOM(MAXC)
CALL SYMELM
CALL NOTHEM
CALL KOBSEL( 84 )
C
100 CALL STEUPA(KOBLME,DELV,DELP,NZYK,25,NE*2+3,.TRUE.)
CALL KAWERB
CALL EMLAST
CALL INTERA(1,2,.FALSE.)
CALL AKSTPR
CALL SNEWRA(0,2,1.2,.FALSE.)
CALL VZUWA(.001 ,.001 ,.001 , 4)
CALL UNGLKA(.01 ,.01 ,.01 , 4,.FALSE.,.FALSE.)
CALL ARCHPA('TAPE01',20)
CALL SICHERN(1.0,1.0)
C
CSKT2 PROBLEMLAUF VORBEREITEN *****
C
CALL PROBLM(02,' BEISPIEL 2-H / JCHN085 ')
CALL IDIAG
DO 205 I = 1,84
```

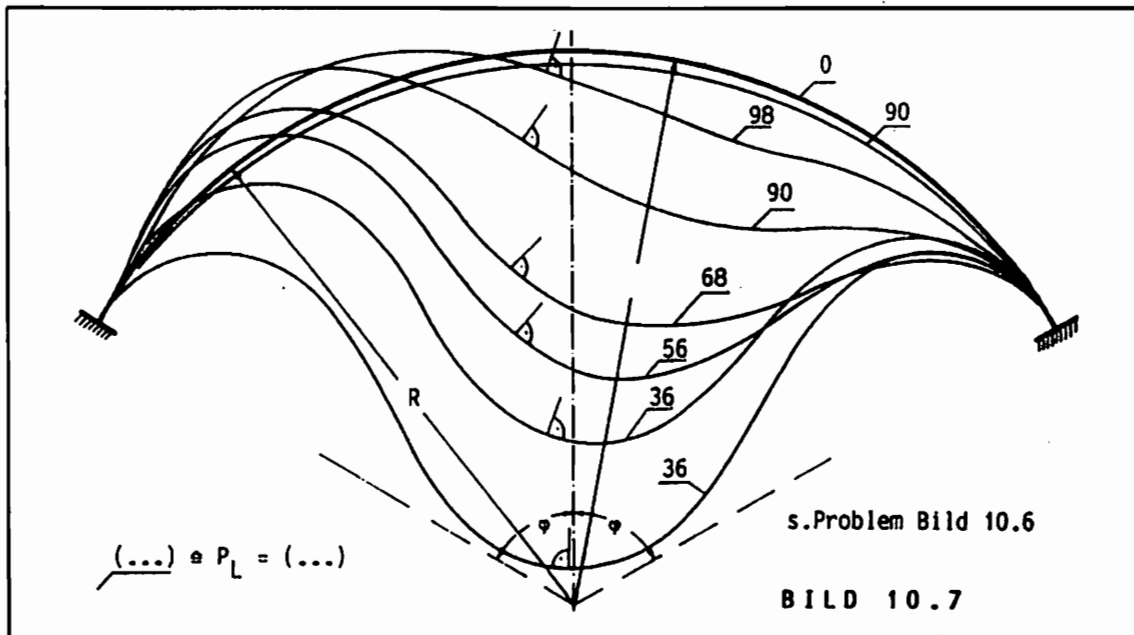
```
IF( I.EQ.45 ) GO TO 205
CALL SELWER( I )
205 CONTINUE
WRITE(6,1000) NE,TYPEL,THEORY,PRESS
WRITE(6,2000) E1,H1,CU,L
C
CSKT3 ANFANG PROBLEMLAUF *****
C
301 I = I + 1
CALL AWPZKL(HALTEN,*400)
C
CALL EINGSV(1,0.)
CALL EINGSV(3,0.)
CALL EINGSV(4,0.)
CALL EINGSV(NE*4+1,0.)
CALL EINGSV(NE*4+3,0.)
CALL EINGSV(NE*4+4,0.)
C
CSKT4 SYSTEMEINSTELLUNGEN *****
C
400 CALL EXIST(EXISTC,*600)
IF( ABS(XO(NE*2+3)).LE.20.0 )
:CALL LAST(NE*2+1,1.00)
C
CSKT5 ELEMENTSEKTIONEN *****
C
DO 599 II = 1,NE
CALL EL
I1 = 4*(II-1) + 1
DO 510 J = 1,8
510 CALL KO(J,I1+J-1,1.0)
CALL LRV
NCE = IRCEAD(8)
NVO = IRVOAD( )
NQE = IRPEAD( )
NRFV = IRSOAD( )
LAM = SLAMB( )
CALL HPBMAK(TYPEL,I1,E1,H1,H1,L,CU,HX,NX,X
: ,P1,P1,Q1,Q1,THEORY,PRESS,EXISTC,LAM
: ,CC(NCE),CC(NQE),CC(NRFV),CC(NVO) )
599 CALL SYGL
C
CSKT6 SYSTEMGLEICHUNGEN *****
C
600 CALL LOESE(GAUSS,*400,*799)
C
CSKT7 ERGEBNISSE NACHBEREITEN *****
C
DET = DETN( )
NDGL = NEGDGL( )
AKSTP = AKST( )
NN = NZKL( )
PS = SLAMB( )
MAXDV = IMAXDV( )
DVOMAX = DVMAX( )
DVST = DVSTEU( )
IAA = IAASCH( )
IVST = IVSTEU( )
DLAMP = DLAMB( )
```

```

DK      = EUNDV( )
UK1     = UKN1( )
UK2     = UKN2( )
UK3     = UKN3( )
VZ1     = VZWN1( )
VZ2     = VZWN2( )
VZ3     = VZWN3( )
C
WRITE(5,*) PS, ', ', -XO(NE*2+3), ', ', -XO(NE*2+1), ', ', AKSTP
WRITE(6,3000) PS, XO(NE*2+3), XO(NE*2+1), DET, NDGL, NN, AKSTP
IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
GO TO 730
710 III = III + IIII
DO 720 IS = 1, NE+1
LS = (IS-1)*4 + 1
720 WRITE(6,4000) IS, XO(LS), XO(LS+1), XO(LS+2), XO(LS+3)
730 WRITE(6,5000) UK1, UK2, UK3, DLAMP, IAA, DK
:      , VZ1, VZ2, VZ3, DVST, IVST
C
CSKT7A INTERAKTIVE STEUERUNG *****
C
799 CALL INTSTU(*301, *801)
C
CSKT8 ENDE INDIVIDUELLES PROGRAMM *****
C
801 CALL ENDE
STOP
C
1000 FORMAT(10X, 'NE=', I2, 3X, 'TYPEL=', I2, 3X, 'THEORY=', I2, 3X, 'PRESS=' I2//
:      10X, 'BEISPIEL 2-H / JCHN085 / BOGEN UNTER DRUCKLAST'//)
2000 FORMAT(10X, 'E-MODUL=', E12.7, 3X, 'HOEHE=', E12.7, //
:      10X, 'KRUEMMUNG=', E12.7, 3X, 'EL-LAENGE=', E12.7///)
3000 FORMAT( 1X, '-- P', E12.6, ' W', E10.4, ' U', E10.4, ' DET', E10.4
:      , ' ND', I2, ' IT', I2, ' AS', E10.4 )
4000 FORMAT( 1X, 'NR=', I3, ' ', 4(E12.4, ' '))
5000 FORMAT( 1X, 'UK1=', E10.4, ' UK2=', E10.4, ' UK3=', E10.4
:      , ' DP=', E9.3, ' I=', I3, ' KB=', E9.3/
:      1X, 'VZ1=', E10.4, ' VZ2=', E10.4, ' VZ3=', E10.4
:      , ' DV=', E9.3, ' ISTU(' I2, ' )'
END

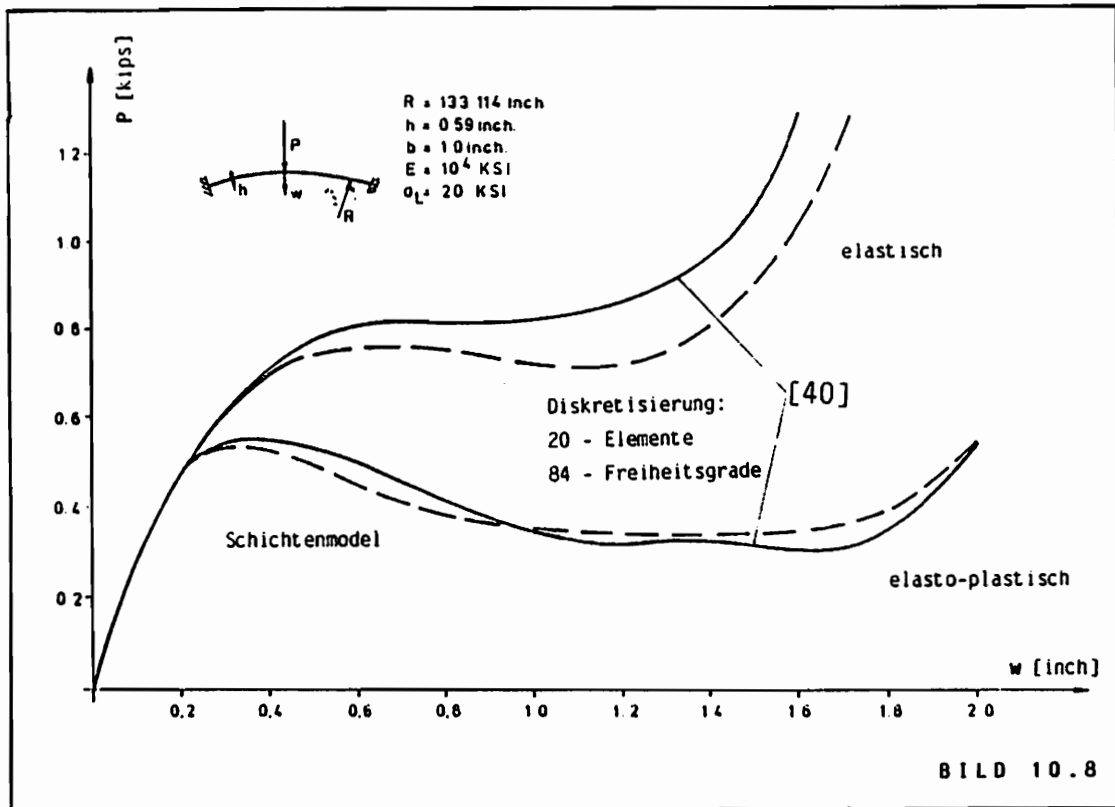
```

Aufrufe in SKT2 erzeugte degenerierte Form der Bogenlänge, die nur noch aus einer Verschiebungskomponente und dem Lastzuwachs gebildet wird und daher den graphischen Interpretationen in Bild 4.5 völlig entspricht. Im Gegensatz zu den vorangegangenen Beispielen ist dann im KOBSEL-Aufruf, der Parameter gleich der Anzahl der Systemverschiebungen zu setzen. Die in SKT4 aufgebrachte horizontale Last ist nur zur Berechnung der asymmetrischen Sekundärpfade (s. auch Deformationsmuster in Bild 10.7) einzuprägen.



10.3.3 KREISBOGEN UNTER EINZELLAST, ELASTO-PLASTISCHE ANALYSE

Bild 10.8 zeigt die elastische und elasto-plastische Berechnung eines eingespannten mäßig steilen Bogens unter Einzellast. Die FE-Diskretisierung erfolgte mit jeweils 20 Balkenelementen. Für die elasto-plastische Analyse wurde die in Tafel 10.3 gezeigte Theorie in eine inkrementelle Form überführt und über ein Schichtenmodell mit konstitutiven Gesetzen versorgt [32]. Dem entsprechend erfolgte eine spezielle Aufbereitung der Elementroutine HPBMAK. Die Routine GAUSINT sorgt für die Integration über die Dickenkoordinate des Balkens. Die eigenen Ergebnisse konnten mit den Berechnungen in [40] verglichen werden. Dabei zeigen sich für die elastische Lösung schon im Vorbeulbereich deutliche Unterschiede, die wahrscheinlich auf ein oft zu beobachtendes versteifendes Verhalten der in [40] gewählten "corotational" Elementformulierung zurückzuführen ist. In der elasto-plastischen Berechnung treten deutlich geringere Abweichungen auf. Eine physikalische Interpretation dieser Ergebnisse insbesondere im Hinblick auf den Einfluß kinematisch fehlerhafter Modelle auf die elasto-plastische Analyse ist äußerst schwierig und bedarf weiterer



Untersuchungen. INDPRO7 unterscheidet sich von den vorangegangenen INDPROi zunächst durch den Aufruf LGESCHT zur Speicherplatzreservierung für die Verformungsgeschichte. Im vorliegenden Beispiel berechnet sich der zugehörige Parameter wie folgt:

NGE = 3×7 Int. Pkt. \times 20 Elem. \times 5 Inform. pro Punkt
(s. auch INITPLA). Die Vorbelegung dieser Speicherplätze beim ersten Durchlauf erfolgt durch die Routine INITPLA. Beachte, daß nach einem möglichen Restart (RSTART) INITPLA nicht mehr aufgerufen werden darf. Mit Hilfe der Verwaltungsfunktion IRGSAD kann dann (s. SKT5) der entsprechende Speicherbereich adressiert werden.

