

DIPLOMARBEIT

Thema

Berechnung monolithischer Decken-Balken-Systeme

-Umsetzung in einer webbasierten Umgebung-

Bearbeitungszeitraum 22.10.2001 - 22.12.2001

Betreuer o.Prof.Dr.Dr.h.c.mult. Peter Jan Pahl
und Dipl.-Ing Urs Wihlfahrt

Jan Tulke
Berlin 2001

Diplomarbeit

am
Fachgebiet Theoretische Methoden der Bau- und Verkehrstechnik
Institut für Bauingenieurwesen
Technische Universität Berlin

Autor: cand.-Ing. Jan Tulke, Matrikelnummer 180 410, 9. Fachsemester

Beginn: 22.10.2001

Abgabe: 21.12.2001

Erklärung

Hiermit versichere ich, die vorliegende Arbeit nur mit den angegebenen Quellen und Hilfsmitteln und ohne fremde Hilfe angefertigt zu haben.

Berlin den 21.12.01

Jan Tulke

Inhaltsverzeichnis

AUFGABENSTELLUNG UND VORGABEN

1. Einleitung	6
2. Vorgaben	7
3. Aufgabenstellung.....	8

KAPITEL A ERWEITERUNG DES PLATTENPROGRAMMS UM DIE MEMBRANWIRKUNG

1. Implementiertes Plattenelement.....	9
2. Membranwirkung	9
3. Implementierung	10

KAPITEL B THEORETISCHE HERLEITUNG DES BALKENELEMENTES

1. Grundlagen.....	14
2. Dehnstab.....	17
2.1 Ausgangselement.....	17
2.2 Transformation in die Grundrißebene.....	19
2.3 Berücksichtigung der Exzentrizität	21
3. Biegebalken.....	24
3.1 Ausgangselement.....	24
3.2 Transformation in die Grundrißebene.....	27
4. Gesamtelementmatrix.....	30
5. Verwendete Koordinatensysteme	32
5.1. Globales Koordinatensystem (Typ 1).....	32
5.2. Lokales Koordinatensystem der Lager (Typ 2).....	32
5.3. Lokales Koordinatensystem des Unterzuges (Typ 3).....	32
5.4. Berechnungsablauf.....	33
5.5. Durchführung der Transformationen.....	33
6. Nachlaufrechnung.....	34

KAPITEL C IMPLEMENTIERUNG DES STABELEMENES

1. Interface.....	36
2. Elementklasse.....	38
3. Berechnung auf Systemebene.....	42

KAPITEL D PROGRAMMTTEST

1. Test des Stabelementes	44
1.1. Handrechnungen.....	44
1.1.1. Dehnanteil	44
1.1.2. Biegeanteil.....	45
1.1.3. Gesamttragverhalten.....	46
1.2. Programmergebnisse	46
2. Test der Membranwirkung.....	48
2.1. Handrechnung	48
2.1.1. Kragsscheibe unter Zugeinwirkung.....	48
2.1.2. Scheibe mit eingprägter Verzerrung.....	48
2.2. Programmergebnisse	50
3. Test der Plattentragwirkung	51

KAPITEL E	BESTIMMUNG DER MITWIRKENDEN PLATTENBREITE	
1. Konzept.....		52
2. Algorithmus.....		53
2.1. Bestimmung der Spannungsverteilung.....		53
2.2. Bestimmung der mitwirkenden Plattenbreite.....		56
2.2.1. Bestimmung der Spannungsfläche.....		56
2.2.2. Bestimmung der mitwirkenden Breite.....		57
2.3. Grundrißansicht.....		58
3. Implementierung.....		58
KAPITEL F	FEM-ANALYSE EINFACHER DECKENSYSTEMEN	
1. Allgemeines.....		59
2. Einfach gelagerter Balken.....		65
2.1. Flächenlast.....		65
2.2. Punktlast.....		68
3. Kragarm.....		69
4. Zweifeldriger Durchlaufträger.....		71
5. Dreifeldriger Durchlaufträger.....		73
6. Konvergenz der Durchbiegung.....		75
7. Beurteilung.....		77
KAPITEL G	SPEZIFIKATION DER OBERFLÄCHE	
1. Konzept der Oberfläche.....		78
2. Paketstruktur.....		79
2. Paketstruktur.....		80
3. Leistungsumfang.....		81
3.1. Makromodelleditor.....		81
3.1. Makromodelleditor.....		82
3.2 Mikromodelleditor.....		82
3.2 Mikromodelleditor.....		83
3.3. FEM-Programm.....		78
4. Erweiterungsmöglichkeiten.....		87
5. Anwendungsbeispiel.....		90
KAPITEL H	EINBETTUNG IN EINE WEBUMGEBUNG	
1. Allgemeines.....		95
2. Überführung in ein Applet.....		95
3. Einbettung in die HTML-Seite.....		96
3.1. Erzeugen eines jar-Archives.....		96
3.1.1. Probleme mit Bildern.....		97
3.2. Das <APPLET>-Tag.....		97
4. Sicherheitsbeschränkungen.....		98
4.1. Java Sandbox.....		98
4.2. Java plug-in.....		99
4.3. policy-files.....		100
4.4. signierte jar-Archive.....		101
4.5. gewählte Variante.....		103
4.6. web-basierte Dokumentation.....		104
KAPITEL I	LITERATUR	105