```
PROGRAM INDPRO7
C
C BEISPIEL 3-D / JCHN085
C
C PLASTISCHES BALKENPROGRAMM MIT INTEGRATIONSMETHODE(GAUSS)
C
C
C PARAMETER(MAXC=7500)
C PARAMETER(DELV=0.05)
C PARAMETER(DELP=0.05)
C PARAMETER(NZYK=20)
C PARAMETER(IIII=5)
C
```

```
EXTERNAL SYSTV,SYM,HALTEN,GAUSS,LAVUSW,VLAUSW,KOBLME
COMMON LC(300),IC(300),CC(MAXC)
COMMON /GGGGG/ AB(10,10),WK(10,10)
INTEGER TYPEL,THEORY,PRESS
LOGICAL LC,EXISTC
REAL L,KRAFT,NX,LAM
DIMENSION HX(3),NX(12,3),X(3)
```

C

```
DATA E1 , H1 , R , L , Q1 , P1 , KRAFT , NE
: / 1.OE7 , 0.59 , 133.114 , 1.704655378 , 0. , 0. , 1000. , 20 /
DATA TYPEL,THEORY,PRESS
: / 0, 1, 0 /
DATA ETA , SIGO , INTG , NINTG , NNGE , NGE
: / 0.0 , 2.OE4 , 7 , 21 , 420 , 2100 /
CU = 1.0/R
III = IIII
I = 0
CALL HPBINT(X,NX,HX)
CALL GAUSINT(AB,WK)
OPEN(5,FILE='BILD')
```

C

CSKT1 MESY - PROGRAMMANGABEN *****

C

```
CALL STATIK(SYSTV)
CALL BANDMT(SYM)
CALL BANDBR(7)
CALL SYSDIM(4*(NE+1),NE)
CALL MAXELD(8)
CALL MAXNKO(8)
CALL MAXCOM(MAXC)
CALL LGESCHT(NGE)
CALL SYMELM
CALL NOTHEM
```

C

```
100 CALL STEUPA(KOBLME,DELV,DELP,NZYK,25,NE*2+3,.TRUE.)
CALL KAWERB
CALL EMLAST
CALL INTERA(1,2,.FALSE.)
CALL AKSTPR
CALL VZUWA(.001 ,.001 ,.001 ,4)
CALL UNGLKA(.01 ,.01 ,.01 , 4,.FALSE.,.FALSE.)
CALL ARCHPA('TAPE01',20)
CALL SICHERN(1.1,1.1)
```

C

CSKT2 PROBLEMLAUF VORBEREITEN *****

C

```
CALL PROBLM(03,' BEISPIEL 3-D / JCHON85')
CALL IDIAG
WRITE(6,1000) NE,TYPEL,THEORY,PRESS
WRITE(6,2000) E1,H1,SIGO,CU,L
```

C

CSKT3 ANFANG PROBLEMLAUF *****

C

```
NLGE = IRGSAD( )
NLGE1 = NLGE + NNGE
EPSO = SIGO/E1
CALL INITPLA(CC(NLGE),CC(NLGE1),NNGE,EPSO,-EPSO,SIGO,-SIGO)
```

```
C
301 I = I + 1
CALL AWPZKL(HALTEN,*400)
C
CALL EINGSV(1,0.)
CALL EINGSV(3,0.)
CALL EINGSV(4,0.)
CALL EINGSV(NE*4+1,0.)
CALL EINGSV(NE*4+3,0.)
CALL EINGSV(NE*4+4,0.)
C
CSKT4 SYSTEMEINSTELLUNGEN *****
C
400 CALL EXIST(EXISTC,*600)
CALL LAST(NE*2+3,-KRAFT)
C
CSKT5 ELEMENTSEKTIONEN *****
C
DO 599 II = 1,NE
CALL EL
I1 = 4*(II-1) + 1
DO 510 J = 1,8
510 CALL KO(J,I1+J-1,1.0)
CALL LRV
NCE = IRCEAD(8)
NVO = IRVOAD( )
NQE = IRPEAD( )
NRFV = IRSOAD( )
NLGE = IRGSAD( )
LG1 = NLGE + (II-1)*NINTG
LG2 = NLGE + (II-1)*NINTG*4 + NNGE
LAM = SLAMB( )
CALL HPBMAK(TYPEL,I1,E1,H1,H1,L,CU,HX,NX,X
: ,P1,P1,Q1,Q1,THEORY,PRESS,EXISTC,LAM,ETA,NINTG
: ,CC(NCE),CC(NQE),CC(NRFV),CC(NVO),CC(LG1),CC(LG2) )
599 CALL SYGL
C
CSKT6 SYSTEMGLEICHUNGEN *****
C
600 CALL LOESE(GAUSS,*400,*799)
C
CSKT7 ERGEBNISSE NACHBEREITEN *****
C
DET = DETN( )
NDGL = NEGDGL( )
AKSTP = AKST( )
NN = NZKL( )
PS = SLAMB( )
IF( KRAFT.NE.O.O )
:PS = SLAMB( )*KRAFT
MAXDV = IMAXDV( )
DVOMAX = DVMAX( )
DVST = DVSTEU( )
IAA = IAASCH( )
IVST = IVSTEU( )
DLAMP = DLAMB( )
DK = EUNDV( )
```

```
UK1    = UKN1( )
UK2    = UKN2( )
UK3    = UKN3( )
VZ1    = VZWN1( )
VZ2    = VZWN2( )
VZ3    = VZWN3( )

C
WRITE(5,*) PS,',',-XO(NE*2+3),',',-XO(NE*2+1),',',AKSTP
WRITE(6,3000) PS,XO(NE*2+3),DET,NDGL,NN,AKSTP
IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
GO TO 730
710 III = III + III
DO 720 IS = 1,NE+1
LS = (IS-1)*4 + 1
720 WRITE(6,4000) IS,XO(LS),XO(LS+1),XO(LS+2),XO(LS+3)
730 WRITE(6,5000) UK1,UK2,UK3,DLAMP,IAA,DK
      :           ,VZ1,VZ2,VZ3,DVST,IVST

C
CSKT7A INTERAKTIVE STEUERUNG *****
C
799 CALL INTSTU(*301,*801)
C
CSKT8 ENDE INDIVIDUELLES PROGRAMM *****
C
801 CALL ENDE
STOP

C
1000 FORMAT(10X,'NE=',I2,' TYPEL=',I2,' THEORY=',I2,' PRESS=',I2/
:          10X,'BEISPIEL 3-D / JCHN085 / BELYTSCHKO/'
2000 FORMAT(10X,'E-MODUL=',E12.7,3X,'HOEHE=',E12.7,3X,'SIGO=',E12.7//
:          10X,'KRUENNUNG=',E12.7,3X,'EL-LAENGE=',E12.7/// )
3000 FORMAT( 1X,'-- P=',E12.6,' W=',E10.4,' DET=',E10.4
:          :   ' ND=',I2,' IT=',I2,' AS=',E10.4           )
4000 FORMAT( 1X,'NR=',I3,' ',4(E12.4,' '))
5000 FORMAT( 1X,'UK1=',E10.4,' UK2=',E10.4,' UK3=',E10.4
:          :   ' DP=',E9.3,' I=',I3,' KB=',E9.3/
:          1X,'VZ1=',E10.4,' VZ2=',E10.4,' VZ3=',E10.4
:          :   ' DV=',E9.3,' ISTU(',I2,')'           )
END