Aufgabenstellung und Vorgaben

1. Einleitung

Plattenbalkenquerschnitte sind ein wichtiges Tragelement des konstruktiven Stahlbetonbaus. Die in den Zustand II übergehende Zugzone enthält weniger totes Gewicht und die Druckzone wird durch das Mitwirken der Platte vergrößert. Die daraus resultierende erhöhte Tragfähigkeit, die verminderte Verformung sowie der geringere Stahlbedarf gegenüber reinen Plattensystemen sind Gründe für die häufige Anwendung im Ingenieurhochbau.

Doch obwohl heutzutage das Tragverhalten nahezu aller Plattensysteme mit den Methoden der modernen numerischen Mathematik analysiert wird, wird dieser Weg bei monolithischen Decken-Balken-Systemen noch sehr selten beschritten. Die Gründe dafür liegen in der Problematik der Erfassung zweier unterschiedlicher Tragsysteme, nämlich Platte und Balken, in einer einheitlichen Theorie.

Zwar ist eine einheitliche Modellierung mit dreidimensionalen finiten Elementen möglich, jedoch führt dies zu einem unverhältnismäßig großen Berechnungsaufwand, der selbst für heutige Rechnersysteme beträchtlich ist.

Hinzu kommt die Schwierigkeit der ingenieurmäßigen Interpretation der Ergebnisse einer solchen Berechnung. So handelt es sich bei den Kraftgrößen um Spannungen und nicht wie aus der alltäglichen Arbeit bekannt um deren integrale Größen, den Schnittkräften. Dies bedeutet, daß die so gewonnenen Ergebnisse nicht ungefiltert der Bemessung zugeführt werden können.

Aus den eben erläuterten Gründen wird meist versucht, das Tragverhalten von monolithischen Decken-Balken-Systemen durch Einführung bestimmter Näherungen mit Hilfe zweidimensionaler Elemente zu erfassen. Im einfachsten Fall durch die bloße Erhöhung der Plattendicke einzelner Elemente.

Eine Verfeinerung liefert die Kopplung von eindimensionalen Stabelementen mit zweidimensionalen Plattenelementen. Die Berechnungsebene bleibt hierbei die Plattenmittelebene, in der auch die Balkenschwerachse angeordnet wird, wodurch sich Überschneidungen der Volumina und eine nicht dem wirklichen System entsprechende Anordnung des Balkens ergibt. Der Vorteil dieser Vorgehensweise ist, daß neben den Plattenschnittgrößen gleichfalls die Balkenschnittgrößen ermittelt werden, die unmittelbar der Bemessung zugeführt werden können.

Diesen Modulierungsmethoden ist allerdings gemeinsam, daß sie die Tragwirkung des Balkens nur durch eine reine Erhöhung der Biegesteifigkeit der Platte erfassen, welches nur einem Teil des wirklichen Tragverhaltens entspricht. Tatsächlich werden, durch die exzentrische Anordnung des Balkens bezüglich der Plattenmittelebene, im Anschnitt Balken/Platte Schubkräfte übertragen, wodurch ein Teil der Platte am Abtrag der Biegedruckkräfte partizipiert wird. Somit kann die Bemessung für eine breitere Druckzone sowie einen größeren Hebelarm der inneren Kräfte erfolgen, welches unmittelbar ein größeres aufnehmbares äußeres Moment zur Folge hat.