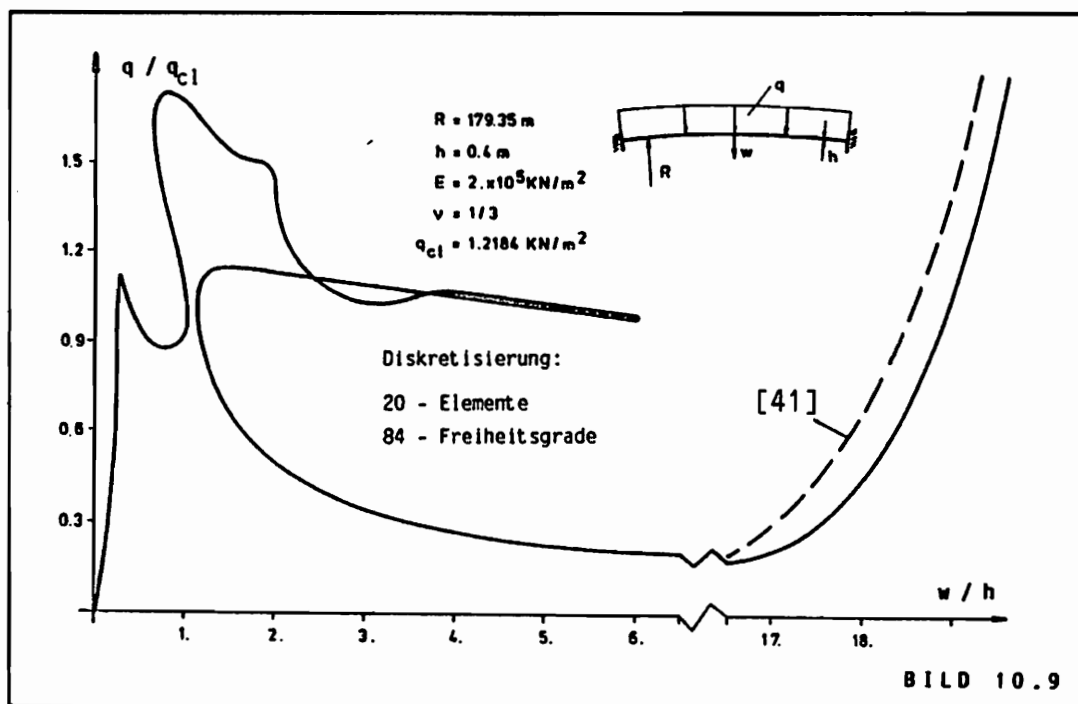
C
SUBROUTINE I N I T P L A (IPL,PLGE,NNGE,EPSO,EPSU,SIGO,SIGU)
DIMENSION IPL(NNGE),PLGE(4,NNGE)
DO 100 I = 1,NNGE
IPL (I) = 1
PLGE(1,I) = EPSO
PLGE(2,I) = EPSU
PLGE(3,I) = SIGO
100 PLGE(4,I) = SIGU
RETURN
END
```


10.4. SCHALENBEISPIELE

10.4.1. KUGELKALOTTE UNTER RADIALER FLÄCHENLAST

Eine Vielzahl numerischer und experimenteller Arbeiten haben sich mit dem Tragverhalten von Kugelkalotten unter gleichförmiger radialer Flächenlast beschäftigt. Diese Untersuchungen beschränken sich in der Regel auf die Bestimmung kritischer Lasten, d.h. Durchschlags- oder Verzweigungslasten.

Eine vollständige Lösung des axialsymmetrischen Problems einer eingespannten Kugelkalotte mit Kennparameter $\lambda = 2[3(1-\nu^2)]^{1/4}(H/h)^{1/2} = 8$ ist in Bild 10.9 dargestellt. Die FE-Diskretisierung erfolgte mit 20 Schalenringelementen, die auf der Theorie finiter Rotationen gemäß Tafel 10.2 aufgebaut sind. Die zugehörige Makro-Elementroutine in INDPROB ist HPRMAK. Sie wird durch die Problemdatei-, Geometrie-, Integrations-, und Organisationsroutinen HPRINP, HPRGEO, HPRINT und HPRORG versorgt. Systemrandbedingungen werden unabhängig vom Problemdateisatz durch entsprechende EINGSV-Aufrufe in SKT3 realisiert. Die Parameter THEORY und PRESS geben an, ob mit einer Theorie moderater Rotationen (THEORY=0) [27], der Theorie finiter Rotationen (THEORY=1), verformungsunabhängiger "Totlast" (PRESS=0) oder verformungsabhängigen



(PRESS=1) Flächenlasten gerechnet werden soll. Es ist anzumerken, daß zur Analyse einer solchen flachen Schale schon eine vereinfachte Theorie moderater Rotationen ausreichen würde [28,29]. Die eigenen Ergebnisse für den Lastfall "Totlast" konnten mit den Berechnungen in [41,42] verglichen werden. Eine linearisierte und eine vollständig nichtlineare Verzweigungsanalyse findet sich in [28]. Die Ergebnisse infolge einer verformungsabhängigen und -unabhängigen radialen Flächenlast weichen nicht voneinander ab (s. auch [33]). Die Komplexität der Last-Verschiebungsbeziehungen mit Auftreten mehrerer horizontaler und vertikaler Tangenten sowie Durchdringungen erforderte in diesem Beispiel einen leistungsstarken Steueralgorithmus.

Wie INDPRO8 zeigt, wurde die selektive konstante Bogenlängenmethode gemäß Bild 4.5a mit Kontrolle über den aktuellen Steifigkeitsparameter im Modus Interaktiv ausgewählt (KOBSEL, STEUPA, INTERA, AKSTPR, VZUWA, UNGLKA, SELWER, INTSTU). Die Notwendigkeit und/oder Wirkung zusätzlicher Routinen-Aufrufe, z.B. SYMELM, NOTHEM, KAWERB, EMLAST, SICHERN, EXIST, ... ist in den vorangegangenen numerischen Beispielen behandelt worden.

```
PROGRAM INDPRO8
C
C BEISPIEL 4 / JCHN085
C
C AXIALSYMMETRISCHE VERFORMUNGEN VON ROTATIONSSCHALEN
C
C UNTER TOT- UND DRUCKLASTEN
C
C
C PARAMETER(MAXC=2100)
C PARAMETER(DELV=+0.2000)
C PARAMETER(DELP=+0.2000)
C PARAMETER(NFVG=3)
C PARAMETER(NZYK=50)
C PARAMETER(IIII=10)
C
C EXTERNAL SYSTV,SYM,HALTEN,GAUSS,LAVUSW,VLAUSW,KOBLME
C COMMON LC(300),IC(300),CC(MAXC)
C INTEGER TFEV,OUT,TYPEL,THEORY,PRESS
C LOGICAL LC,EXISTC
C REAL LV,NX,NUE12
C DIMENSION NEV(15),TFEV(15),RV(15),GP(115),HV(115)
C : ,PL(101),QL(101),LV(100),HX(3),NX(12,3),X(3)
C : ,AV(15),PHIV(101),RIV(101),PLM(101),QLM(101)
C
C DATA THEORY / 1 / , PRESS / 1 /
C CALL HPRINP(NE,NPA,E1,E2,NUE12,G12,NEV,TFEV,RV,AV,GP,HV
C : ,OUT,TYPEL,PL,QL )
C CALL HPRGEO(NE,NPA,NEV,RV,AV,TFEV,GP,LV,PHIV,RIV)
C CALL HPRINT(X,NX,HX)
C III = IIII
C I = 0
C OPEN(5,FILE='BILD')
```

```
C
CSKT1 MESY - PROGRAMMANGABEN *****
C
  CALL STATIK(SYSTV)
  CALL BANDMT(SYM)
  CALL BANDBR(7)
  CALL SYSDIM(4*(NE+1),NE)
  CALL MAXELD(8)
  CALL MAXNKO(8)
  CALL MAXCOM(MAXC)
  CALL SYMELM
  CALL NOTHEM
  CALL KOBSEL( 4 )
100 CALL STEUPA(KOBLME,DELV,DELP,NZYK,10,NFVG,.TRUE.)
  CALL KAWERB
  CALL EMLAST
  CALL INTERA(1,2,.FALSE.)
  CALL AKSTPR
  CALL VZUWA(.001 ,.001 ,.001, 4)
  CALL UNGLKA(.01 ,.01 ,.01 , 4,.FALSE.,.FALSE.)
  CALL SICHERN(2.0,2.0)
C
CSKT2 PROBLEMLAUF VORBEREITEN *****
C
  CALL PROBLM(02,' BEISPIEL 4 / JCHN085 ')
  CALL IDIAG
  CALL SELWER( 1 )
  CALL SELWER( 2 )
  CALL SELWER( 4 )
  WRITE(6,1000) THEORY,PRESS
C
CSKT3 ANFANG PROBLEMLAUF *****
C
  301 I = I + 1
  CALL AWPZKL(HALTEN,*400)
C
  CALL EINGSV(1,0.)
  CALL EINGSV(4,0.)
  CALL EINGSV(NE*4+1,0.)
  CALL EINGSV(NE*4+3,0.)
  CALL EINGSV(NE*4+4,0.)
C
CSKT4 SYSTEMEINSTELLUNGEN *****
C
  400 CALL EXIST(EXISTC,*600)
C
CSKT5 ELEMENTSEKTIONEN *****
C
  CALL HPRORG(E1,E2,NUE12,G12,HV,LV,RV,RIV,PHIV,HX
:           ,NX,X,TYPEL,TFEV,PL,QL,NEV,NE,NPA,PRESS,THEORY)
C
CSKT6 SYSTEMGLEICHUNGEN *****
C
  600 CALL LOESE(GAUSS,*400,*799)
C
```

CSKT7 ERGEBNISSE NACHBEREITEN *****

C

```
DET = DETN( )
NDGL = NEGDGL( )
AKSTP = AKST( )
NN = NZKL( )
PS = SLAMB( )
DVST = DVSTEU( )
IAA = IAASCH( )
IVST = IVSTEU( )
DLAMP = DLAMB( )
DK = EUNDV( )
UK1 = UKN1( )
UK2 = UKN2( )
UK3 = UKN3( )
VZ1 = VZWN1( )
VZ2 = VZWN2( )
VZ3 = VZWN3( )
```

C

```
WRITE(5,*) PS,',','-XO(NFVG),',',AKSTP
WRITE(6,2000) PS,XO(NFVG),DET,NDGL,NN,AKSTP
IF( I.EQ.III .OR. I.EQ.NZYK ) GO TO 710
GO TO 730
710 III = III + IIII
DO 720 IS = 1,NE+1
LS = (IS-1)*4 + 1
720 WRITE(6,3000) IS,XO(LS),XO(LS+1),XO(LS+2),XO(LS+3)
730 WRITE(6,4000) UK1,UK2,UK3,DLAMP,IAA,DK
: ,VZ1,VZ2,VZ3,DVST,IVST
```

C

CSKT7A INTERAKTIVE STEUERUNG *****

C

```
799 CALL INTSTU(*301,*801)
```

C

CSKT8 ENDE INDIVIDUELLES PROGRAMM *****

C

```
801 CALL ENDE
STOP
```

C

```
1000 FORMAT( 1X,'THEORY=',I3,10X,'PRESS=',I3///)
2000 FORMAT( 1X,'-- P=',E12.6,' W=',E10.4,' DET=',E10.4
: , ND=',I2,' IT=',I2,' AS=',E10.4
3000 FORMAT( 1X,'NR=',I3,' ',4(E12.4,' '))
4000 FORMAT( 1X,'UK1=',E10.4,' UK2=',E10.4,' UK3=',E10.4
: , DP=',E9.3,' I=',I3,' KB=',E9.3/
: 1X,'VZ1=',E10.4,' VZ2=',E10.4,' VZ3=',E10.4
: , DV=',E9.3,' ISTU(',I2,')'
END
```

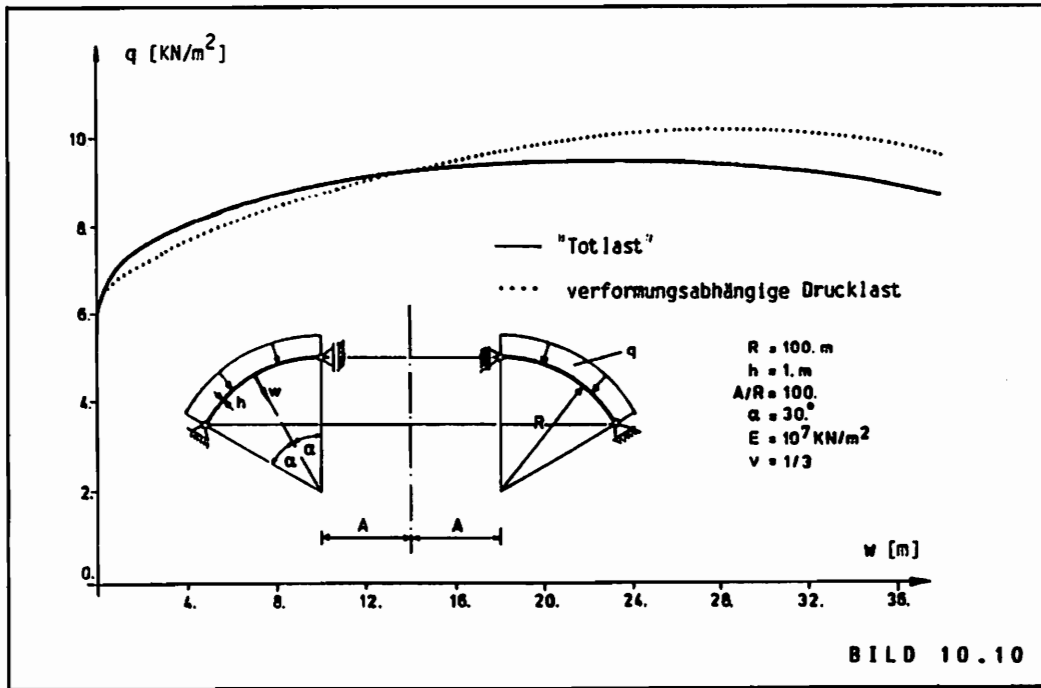
C