Das bisher meist verwendete Konzept der mitwirkenden Plattenbreite nach Heft 220 DAfStb versucht dieses Tragverhalten von Decken-Balken-Systemen

durch die Bemessung an einem Stab mit einer äquivalenten Breite, der mitwirkenden Plattenbreite, zu erfassen. Grundidee ist dabei die Umwandlung des variablen Spannungsverlaufs längs eines senkrecht zum Balken verlaufenden Schnittes durch die Platte in einen konstanten, sich über die mitwirkende Breite erstreckenden, dem Maximalwert entsprechenden Spannungsbereich. Die Ermittlung dieser Breite erfolgt dabei mittels tabellarischen Rechenhilfen in Abhängigkeit systembeschreibender Parameter, die auf empirischen Ermittlungen beruhen.

Ziel dieser Arbeit soll es sein, die Ermittlung der viel zitierten mitwirkenden Plattenbreite wie sie zur Durchführung einer dem Heft 220 DAfStb entsprechenden Bemessung notwendig ist, auf die gleiche Grundlage wie die Berechnung anderer Deckensysteme zu stellen. Dies bedeutet insbesondere, daß die Schubeinleitung des in Wirklichkeit exzentrisch zur Platte liegenden Balkens in die Platte und die daraus resultierende Scheibenwirkung der Platte erfaßt werden müssen.

Die dieser Arbeit zugrunde liegende Modellierung erfolgt dabei für die Platte mit einem aus Scheiben- und Plattenwirkung zusammengesetzten ebenem Faltwerkselement und für den Unterzug mit einem exzentrisch angeordneten, Biege- und Dehnsteifigkeit erfassenden Balkenelement.

Die Modellierung des Unterzugs wäre auch durch vertikal angeordnete Scheibenelemente möglich, was jedoch hier nicht weiter verfolgt wird. Siehe dazu auch den Aufsatz „Modellierung und Berechnung von Deckenplatten mit Unterzügen“ von W. Wunderlich, G. Kiener und W. Ostermann, veröffentlicht im „Der Bauingenieur“([16]). Hierin wird insbesondere auch auf den Realitätsbezug der einzelnen Modellierungsmethoden eingegangen.

2. Vorgaben

Vor Beginn der Arbeit stand ein Plattenprogramm mit Oberfläche zur Erfassung einfacher schiefwinkliger Rechteckgeometrien zur Verfügung, welches unmittelbaren Bezug auf das, die Vorlesung „Vertiefung II“ des Fachgebietes Theoretische Methoden der Bau- und Verkehrstechnik der TU Berlin, begleitende Skript „Plates – Finite Element Theory and Computer Implementation“ von Prof. Dr. Dr.h.c.mult. Peter Jan Pahl nimmt.

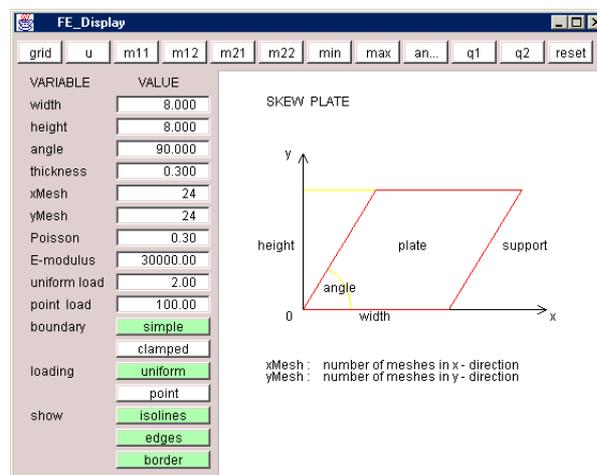


Bild 1: vorgegebenes Plattenprogramm

Über das Kapitel vier des erwähnten Skriptes hinausgehende Dokumentationen des Programmpaketes standen nicht zur Verfügung.

Aus vorangegangenen eigenen bzw. in Gruppenarbeit verfassten Vertiefungshausaufgaben standen wesentliche Teile des Makromodell- und Mikromodelleditors zur Verfügung.

Der verwendete Netzgenerator war am Fachbereich bereits vorhanden.

3. Aufgabenstellung

Zentraler Punkt der Arbeit soll die Erstellung eines rechnergestützten Werkzeuges zur Erfassung des Tragverhaltens monolithischer Decken-Balken-Systeme unter Berücksichtigung aller der bereits oben erwähnten wesentlichen Traganteile sein. Dafür ist neben dem eigentlichen Rechenkern auch die Oberfläche in wesentlichen Teilen so zu verändern, daß die Eingabe allgemeinerer Geometrien, sowie eine problemspezifische Visualisierung ermöglicht wird.