```
SUBROUTINE HPRKIN(I1)
DO 150 I=1,8
150 CALL KO(I,I1+I-1,1.0)
RETURN
END
```

```
C
SUBROUTINE HPRMAK(TYPEL,TFE,I1,E1,E2,NUE12,G12,H1,H2,L,CU,RI,PHI
:
,   ,HX,NX,X,RVER,P1,P2,Q1,Q2,PRESS,THEORY)
C
C   MAKRO-ELEMENT-ROUTINE FUER GEOM. NICHTL. AXIALSYMMETRISCHE
C
C   VERFORMUNGEN VON ROTATIONSSCHALEN
C
C   INTEGER TYPEL,TFE,THEORY,PRESS
C   REAL L,NX,NUE12
C   DIMENSION NX(12,3),RVER(8),CE(36),RFV(8),HX(3),X(3),QE(8)
C
C   CALL EL
C   CALL HPRKIN(I1)
C
C   CALL HPRCE(E1,E2,NUE12,G12,L,H1,H2,X,NX,HX,PHI
:
,   ,RI,CU,TFE,RVER,RFV,CE,PRESS,Q1,Q2,THEORY)
C
C   IF( TYPEL.EQ.0 ) GOTO 10
C   CALL HPRLAS(TFE,PHI,RI,L,P1,P2,Q1,Q2,QE,X,NX,HX,CU)
C   CALL ELORGP(QE,8)
C
C   10 DO 310 I=1,8
C   310 CALL ASL(I, -RFV(I))
C   CALL ELORG(CE,8,CE,0,CE,0)
C   CALL SYGL
C   RETURN
C   END
```

10.4.2 TORUSSEGMENT UNTER VERFORMUNGSABHÄNGIGER DRUCKLAST

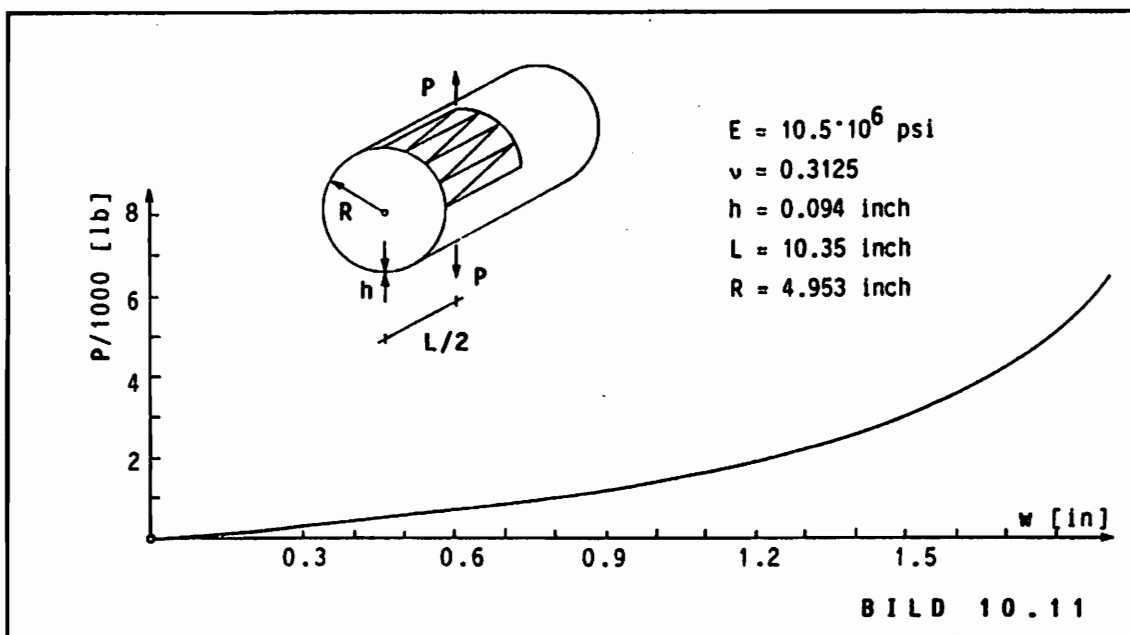
Im Gegensatz zu dünnen elastischen Balken (s. z. B. 10.3.2) sind die Unterschiede in den Systemantworten dünner elastischer Schalen infolge verformungsabhängiger und verformungsunabhängiger Flächenlasten nicht so ausgeprägt, insbesondere wenn solche Randbedingungen betrachtet werden, die die Existenz des zugehörigen Druckpotentials sicherstellen. Als Ausnahme ist daher die in Bild 10.10 dargestellte Torussegment-Schale mit dem Radienverhältnis $A/R = 100$ unter gleichförmiger radialer Flächenlast anzusehen. Interessanterweise zeigen die Ergebnisse einer axialsymmetrischen Berechnung für die Lastfläche "Totlast" und verformungsabhängiger Drucklast z. B. im Durchschlagspunkt Abweichungen der zugehörigen Lasten und Verschiebungen von 6.5 % bzw. 20.5 %. Eine umfassende Analyse dieses Schalenproblems unter Berücksichtigung verschiedener geometrisch nichtlinearer Schalentheorien findet sich in [33]. Das zugehörige INDPRO ist vollkommen analog zu INDPRO8 aufgebaut. Es unterscheiden sich lediglich die Problemdata



(HPRINT), die geometrischen Randbedingungen (EINGSV) sowie die verschiedenen Inkremente in STEUPA.

10.4.3 DIAMETRAL GEZOGENE ZYLINDERSCHALE

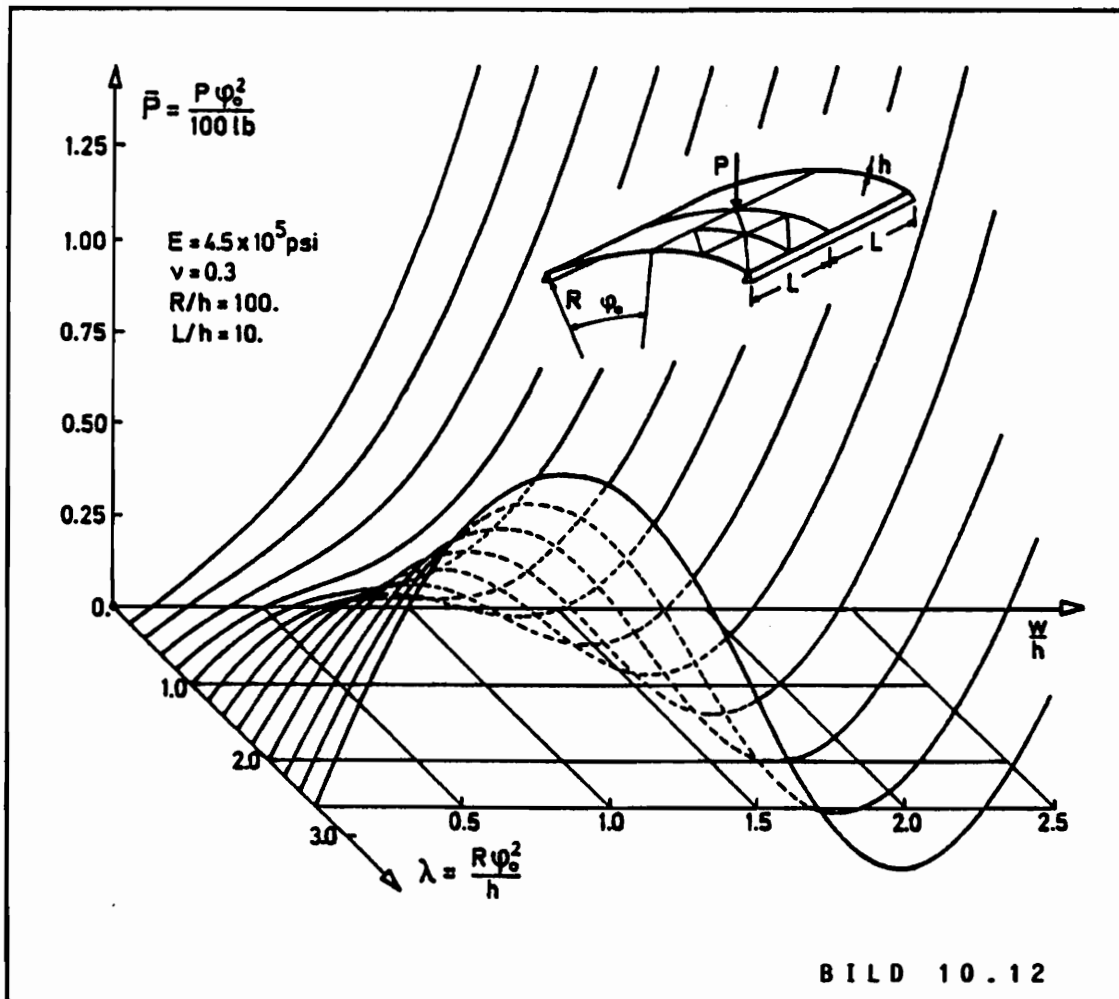
Der im Bild 10.11 dargestellte diametral gezogene Zylinder wurde als hochgradig nichtlineares Schalenbeispiel zur Überprüfung der in Tafel



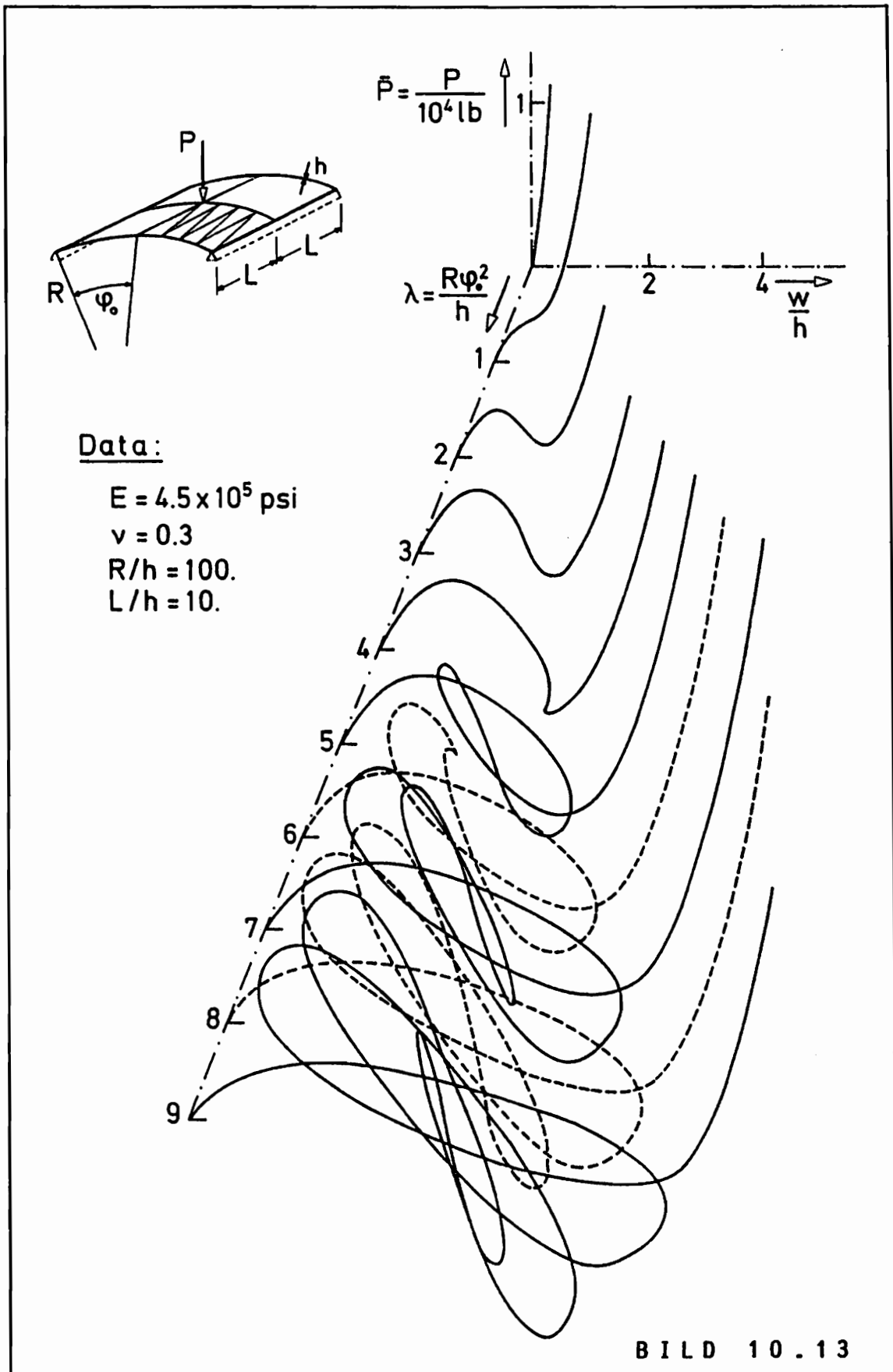
10.1 gezeigten Modellvariante groß/kleiner Rotationen ausgewählt. Dieses Schalenproblem gilt als geeignetes Testbeispiel für die Beurteilung der Qualität geometrisch nichtlinearer Theorien bei Auftreten großer Rotationen [36]. Die Berechnung wurde für ein Achtel der Schale mit 8 vektorisierten Dreieckelementen [34] ausgeführt. Im gesamten Lösungsbereich stimmen die eigenen Ergebnisse mit denen einer FEM-Analyse auf Basis einer aufwendigen kubischen Modellvariante [36] völlig überein. Das reine Spannungsproblem ohne Stabilitätseffekte gestattet hier die Laststeuerung für die gesamte Last-Verschiebungskurve. Auf eine Darstellung und Erläuterung des zugehörigen INDPRO wird wie auch im nächsten Beispiel verzichtet. Seine Struktur entspricht völlig denen vorausgegangener Untersuchungen. Anzumerken ist jedoch, daß die Berechnung komplexerer Schalenprobleme (s. 10.4.4) im BATCH-mode auf der Bochumer CDC Cyber 205 Vektorrechner-Installation ohne interaktive Zugriffsmöglichkeiten die pseudo-vollautomatische Steuerung gemäß Kapitel 9 erfordert.

10.4.4 KREISZYLINDERSCHALENFELD UNTER EINZELLAST

Das Beispiel eines Kreiszylinderschalenfeldes unter Einzellast wurde ausgewählt, um die Effizienz einerseits des vektorisierten FE-Algorithmus für das doppelt gekrümmte Schalen-Dreieckelement mit 54 Freiheitsgraden und andererseits der zur Berechnung notwendigen Steueralgorithmen aufzuzeigen. Zwei gelenkig und frei gelagerte Ränder an jeweils gegenüberliegenden Seiten bilden die geometrischen Randbedingungen des vorliegenden Schalenproblems (s. auch [36] und dort zitierte Arbeiten). Die Behandlung dieses in der Literatur bekannten Beispiels beschränkte sich bisher aus Rechenzeitgründen im wesentlichen auf Systeme mit einem Parameter. Bild 10.12 zeigt nun die Gleichgewichtsfläche eines Zwei-Parameter-Systems, wobei die Kenngröße λ die "Tiefe" der Schalengeometrie beschreibt. Für die erforderlichen 261 Inkrementsschritte mit durchschnittlich 3-5 Iterationen benötigte man 20.0 CPU-Minuten. Die mittlere CPU-Zeit für eine Element-Routine einschließlich Aufbau der Systemgleichungen ist 0.14 CPU-Sekunden. Im Vergleich zur Implementierung auf dem Skalar-Rechner CDC Cyber 855 mit einem vergleichbaren Processor konnte ein Beschleunigungsfaktor von 18 erzielt werden. Anzumerken ist, daß die Projektionen der relativen



Extrema in die (P, λ) -Ebene die sogenannte Cusp-Katastrophe des Durchschlagbeulens liefert. Mit anwachsenden Kenngrößen λ ändert sich jedoch das klassische Antwortspektrum und man erhält eine sich mehrfach durchdringende Gleichgewichtsfläche gemäß Bild 10.13 ("Looping-Effekt"). Dafür wurden 679 Inkrementsschritte mit durchschnittlich 4-5 Iterationen, d. h. insgesamt 62.6 CPU-Minuten benötigt. Zur Berechnung dieser komplizierten Systemantworten insbesondere im Batch-mode der CDC Cyber 205 ohne interaktive Zugriffsmöglichkeiten ist es ratsam, die pseudo-vollautomatische Steuerung des Kapitels 9 zu wählen. Dabei zeigten Vergleichsrechnungen, daß alle drei Varianten LAVUSW, VLAUSW und KOBLME hinreichend stabiles Steuerungsverhalten aufweisen. Außerdem kann in Übereinstimmung mit den Ergebnissen vorangegangener Balkenbeispiele festgestellt werden, daß keine der drei vorgenannten Steuer-routinen zu favorisieren ist.



ANHANG A: FELDER DER GLOBALEN DATENBASIS

Eingerichtet in Subroutine: FLDIRG									
Formelzeichen	Name (MECHEST)	Typ	Dimension	Adresse im Common	Markierung Feld	Erklärungen / Bemerkungen			
						log. Bes.			
-	INTR(i)	I	20	IRINTR	.TRUE		5-Schritt-Informationsblock im Modus Interaktiv Kap. 9.2-3		
-	CINTR(i)	R	45	IRINTR+20	.TRUE		Lastgeschichte Kap. 2.1		
-	GESC(i)	I,R,L	NGESCH	IRGESC	LGESCH		Felder für verschiedene Bezugsgrößen im Modus Konvergenzkriterien Kap. 5 Anzahl der Freiheits-VPKNO = grade pro Knoten		
-	HUGK(i)	R	VPEKNO	IRWUGK	LOUNGL				
-	PUGK(i)	R	VPEKNO	IRPUGK	LOUNGL				
-	XZHA(i)	R	VPEKNO	IRXZHA	LOUNGL. OR. LOVZUH				
-	PUAL(i)	R	VPEKNO	IRPUAL	LOUNGL				
R-alt	PALT(α)	R	NU	IRPALT	LLAAEP	RELAEP	"Alter" System-Lastvektor Kap. 2.2		
R-alt	CJ(α)	R	NU	IRCJ	LLAVPA		s. Gl. 4.2.1 und Gl. 4.3.4		
L	CJQ(α)	R	NU	IRCJQ	(LLAVPA .AND. NEWRAP .AND. MKBLM) .OR. LINTRA	LCJQ	ohne Aktual. beim Mod. New.Raph.		
Y	X01(α)	R	NU	IRX01	LSICHE		Systemverschiebungen		
ΔY	XAI(α)	R	NU	IRXAI	LSICHE		Inkrem. d. Systemversch. Gl. 1.2.1		
R-alt	PAL1(α)	R	NU	IRPAL1	LSICHE. AND. LLAEEP		"Alter" System-Lastvektor Kap. 2.2		
-	GES1(i)	I,R,L	NGESCH	IRGES1	LSICHE. AND. LGESCH		Lastgeschichte Kap. 2.2		
S	WUGQ(i)	L	VPEKNS	IRWUGQ	SELEKT		Diagonalvektor von § Gl. 4.3.8		
Eingerichtet in Subroutine: IRGEL									
Pe	PE(γ)	R	NED	IRPE	PEB0	PEB	Elementlastvektor Kap. 3.1		

ANHANG B: REFERENZLISTE DER COMMONVARIABLEN

LC(190) =	OUTACT	Parameter für die Protokollierung der Aktionen von AENSCHR
LC(191) =	SELECT	Modusvariable für selektive Bogenlänge
LC(192) =	LCJQ	Parameter für die Platzreservierung für CJQ bei der Methode der konstanten Bogenlänge ohne Aktualisierung der Steifigkeitsmatrix
LC(193) =	MKBLM	Modusvariable für das Beibehalten der Orthogonalitätsbedingung in der Methode der konstanten Bogenlänge ohne Aktualisierung der Steifigkeitsmatrix
LC(194) =	LGESCH	Modusvariable für die Lastgeschichte
LC(195) =	LSICH1	Parameter für die Lastgeschichte für die Sicherungsoption
LC(196) =	LSICHE	Modusvariable für die Sicherungsoption
LC(197) =	SYMME	Modusvariable für symmetrische Elementsteifigkeitsmatrizen
LC(198) =	HEBO	Modusvariable für den Nichtgebrauch der Elementhilfsmatrix
LC(199) =	PEBO	Modusvariable für den Elementlastvektor