Zum Einsatz kommen übliche Weggrößenelemente unter Voraussetzung linearelastischem Werkstoffverhaltens, linearer Kinematik, sowie Gleichgewichtsbetrachtungen nach Theorie I. Ordnung, d.h. Gleichgewicht am unverformten System.

Als Resultat der Tragwerksanalyse soll neben der Darstellung des Trag- und Verformungsverhaltens der Betrag der mitwirkenden Plattenbreite stehen. Die Veränderlichkeit der mitwirkenden Plattenbreite in Abhängigkeit vom Ort kann dabei im Gegensatz zur Bestimmung nach Heft 220 DAfStb mit erfaßt werden.

Nach Untersuchungen der Güte der erhaltenen Ergebnisse bezüglich Konvergenz, Näherungseinfluß und Abweichungen zu den nach dem herkömmlichen Verfahren ermittelten Werten soll die Umsetzung des Werkzeuges in eine web-basierte Umgebung vollzogen werden.

Damit eventuell verbundene Einschränkungen in der Funktionalität sollen aufgezeigt werden. Ebenso ist eine web-basierte Dokumentation anzufertigen.

Die original Aufgabenstellung ist der Arbeit beigelegt.

Kapitel A Erweiterung des Plattenprogramms um die Membranwirkung

Die Platte der untersuchten Deckensysteme soll, wie bereits beschrieben, im FEM-Modell mit einem ebenen Faltwerkelement abgebildet werden um die durch den Balken eingeleiteten Betondruckspannungen erfassen zu können. Hierbei soll unmittelbar auf das vorgegebene Programmpaket aufgebaut werden.

1. Implementiertes Plattenelement

Zur Beschreibung des reinen Plattentragverhaltens (basierend auf die Kirchhoffhypothese) ist im vorgegebenen Programmpaket ein schubstarres Standardweggrößenelement mit einem nicht vollständigem kubischem Ansatz der Art:

$$u_3 = \vec{f}^T \cdot \vec{c}$$

$$\text{mit } \vec{f}^T = \begin{bmatrix} z_1 & z_2 & z_3 & z_2 z_3 & z_3 z_1 & z_1 z_2 & (z_2 - z_3)^3 & (z_3 - z_1)^3 & (z_1 - z_2)^3 \end{bmatrix}$$

$$\vec{c}^T = \begin{bmatrix} c & c_2 & c_3 & c_4 & c_5 & c_6 & c_7 & c_8 & c_9 \end{bmatrix}$$

und den drei Freiwerten u_3, α_1, α_2 je Knoten implementiert.

Die Diskretisierung erfolgt mittels eines FEM-Netzes bestehend aus 3-Knoten-Dreieckselementen.

Es bestehen die Beziehungen:

$$\alpha_1 = \frac{\partial u_3}{\partial x_2} \quad \text{sowie} \quad \alpha_2 = -\frac{\partial u_3}{\partial x_1}$$

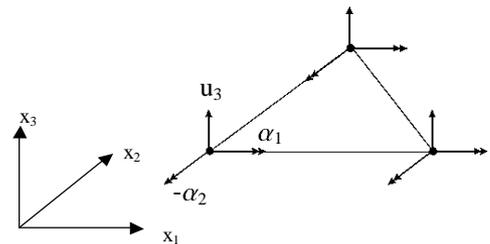


Bild 2: Freiwertbelegung des Plattenelements

Für nähere Angaben bezüglich der Plattentheorie und insbesondere der speziellen Implementierung des vorgegebenen Plattenprogramms siehe [?]

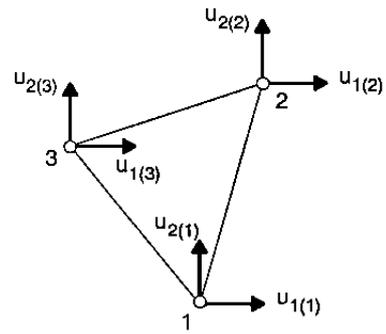
2. Membranwirkung

Für die Abbildung der Membranwirkung der Stahlbetonplatte wird, um die geometische Überlagerung beider Elemente zu ermöglichen, ebenfalls ein 3-Knoten-Dreieckselement gewählt.

Es handelt sich hierbei um das sog. CST-Element, welches das einfachste der Standardweggrößenelemente für die Scheibe ist. Es besitzt je zwei Freiwerten u_1 und u_2 an den Knoten die linear über das Element interpoliert werden, woraus ein konstanter Verzerrungs- und Spannungszustand im Element resultiert. (constant strain triangle)

Ansatz:

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} z_1 & 0 & z_2 & 0 & z_3 & 0 \\ 0 & z_1 & 0 & z_2 & 0 & z_3 \end{pmatrix} \cdot \begin{pmatrix} u_{1(1)} \\ u_{2(1)} \\ u_{1(2)} \\ u_{2(2)} \\ u_{1(3)} \\ u_{2(3)} \end{pmatrix}$$



Aus dem Ansatz konstanter Spannungen im Elementinneren resultiert unmittelbar die

Schwäche dieses Elementes, Biegespannungen nur sehr schlecht darstellen zu können. Dies sei an einem Kragarm verdeutlicht:

Bild 3: Freiwertbelegung des Scheibenelements

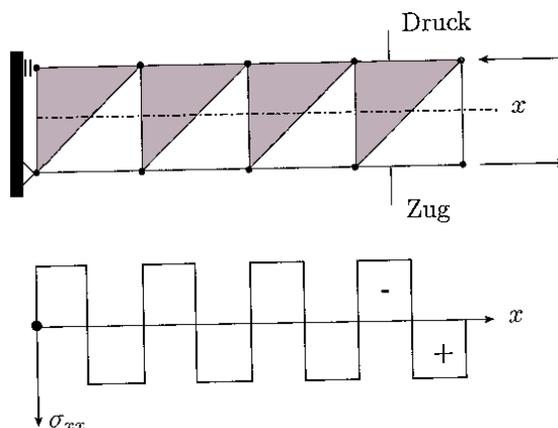


Bild 4: Kragträger mit Endmoment. Berechnung mit CST-Element: Die Normalkraft wechselt sprunghaft das Vorzeichen

Das Element wird trotz dieses Defizits aufgrund der Übereinstimmung mit der Geometrie des Plattenelementes und aufgrund seines einfachen Aufbaus gewählt.

3. Implementierung

Für die Implementierung der Membranwirkung wurde keine weitere Klasse erstellt. Anstelle dessen wurde die Klasse Element des Paketes standard so angepaßt, daß sie beide Tragkomponenten Platten- bzw. Scheibenwirkung erfaßt. Aus objektorientierter Sicht handelt es sich hier somit um ein Objekt. Dies kann mit der Bezeichnung ebenes Faltwerkselement benannt werden.

Im wesentlichen werden in der Klasse zunächst beide Tragkomponenten getrennt von einander betrachtet, d.h. die Elementmatrizen getrennt von einander aufgestellt. Sodann werden, um die Verarbeitung außerhalb der Klasse als eigenständiges Element zu gewährleisten, die beiden entkoppelten Teilmatrizen der Dimensionen (6x6) und (9x9) zu einer Gesamtelementmatrix der Dimension (15x15) verschmolzen. Der zugehörige Elementstützvektor der Verschiebungen, der sich aus den drei Knotenstützvektoren zusammensetzt, hat folglich ebenfalls die Dimension 15.

Nach dem Lösen der Systemgleichung liegen die Verschiebungsergebnisse in den Knotenstützvektoren vor. Die Nachlaufrechnungen zur Darstellung der

entkoppelten Kraftgrößenverläufe werden wiederum nur mit den Einzeltraganteilen berechnet, um unnötige Operationen zu vermeiden.

Der erste wesentliche Schritt war somit die Anpassung aller im gesamten Programmpaket vorhandenen Objekte sowie Speicherungsstrukturen und Zugriffsmechanismen auf die erhöhte Anzahl von Knotenfreiwerten. Der Verschiebungsvektor der Knoten wurde von vorher 3 auf jetzt 5 Freiwerte erhöht. Gleiches gilt für die Objekte zur Repräsentation von Lagerungen und Lasten. Zu beachten war auch der veränderte Ort der Plattenverschiebungsfreiwerte im Knotenstützvektor, der jetzt folgende Gestalt besitzt:

$$\bar{u}_k = \left(\underbrace{u_1 \quad u_2}_{\text{Scheibe}} \quad \underbrace{u_3 \quad \alpha_1 \quad \alpha_2}_{\text{Platte}} \right)$$

Bei den in der Praxis vorkommenden Auflagerungen des Deckensystems auf Wänden ist eine Einprägung der Lagerungsbedingungen parallel bzw. orthogonal zur Auflagerlinie wünschenswert. Hierzu wird schon im vorgegebenem Plattenprogramm an den betreffenden Knoten mit lokalen Koordinatensystemen gearbeitet. Folglich ist eine Drehung der diesen Knoten zugeordneten Anteile der Elementsteifigkeitsmatrizen in das neue Koordinatensystem notwendig. Die bereits vorhandene Methode `rotateStiffnessMatrix()` wurde an die neuen Gegebenheiten angepaßt.