LC(200) =	PEB	Parameter für den Elementlastvektor
LC(201) =	LSTEUP	Modusvariable für die Steuerroutine "STEUPA"
LC(202) =	LSTANL	Modusvariable für Standard-Last-Steuerung
LC(203) =	KSTANL	Kontrollvariable für Standard-Last-Steuerung
LC(204) =	LSTANV	Modusvariable für Standard-Verschiebungs-Steuerung
LC(205) =	LLAVPA	Modusvariable für Last-Verschiebungs-u.s.w.-Steuerung
LC(206) =	VERSCH	Parameter für Verschiebungssteuerung
LC(207) =	LLAVUS	Modusvariable für Verschiebungs-Last-u.s.w.-Steuerung
LC(208) =	LVLAUS	Parameter für die Platzreservierung für CJ bei Verschiebungssteuerung im allgemeinen und bei der Methode der konstanten Bogenlänge
LC(209) =	LKOBLM	Modusvariable für die Methode konstanter Bogenlänge
LC(210) =	SZYKLS	Modusvariable für Standard-MESY3-Lastzyklen
LC(211) =	FESTV	Parameter für das Beibehalten des Kontrollfreiheitsgrades

LC(212)	=	SCHRIT	Parameter zum Einleiten einer neuen Laststufe
LC(213)	=	TAUSCH	Parameter zum Wechseln der Lösungsmethode zwischen Last- und Verschiebungssteuerung
LC(214)	=	NAKTIT	Parameter zum Überspringen des ersten Schritts der Gauß-Elimination, die Steifigkeitsmatrix ist auf Dreiecksform
LC(215)	=	CEXIST	Parameter zum Beibehalten der alten Steifig- keitsmatrix
LC(216)	=	IEXIST	Parameter für Iterationszyklen
LC(217)	=	TEXIST	Parameter zum Beibehalten von Steifigkeitsmatrix und Restkraftvektor
LC(218)	=	WERBPA	Modusvariable für das Beibehalten der Rand- bedingungen
LC(219)	=	RESTPA	Modusvariable für Restart oder interaktive Steuerung
LC(220)	=	ONEAND	Parameter zum Beibehalten der Steuerparameter beim Restart
LC(221)	=	LARCHP	Modusvariable für Möglichkeit der Archivierung von Ergebnissen für einen Restart
LC(222)	=	LAKSTP	Modusvariable für den Steifigkeitsparameter
LC(223)	=	ERSTPA	Parameter für das Vorliegen des ersten Steifig- keitsparameters
LC(224)	=	AKSTPA	Parameter für das Vorliegen des aktuellen Steifigkeitsparameters
LC(225)	=	LLAAEP	Modusvariable für nichtproportionale Last- änderung
LC(226)	=	RELAEP	Parameter für das Vorhandensein von Platz für den alten Lastvektor
LC(227)	=	LPDEAD	Modusvariable für Totlast ("dead-load")
LC(228)	=	DEADPL	Parameter für Totlast für die lokalen Konver- genzkriterien für die Ungleichgewichts- kräfte
LC(229)	=	DEADPG	Parameter für Totlast für das globale Konver- genzkriterium für die Ungleichgewichts- kräfte
LC(230)	=	NEWRAP	Modusvariable für den Newton-Raphson- Algorithmus

LC(231) = LSNEWR	Modusvariable für den Standard-Newton-Raphson-Algorithmus
LC(232) = LMNEWR	Modusvariable für den modifizierten Newton-Raphson-Algorithmus
LC(233) = LUNGLK	Modusvariable für die Konvergenzkriterien mit Ungleichgewichtskräften
LC(234) = LOUNGL	Parameter für die lokalen Konvergenzkriterien mit Ungleichgewichtskräften
LC(235) = GLUNGL	Parameter für das globale Konvergenzkriterium mit Ungleichgewichtskräften
LC(236) = LVZUWA	Modusvariable für die Konvergenzkriterien mit Verschiebungszuwächsen
LC(237) = LOVZUW	Parameter für die lokalen Konvergenzkriterien mit Verschiebungszuwächsen
LC(238) = GLVZUW	Parameter für das globale Konvergenzkriterium mit Verschiebungszuwächsen
LC(239) = KONVGK	Parameter für das Erreichen der Konvergenz in den Ungleichgewichtskräften
LC(240) = KONVGV	Parameter für das Erreichen der Konvergenz in den Verschiebungszuwächsen
LC(241) = KONVG	Parameter für das Erreichen der Konvergenz
LC(242) = LINCR	Modusvariable für den Bezug der Ungleichgewichtskräfte auf das Lastinkrement
LC(243) = LREFE	Modusvariable für den Bezug der Ungleichgewichtskräfte auf die Referenzlast
LC(244) = LINTRA	Modusvariable für interaktiven Betrieb
LC(245) = FEHLER	Parameter für Fehler im aktuellen Berechnungsschritt
LC(246) = FULLI	Modusvariable für voll interaktiven Betrieb im Gegensatz zum halbautomatischen Betrieb
LC(247) = FEHLE1	Parameter für Fehler im vorhergegangenen Berechnungsschritt

IC(227) = MAXDV	Index für die Position des maximalen Verschiebungsinkrements
IC(231) = NVMAX	Index für die Position der Kontroll- und Steuerverschiebung
IC(232) = NVMAX1	Index für die Position der Kontroll- und Steuerverschiebung im letzten Berechnungsschritt
IC(233) = ALAZYK	Anzahl der Berechnungszyklen
IC(234) = ASCHRT	relative Nummer des aktuellen Berechnungsschritts bei einem Restart
IC(235) = ASREST	Nummer des aktuellen Berechnungsschritts
IC(236) = ASOARC	Anzahl der Schritte ohne Archivierung für einen Restart
IC(237) = RGARCH	relative Nummer des nächsten Berechnungsschritts bei einem Restartlauf, der archiviert wird
IC(238) = SUMRGA	Nummer des nächsten Berechnungsschritts, der archiviert wird
IC(239) = ASOIT	Anzahl der Berechnungsschritte ohne Iteration in "SNEWRA"
IC(240) = ITSOAK	Anzahl der Iterationen ohne Aktualisierung der Steifigkeitsmatrix in "SNEWRA"
IC(241) = LESMIT	Nummer des letzten Berechnungsschritts mit Iteration
IC(242) = ASAKST	Anzahl der Berechnungsschritte ohne Änderung der Steifigkeitsmatrix in "MNEWRA"
IC(243) = LEAKST	Nummer des letzten Berechnungsschritts mit Änderung der Steifigkeitsmatrix in "MNEWRA"
IC(244) = RESOIT	Nummer des nächsten Berechnungsschritts, in dem iteriert wird
IC(255) = REITOA	Nummer der nächsten Iteration, in der die Steifigkeitsmatrix aktualisiert wird
IC(256) = RESAKS	Nummer des nächsten Berechnungsschritts, in dem die Steifigkeitsmatrix geändert wird
IC(257) = VPEKNO	Anzahl der Knotenfreiheitsgrade, zum Gebrauch in den relativen Konvergenzkriterien
IC(258) = IRPALT	Indexregister für den "alten" Lastvektor PALT

IC(259) =	IRCJ	Indexregister für das Feld CJ
IC(260) =	IRWUGK	Indexregister für das Feld WUGK
IC(261) =	IRPUGK	Indexregister für das Feld PUGK
IC(262) =	IRXZWA	Indexregister für das Feld XZWA
IC(264) =	IRCJQ	Indexregister für das Feld CJQ
IC(267) =	IRPUAL	Indexregister für das Feld PUAL
IC(269) =	IRPE	Indexregister für das Feld PE
IC(271) =	NTAPER	Kanalnummer für die Restartdatei
IC(272) =	NTAPEA	Kanalnummer für die Archivdatei
IC(273) =	IRXO1	Indexregister für das Feld XO1
IC(277) =	IRXA1	Indexregister für das Feld XA1
IC(282) =	IRPAL1	Indexregister für das Feld PAL1
IC(285) =	IRGESC	Indexregister für das Feld GESC
IC(287) =	NGESCH	Länge von GESC
IC(290) =	IRGES1	Indexregister für das Feld GES1
IC(297) =	IRINTR	Indexregister für das Feld INTR
IC(298) =	NTAPEO	OUTPUT-Kanalnummer für den interaktiven Betrieb
IC(299) =	NTAPEI	INPUT-Kanalnummer für den interaktiven Betrieb
IC(300) =	VPEKNS	Anzahl der Freiheitsgrade je Knoten für die selektive Bogenlänge

CC(23) =	DLAMV	Steuerungsinkrement für die Verschiebung
CC(24) =	DLAMP	Steuerungsinkrement für den Lastkoeffizienten
CC(25) =	DARCC	Steuerungsinkrement für die Bogenlänge
CC(26) =	DARC	aktuelles Inkrement der Bogenlänge
CC(27) =	DVMAX	Verschiebungsinkrement für die Kontroll- und Steuerungsverschiebung
CC(28) =	DVMAX1	Verschiebungsinkrement für die Kontroll- und Steuerungsverschiebung im letzten Berechnungsschritt
CC(29) =	DLAM	Zuwachs des Lastfaktors in der Iteration
CC(30) =	DLAM1	Inkrement des Lastfaktors im letzten Berechnungsschritt
CC(31) =	DLAMIT	Inkrement des Lastfaktors
CC(32) =	ADLAM	normiertes Inkrement des Lastfaktors in der Methode der konstanten Bogenlänge
CC(33) =	ZCONST	Selbstkorrekturfaktor erster Stufe
CC(34) =	SUMLAM	Lastfaktor
CC(35) =	SUMLA1	Lastfaktor im letzten Berechnungsschritt
CC(36) =	EPSLK	Genauigkeitsparameter für das lokale Konvergenzkriterium für die Ungleichgewichtskräfte
CC(37) =	EPSLKM	Genauigkeitsparameter für das lokale Konvergenzkriterium für die maximale Ungleichgewichtskraft
CC(38) =	EPSGK	Genauigkeitsparameter für das globale Konvergenzkriterium für die Ungleichgewichtskräfte
CC(39) =	EPSLV	Genauigkeitsparameter für das lokale Konvergenzkriterium für die Verschiebungszuwächse
CC(40) =	EPSLVM	Genauigkeitsparameter für das lokale Konvergenzkriterium für den maximalen Verschiebungszuwachs
CC(41) =	EPSGV	Genauigkeitsparameter für das globale Konvergenzkriterium für die Verschiebungszuwächse