Der Aufbau der Systemsteifigkeitsmatrix wurde ebenfalls an die erhöhte Dimension angepaßt.

Nach dem Lösen dieses Gleichungssystems verblieben die Ergebnisse der Verschiebungsfreiwerte in den jeweiligen am zugehörigen Knoten vorhandenem Koordinatensystem und wurden somit in einer uneinheitlichen Basis im Speicher vorgehalten. Bei Ausführung der Nachlaufrechnung zur Darstellung der Kraftgrößenverläufe bezüglich des globalen Koordinatensystems wurde dann die Rücktransformation notwendig. Dies wurde geändert. Es wird nun bei den entsprechenden Knoten sofort beim Zurückspeichern der Ergebnisse aus der Systemgleichung in die einzelnen Knotenobjekte eine Rücktransformation in das globale Koordinatensystem vorgenommen. Es wird somit eine einheitliche Datenbasis geschaffen. Die Rücktransformation bei den Nachlaufrechnungen entfällt.

Die Nachlaufrechnungen für die Bestimmung der einzelnen Kraftgrößen wurden implementiert und die Funktionalität der Oberfläche um die Darstellung der Scheibenkraft- und -Weggrößenverläufe erweitert. Es wird auch die Darstellung der Hauptspannungen ermöglicht. Zusätzlich wird Eine Methode zur Berechnung der Normalspannung in einem beliebigem Schnitt unter Angabe des zugehörigen Winkels bereitgestellt, die zur Auswertung der mitwirkenden Plattenbreite benötigt wird.

Die theoretischen Grundlagen dieser Methode sind dem nachfolgenden Auszug aus [7] zu entnehmen.

Der -unabhängig von der Richtung einer möglichen Schnittfläche- wirkende Spannungszustand wird durch den Spannungstensor S im globalen Koordinatensystem ausgedrückt. Aus diesem läßt sich -entsprechend den Regeln der Tensoranalysis- der auf einer Fläche mit der Normalen \mathbf{n} wirkende Spannungsvektor \mathbf{s} durch das Skalarprodukt

$$\mathbf{s} = \mathbf{S} \cdot \mathbf{n} \tag{4-3}$$

berechnen, siehe Bild 4-2:

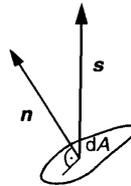


Bild 4-2 Spannungsvektor auf einem differenziellen Flächenelement dA

Analytische Transformation

Gegeben sei ein Punkt mit dem im kartesischen Koordinatensystem x,z angegebenen Spannungszustand. Legt man durch diesen Punkt eine Fläche, deren Nonnalenvektor \mathbf{n} entsprechend Bild 4-4 gegenüber der x -Achse um den Winkel α geneigt ist, so läßt sich der auf diese Fläche wirkende Spannungsvektor \mathbf{s} entsprechend Gleichung (4-3) berechnen.

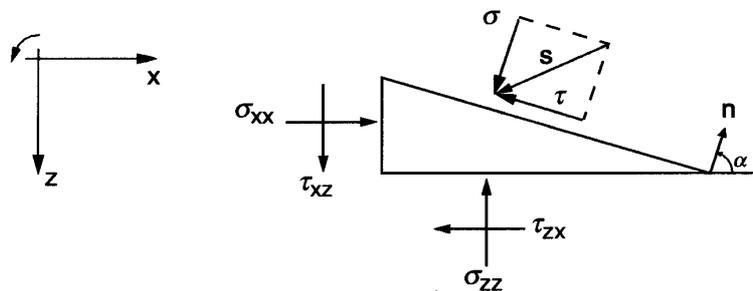


Bild 4-4

Für den ebenen Fall gilt:

$$\mathbf{s} = \begin{bmatrix} s_x \\ s_z \end{bmatrix} = \begin{bmatrix} \sigma_{xx} & \tau_{xz} \\ \tau_{zx} & \sigma_{zz} \end{bmatrix} \cdot \begin{bmatrix} n_x \\ n_z \end{bmatrix} \tag{4-6}$$

mit dem Nonnalenvektor

$$\mathbf{n} = \begin{bmatrix} n_x \\ n_z \end{bmatrix} = \begin{bmatrix} \cos\alpha \\ -\sin\alpha \end{bmatrix} \tag{4-7}$$