CC(42) =	UGK1	Norm für das lokale Konvergenzkriterium für die Ungleichgewichtskräfte

CC(43) =	UGK2	Norm für das lokale Konvergenzkriterium für die maximale Ungleichgewichtskraft
CC(44) =	UGK3	Norm für das globale Konvergenzkriterium für die Ungleichgewichtskräfte
CC(45) =	VZW1	Norm für das lokale Konvergenzkriterium für die Verschiebungszuwächse
CC(46) =	VZW2	Norm für das lokale Konvergenzkriterium für den maximalen Verschiebungszuwachs
CC(47) =	VZW3	Norm für das globale Konvergenzkriterium für die Verschiebungszuwächse
CC(48) =	PGDEAD	Bezugsgröße für das globale Konvergenzkriterium für die Ungleichgewichtskräfte
CC(80) =	PJREF	fiktive Auflagerkraft für die Referenzlast
CC(81) =	RJ	fiktive Auflagerkraft für die Ungleichgewichtskräfte
CC(82) =	PJQ	fiktive Auflagerkraft für das Inkrement der Steuerungsverschiebung
CC(86) =	STATPO	Steifigkeitsparameter im ersten Berechnungsschritt
CC(87) =	STATP1	aktueller Steifigkeitsparameter
CC(88) =	PADEAD	Bezugsgröße für das globale Konvergenzkriterium für die Ungleichgewichtskräfte im letzten Berechnungsschritt
CC(90) =	PJQC	Komponente von CJ auf der Steuerposition
CC(92 bis 97)		Name der Restartdatei
CC(98) =	DETO	Mantisse der Determinante im ersten Berechnungsschritt
CC(99) =	IFKTO	Exponent der Determinante im ersten Berechnungsschritt
CC(100) =	DDLAM	euklidische Norm aus dem Verschiebungsinkrement
CC(161 bis 167)		Name der Archivdatei
CC(167) =	GRDP	Sicherungsfaktor für die Last
CC(168) =	GRDV	Sicherungsfaktor für die Verschiebung

ANHANG C: ALPHABETISCHES ROUTINEN - LISTING

	<u>Seite</u>		<u>Seite</u>		<u>Seite</u>			
1	AENSCHR	38	39	IRHEAD	32	77	RLDLT2	-
2	AKINTR	-	40	IRMEAD	32	78	RRDS2	-
3	AKST	33	41	IRPEAD	32	79	RSL	32
4	AKSTPR	24	42	IRSOAD	32	80	RSTART	30
5	ARCH1	-	43	IRVOAD	32	81	SCHNLA	-
6	ARCH2	-	44	IVSTEU	33	82	SELWER	25,24
7	ARCHPA	30	45	KAWERB	24	83	SICHERN	30,8
8	AWELVA	-	46	KOBLME	20	84	SLAMB	33
9	AWLAUF	-	47	KOBSEL	24	85	SNEWRA	24,21
10	AWPZKL	24	48	LAAEPA	7	86	STANLA	20
11	AWSTER	-	49	LAST	-	87	STANV	20
12	AWSYVA	-	50	LASTVT	-	88	START1	-
13	AWZKL	-	51	LAVUSW	20	89	START2	-
14	CJKOBL	-	52	LFBED	-	90	STATIK	31
15	COPYXY	-	53	LGESCHT	7	91	STEUPA	30,20
16	DETN	33	54	LOEQ	-	92	SYGL	-
17	DLAMB	33	55	LOESE	39,38,23	93	SYMELM	9
18	DRUKSG	-	56	LOESTD	-	94	TOTLAST	28
19	DVMAX	33	57	LOESYF	-	95	UKN1	33
20	DVSTEU	33	58	LOEVSO	-	96	UKN2	33
21	DXO	34	59	MDFSMT	-	97	UKN3	33
22	ELORG	32,9	60	MEMOCA	-	98	UNGLKA	27,24
23	ELORGP	32,9	61	MEMOVR	-	99	VLAUSW	20
24	EMLAST	28,9	62	MEMSTR	-	100	VZUWA	29
25	ERHCJ	-	63	MNEWRA	24,22	101	VZWN1	33
26	EUNDV	33	64	MTRTRF	-	102	VZWN2	33
27	EXIST	23	65	NILAST	-	103	VZWN3	33
28	FLDIRG	-	66	NOTHEM	9	104	WUNGLK	28
29	GAUSS	-	67	NRSTLA	-			
30	IAASCH	33	68	NRVEIT	-			
31	IMAXDV	33	69	NRVESR	-			
32	INCREM	-	70	NUNGLK	-			
33	INTERA	38	71	NVZUWA	-			
34	INTSTU	39	72	PVKOBL	-			
35	IRCEAD	32	73	RAKSTPR	-			
36	IRDEAD	32	74	REDEL	-			
37	IRGEL	-	75	REDMUV	-			
38	IRGSAD	32	76	RGECCS	-			

LITERATUR

- [1] PB. Bergan, T.H. Soreide, Solution of Large Displacement and Stability Problems Using the Current Stiffness Parameter, Intern. Conf. Finite Elements Nonl. Solid. Struct. Mech., Geilo, Norway, Vol. 1, 1-23, (1977).
- [2] J.H. Argyris, Continua and Discontinua, Proc. 1st Conf. "Matrix. Meth. Struct. Mech.", Wright-Patterson A.F. Base, Ohio, 11-189, (1965).
- [3] T.H.H. Pian, P. Tong, Variational Formulation of Finite Displacement Analysis, IUTAM Symp. on "High Speed Computing of Elastic Structures", Liege, 43-63, (1970).
- [4] O.C. Zienkiewicz, Incremental Displacement in Non-Linear Analysis, Int. J. Num. Meth. Engng. 3, 587-588, (1971).
- [5] Ch.D. Desai, J.F. Abel, Introduction to the Finite Element Method, V. Nostrand Reinhold Co., (1972).
- [6] J.W. Butterwords, Numerical Post-buckling Analysis, Structural Instability, Course Dept. Civil Engng. Univ. Surrey, IPC Sci. Technol. Press, 111-123, (1973).
- [7] A.C. Lock, A.B. Sabir, Algorithm for Large Deflection Geometrically Nonlinear Plane and Curved Structures, In "Mathematics of Finite Elements and Applications" (ed. J.R. Whiteman), Academic Press, N.Y. 1973, 483-494.
- [8] W.E. Haisler, J.A. Stricklin, J. Key, Displacement Incrementation in Nonlinear Structural Analysis by the Self-Correcting Methods, Int. J. Num. Meth. Engng. 11, 3-10, (1977).
- [9] E. Ramm, Strategies for Tracing the Nonlinear Response Near Limit Points, Nonl. FEM Analysis in Struct. Mech., W. Wunderlich, E. Stein, K.-J. Bathe (Editors), Springer-Verlag Berlin, 63-89, (1981).
- [10] Z. Waszczyszyn, Problemy numeryczne nieliniowej analizy statecznosci konstrukcij sprezystych, Wspolczesne metody analizy statecznosci konstrukcij, Ossolineum, 341-380, (1981).
- [11] E. Riks, The Application of Newton's Method to the Problem of

- Elastic Stability, J. Appl. Mech., 39, 1060-1066, (1972).
- [12] G.A. Wempner, Discrete Approximations Related to Nonlinear Theories of Solids, Int. J. Solids Struct. 7, 1581-1599, (1971).
- [13] E. Riks, The Incremental Solution of Some Basic Problems in Elastic Stability, NLR TR 74005 U, Amsterdam, (1973).
- [14] E. Riks, A Unified Method for the Computation of Critical Equilibrium States of Nonlinear Elastic Systems, Acta Technica Acad. Sci. Hung., 87, 121-141, (1978).
- [15] E. Riks, An Incremental Approach to the Solution of Snapping and Buckling Problems, Int. J. Solids Struct., 15, 529-551, (1979).
- [16] M. Wessels, Das statische und dynamische Durchschlagsproblem der imperfekten flachen Kugelschale bei elastischer rotations-symmetrischer Verformung, Dissertation, Mitteil. Nr. 23 des Instituts für Statik, TU Hannover, (1977).
- [17] M.A. Crisfield, A Fast Incremental/Iterative Solution Procedure that Handles "Snap-Through", Proc. Symp. Comp. Meth. Nonl. Struct. Solid Mech., Washington, (1980).
- [18] U. Eckstein, Nichtlineare Stabilitätsberechnung elastischer Schalentragwerke, TWM 83-3, Ruhr-Univ. Bochum, (1983).
- [19] K.-H. Schrader, MESY3 Benutzerhandbuch Version CDC CYBER 175 FORTRAN 77, Ruhr-Univ. Bochum, (1983).
- [20] K.-H. Schrader, Die Deformationsmethode als Grundlage einer problemorientierten Sprache, Bibliographisches Institut - Mannheim/Zürich, (1969).
- [21] R. Diekkämper, K.-H. Schrader, Benutzerhandbuch zum Programmsystem MESY2, Ruhr-Univ. Bochum, (1981).
- [22] K.-H. Schrader, MESY - Einführung in das Konzept und Benutzeranleitung für das Programm MESY-MINI, Ruhr-Univ. Bochum, (1978).
- [23] J.A. Stricklin, W.E. Haisler, W.A. Riesenmann, Self-Correcting InitialValue Formulations in Nonlinear Structural Mechanics, AIAA Journal, Vol. 9, 2066-2067, (1971).
- [24] J.A. Stricklin, W.E. Haisler, W.A. Riesenmann, Evaluation of Solution Procedures for Material and/or Geometrically Nonlinear

- Structural Analysis, AIAA Journal, Vol.11, 292-299, (1973).
- [25] P.G. Bergan, R.W. Glough, Convergence Criteria for Iterative Processes, AIAA Journal, Vol. 10, 1107-1108, (1972).
- [26] W. Pietraszkiewicz, Finite rotations and Lagrangean description in the non-linear theory of shells, Polish Scientific Publishers, Warszawa-Posnan, (1979).
- [27] W. Pietraszkiewicz, Finite rotations in the non-linear theory of thin shells, CISM-Course, "Thin Shells", Udine, (1977).
- [28] L.-P. Nolte, Beitrag zur Herleitung und vergleichende Untersuchung geometrisch nichtlinearer Schalentheorien unter Berücksichtigung großer Rotationen, Dissertation, IfM 39, Ruhr-Univ. Bochum, (1983).
- [29] L.-P. Nolte, J. Makowski, H. Stumpf, On the derivation and comparative analysis of large rotation shell theories, to appear in Ing.-Arch, (1985).
- [30] J. Makowski, L.-P. Nolte, Simple equations in terms of displacements for finite axisymmetric deflections of shells of revolution, sent for publication.
- [31] J. Makowski, L.-P. Nolte, H. Stumpf, Finite in-plane deformations of flexible rods - Insight into nonlinear shell problems, IFM 46, Ruhr-Univ. Bochum, (1985).