Um die auf die Fläche wirkenden Normal- und Schubspannungen zu erhalten, ist der Spannungsvektor des globalen Systems in ein zur Fläche paralleles Koordinatensystem zu transformieren. Die dafür notwendigen trigonometrischen Beziehungen lassen sich in der Transformationsvorschrift

$$\begin{bmatrix} \sigma \\ \tau \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \cdot \begin{bmatrix} s_x \\ s_z \end{bmatrix} \quad (4-8)$$

zusammenfassen. Die Matrix

$$\mathbf{T} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \quad (4-9)$$

wird dabei als Transformationsmatrix für eine ebene Drehung des Koordinatensystems um den Winkel α bezeichnet.

Durch Zusammenfassen der Gleichungen (4-6) und (4-8) erhält man für die Spannungen auf der um α geneigten Fläche die Beziehungen

$$\begin{aligned} \sigma &= \frac{\sigma_{xx} + \sigma_{zz}}{2} + \frac{\sigma_{xx} - \sigma_{zz}}{2} \cdot \cos 2\alpha - \tau_{xz} \cdot \sin 2\alpha \\ \tau &= \frac{\sigma_{xx} - \sigma_{zz}}{2} \cdot \sin 2\alpha + \tau_{xz} \cdot \cos 2\alpha. \end{aligned} \quad (4-10)$$

Das Interface `IcElement` wurde den Veränderungen entsprechend angepaßt, so daß nach außen hin die Form des Zugriffs auf das Element fast unverändert bleibt. Lediglich der Rückgabotyp der Methode `computeDisplacementInterpolationMatrix(..)` wurde von einem Vektor auf eine Matrix geändert. Die alte Methode ist jedoch noch unter `computeDisplacementInterpolationMatrixPlate(..)` ansprechbar.

Das Interface stellt nun folgende zusätzliche Funktionen zur Verfügung.

```
double[][] computeForceAtPoint(double z1, double z2, double z3) ;
// Computes the state of force (n1,n2) at the point in the
// element with the local coordinates z[].

double computeN (double alpha, double x, double y);
double computeU1 (double x, double y) ;
double computeU2 (double x, double y) ;
double computeN11 (double x, double y);
double computeN12 (double x, double y);
double computeN21 (double x, double y);
double computeN22 (double x, double y);
// Compute the scalar value at point (x,y) in the element

double[] computePrincipalN(double x, double y) ;
// Compute the principal values of stress
```

1. Grundlagen

Die Herleitung erfolgt nach Theorie I. Ordnung, d.h. es wird das Gleichgewicht am unverformten System gebildet sowie lineare Kinematik und lineares Stoffgesetz vorausgesetzt.

Ausgangspunkt der Betrachtungen sind die Formulierung der Gleichgewichtsbedingungen der Dehnung und Biegung am differentiell kleinen Stabelement:

$$N' + n = 0 \quad (1)$$

$$M'' + p = 0 \quad (2)$$

Der Zusammenhang zwischen Beanspruchung und Verformung wird mit Hilfe des Hookschen Gesetzes und der Biegedifferentiengleichung beschrieben:

$$\varepsilon = \frac{\sigma}{E} \quad (3) \quad w'' + \frac{M}{EI} = 0 \quad (4)$$

Aus baustatische Sicht beschreiben die ersten beiden Gleichungen das Gleichgewicht und die Gleichung (3) und (4) die Verträglichkeit am differentiell kleinen Element des statischen Systems.