- [32] N. Tewes, Beitrag zur vergleichenden Untersuchung nichtlinearer eindimensionaler Strukturprobleme auf der Basis des Programmsystems MESY3, Diplomarbeit, IfM, Ruhr-Univ. Bochum, (1985).
- [33] L.-P. Nolte, J. Makowski, Pressure loaded shells undergoing different levels of nonlinearity, to appear in MRC, (1985).
- [34] L.-P. Nolte, B. Schieck, Vectorized high precision finite elements with applications to nonlinear shell problems, Parallel Computing 85, North-Holland, Amsterdam, (1986).
- [35] G. R. Cowper, CURSHL: A high-precision finite element for shells of arbitrary shape, N.R.C. of Canada, Aeronautical Report, (1972).
- [36] R. Harte, Doppelt gekrümmte finite Dreieckelemente für die lineare

und geometrisch nichtlineare Berechnung allgemeiner Flächentragwerke, TWM 82-10, Ruhr-Univ. Bochum, (1982).

- [37] A. B. Sabir, A. C. Lock, Large deflection, geometrically nonlinear finite element analysis of circular arches, Int. J. Mech. Sci., 37-47, 15, (1973).
- [38] L.-P. Nolte, R. Diekkämper, J. Makowski, Two efficient finite element formulations in the analysis of highly nonlinear deformations of thin elastic beams, ZAMM 65, 4, T162-164, (1985).
- [39] J. Argyris, S. Symeonidis, Comp. Meth. Appl. Mech. Eng. 26, 75, (1981).
- [40] T. Belytschko, L. W. Glamm, Applications of high corotational stretch theories to nonlinear FE-analysis, Comp. & Struc., Vol. 10, 175-182, (1979).
- [41] J. Mescall, Numerical solutions of nonlinear equations for shells of revolution, AIAA Journal, Nr. 4, Technical Notes, 2041-2043, (1966).
- [42] A. Peer-Kasperska, Z. Kasperski, J. Makowski, Rozwiazanie zagadnienia nieliniowej statecznosci powlok obrotowych metoda elementow skonczonych. Mat. V Konf. "Metody Komputerowe w Mechanice Konstrukcji", (1981).

Mitteilungen aus dem Institut für Mechanik

- Nr. 1 Theodor Lehmann:
Große elasto-plastische Formänderungen
- Nr. 2 Bogdan Raniecki/Klaus Thermann:
Infinitesimal Thermoplasticity and Kinematics of Finite Elastic-Plastic Deformations. Basic Concepts
- Nr. 3 Wolfgang Krings:
Beitrag zur Finiten Element Methode bei linearem, viscoelastischem Stoffverhalten
- Nr. 4 Burkhard Lücke:
Theoretische und experimentelle Untersuchung der zyklischen elasto-plastischen Blechbiegung bei endlichen Verzerrungen
- Nr. 5 Knut Schwarze:
Einfluß von Querschnittsverformungen bei dünnwandigen Stäben mit stetig gekrümmter Profilmittellinie
- Nr. 6 Hubert Sommer:
Ein Beitrag zur Theorie des ebenen elastischen Verzerrungszustandes bei endlichen Formänderungen
- Nr. 7 H. Stumpf/F.J. Biehl:
Die Methode der orthogonalen Projektionen und ihre Anwendung zur Berechnung orthotroper Platten
- Nr. 8 Albert Meyers:
Ein Beitrag zum optimalen Entwurf von schnelllaufenden Zentrifugenschalen
- Nr. 9 Berend Fischer:
Zur zyklischen elastoplastischen Beanspruchung eines dickwandigen Zylinders bei endlichen Verzerrungen
- Nr. 10 Wojciech Pietraszkiewicz:
Introduction to the non-linear theory of shells
- Nr. 11 Wilfried Ullenboom:
Optimierung von Stäben unter nichtperiodischer dynamischer Belastung

- Nr. 12 Jürgen Güldenpfennig:
Anwendung eines Modells der Vielkristallplastizität auf ein Problem gekoppelter elasto-plastischer Wellen
- Nr. 13 Pawel Rafalski:
Minimum Principles in Plasticity
- Nr. 14 Peter Hilgers:
Der Einsatz eines Mikrorechners zur hybriden Optimierung und Schwingungsanalyse
- Nr. 15 Hans-Albert Lauert:
Optimierung von Stäben unter dynamischer periodischer Beanspruchung bei Beachtung von Spannungsrestriktionen
- Nr. 16 Martin Fritz:
Berechnung der Auflagerkräfte und der Muskelkräfte des Menschen bei ebenen Bewegungen aufgrund von kinematographischen Aufnahmen
- Nr. 17 H. Stumpf/F.J. Biehl:
Approximations and Error Estimates in Eigenvalue Problems of Elastic Systems with Application to Eigenvibrations of Orthotropic Plates
- Nr. 18 Uwe Kolberg:
Variational principles and their Numerical Application to Geometrically Nonlinear v. Karman Plates
- Nr. 19 Heinz Antes:
Über Fehler und Möglichkeiten ihrer Abschätzung bei numerischen Berechnungen von Schalentragwerken
- Nr. 20 Czeslaw Wozniak:
Large Deformations of Elastic and Non-Elastic Plates, Shells and Rods
- Nr. 21 Maria K. Duzek:
Problems of Geometrically Non-Linear Theory of Plasticity
- Nr. 22 Burkhard von Bredow:
Optimierung von Stäben unter stochastischer Erregung
- Nr. 23 Jürgen Preuss:
Optimaler Entwurf von Tragwerken mit Hilfe der Mehrzielmethode

- Nr. 24 Ekkehard Goßmann:
Kovarianzanalyse mechanischer Zufallschwingungen bei Darstellung der mehrfachkorrelierten Erregungen durch stochastische Differentialgleichungen
- Nr. 25 Dieter Weichert:
Variational Formulation and Solution of Boundary-Value Problems in the Theory of Plasticity and Application to Plate Problems
- Nr. 26 Wojciech Pietraszkiewicz:
On Consistent Approximations in the Geometrically Non-Linear Theory of Shells
- Nr. 27 Georg Zander:
Zur Bestimmung von Verzweigungslasten dünnwandiger Kreiszyylinder unter kombinierter Längs- und Torsionslast
- Nr. 28 Pawel Rafalski:
An Alternative Approach to the Elastic-Viscoplastic Initial-Boundary Value Problem
- Nr. 29 Heinrich Oeynhausen:
Verzweigungslasten elastoplastisch deformierter, dickwandiger Kreiszyylinder unter Innendruck und Axialkraft
- Nr. 30 Franz-Josef Biehl:
Zweiseitige Eingrenzung von Feldgrößen beim einseitigen Kontaktproblem
- Nr. 31 Maria K. Duszek:
Foundations of the Non-Linear Plastic Shell Theory
- Nr. 32 Reinhard Piltner:
Spezielle finite Elemente mit Löchern, Ecken und Rissen unter Verwendung von analytischen Teillösungen
- Nr. 33 Petrisor Mazilu:
Variationsprinzip der Thermoplastizität I. Wärmeausbreitung und Plastizität
- Nr. 34 Helmut Stumpf:
Unified Operator Description, Nonlinear Buckling and Post-Buckling Analysis of Thin Elastic Shells

- Nr. 35 Bernd Kaempf:
Ein Extremal-Variationsprinzip für die instationäre Wärmeleitung mit
einer Anwendung auf thermoelastische Probleme unter Verwendung
der finiten Elemente
- Nr. 36 Alfred Kraft:
Zum methodischen Entwurf mechanischer Systeme im Hinblick auf
optimales Schwingungsverhalten
- Nr. 37 Petrisor Mazilu:
Variationsprinzip der Thermoplastizität
II. Gekoppelte thermomechanische Prozesse
- Nr. 38 Klaus-Detlef Mickley:
Punktweise Eingrenzung von Feldgrößen in der Elastomechanik und
ihre numerische Realisierung mit Fundamental-Splinefunktionen
- Nr. 39 Lutz-Peter Nolte:
Beitrag zur Herleitung und vergleichende Untersuchung geometrisch
nichtlinearer Schalentheorien unter Berücksichtigung großer
Rotationen
- Nr. 40 Ulrich Blix:
Zur Berechnung der Einschnürung von Zugstäben unter Berücksichti-
gung thermischer Einflüsse mit Hilfe der Finite-Element-Methode
- Nr. 41 Peter Becker:
Zur Berechnung von Schallfeldern mit Elementmethoden
- Nr. 42 Dietmar Bouchard:
Entwicklung und Anwendung eines an die Diskrete-Fourier-Trans-
formation angepaßten direkten Algorithmus zur Bestimmung der
modalen Parameter linearer Schwingungssysteme
- Nr. 43 Uwe Zdebel:
Theoretische und experimentelle Untersuchungen zu einem thermo-
plastischen Stoffgesetz
- Nr. 44 Jan Kubik:
Thermosdiffusion flows in a solid with a dominant constituent
- Nr. 45 Horst J. Klepp
Über die Gleichgewichtslagen und Gleichgewichtsbereiche nichtlinearer
autonomer Systeme

Nr. 46 J. Makowski/L.-P. Nolte/H. Stumpf

Finite in-plane deformations of flexible rods - insight into nonlinear shell problems

Nr. 47 Franz. K. Labisch

Grundlagen einer Analyse mehrdeutiger Lösungen nichtlinearer Randwertprobleme der Elastostatik mit Hilfe von Variationsverfahren

Nr. 48 J. Chroscielewski/L.-P. Nolte

Strategien zur Lösung nichtlinearer Probleme der Strukturmechanik und ihre modulare Aufbereitung im Konzept MESY

**Mitteilungen aus dem Institut für Mechanik
RUHR-UNIVERSITÄT BOCHUM
Nr. 48**