Durch geeignetes Umformen und Einsetzen von (3) und (4) in die Gleichgewichtsausdrücke (1) und (2) erhält man die beiden das Problem beschreibenden Differentialgleichungen:

$$\text{Mit (1) und } \sigma = \frac{N}{A} \text{ und } \varepsilon = u' \Rightarrow \boxed{EA u'' + n = 0} \quad (\text{Dehnung}) \quad (5)$$

$$\text{Mit (2) und } Q = M', Q' = p \Rightarrow \boxed{EI w'''' - p = 0} \quad (\text{Biegung}) \quad (6)$$

Diese beiden Differentialgleichungen sind von den Zustandsfunktionen $w(\bar{x})$ und $u(\bar{x})$ zu erfüllen. Dies geschieht in der Finiten Element Methode (FEM) näherungsweise durch Ansätze der Form:

$$f(\bar{x}) = \sum_e a_i \cdot g_i(\bar{x}) + c(\bar{x}) \quad (7)$$

Dieser Ansatz ist eine Linearkombination von linear unabhängigen Funktionen $g_i(\bar{x})$, wobei jede Funktion $g_i(\bar{x})$ so gewählt wird, daß sie nur jeweils an einem Knoten des FEM-Netzes den Wert eins und an allen anderen Knoten des Netzes den Wert Null annimmt. Zwischen dem Knoten mit dem Wert eins und einem benachbarten, durch eine Elementkante mit diesem verbundenen Knoten wird der Wert der Funktion $g_i(\bar{x})$ problemspezifisch interpoliert. Die Werte a_i stellen unbekannte skalare Gewichtungsfaktoren dar, deren Werte so zu bestimmen sind, daß sich eine möglichst gute Näherungslösung $f(\bar{x})$ der problembeschreibenden DGL ergibt.

Die Anzahl der verwendeten Funktionen $g_i(\bar{x})$ wird durch die Anzahl der verwendeten finiten Elemente bestimmt, womit die Elementanzahl direkten Einfluß auf die Güte der zu erwartenden Näherungslösung hat.

$c(\bar{x})$ ist eine Funktion zur Erfassung der wesentlichen Randbedingungen, die von vornherein gegeben sind. (Tatsächlich werden diese Randbedingungen erst

zu einem späteren Zeitpunkt direkt im Gleichungssystem gesetzt. D.h. es wird zunächst von einem Ansatz der Art $f(\vec{x}) = \sum_e a_j \cdot g_j(\vec{x})$ ausgegangen.

Später geht dann durch das setzen der Randbedingungen – setzen der betreffenden Werte a_j – eine Teilsumme $\sum a_k \cdot g_k(\vec{x})$ von $f(\vec{x})$ in die dann

bekannte Funktion $c(\vec{x})$ über. Das setzen der Randbedingungen entspricht somit der Beseitigung von Unbekannten)

Um die benötigten Bestimmungsgleichungen für die Skalarfaktoren a_i zu erhalten werden folgende Überlegungen angestellt:

Setzt man den gewählten Ansatz für $u(\vec{x})$ bzw. $w(\vec{x})$ in die Differentialgleichung (5) bzw. (6) ein, so wird sich aufgrund des Näherungscharakters der Ansatzfunktion ein Wert nicht identisch gleich Null ergeben.

Die Summe der sich so an jedem Punkt ergebenden Reste:

$$R_1(x) = EA u'' + n \quad (8)$$

$$R_2(x) = EI w'''' - p \quad (9)$$

sollen sich, geeignet über das Lösungsgebiet gewichtet, zu Null ergeben:

$$\boxed{\int_L \delta f_i(\vec{x}) \cdot R_j(\vec{x}) dx = 0} \quad j = 1,2 \quad (10)$$

Als Gewichtsfunktionen kommen mehrere Ansätze in Betracht.

Nach [1] ergeben sich in der Regel die besten Ergebnisse, wenn, der Methode von Galerkin folgend, die Ansatzfunktionen $f_i(\vec{x})$ selbst als Gewichtsfunktionen $\delta f_i(\vec{x})$ gewählt werden. Diese erhält man leicht durch die Ableitung des Lösungsansatzes (7) nach den Skalarfaktoren:

$$\boxed{\delta f_i(\vec{x}) = \frac{\partial f(\vec{x})}{\partial a_i} = g_i(\vec{x})} \quad (11)$$

Somit stehen genau so viel Gewichtsfunktionen $\delta f_i(\vec{x})$ zur Verfügung, wie unbekannte Skalarfaktoren a_i . Es entsteht also ein Gleichungssystem von n Gleichungen für n Unbekannte, welches nach Einprägung der wesentlichen Randbedingungen gelöst werden kann.

Im folgenden werden die Abhängigkeiten der Funktionen von Variablen so wie ihre Indexierung nur noch dargestellt, wenn sie zwecks Unterscheidung unverzichtbar sind. Die aus (5) und (6) mit Hilfe der beschriebenen Überlegungen erhaltenen Bestimmungsgleichungen stellen sich somit wie folgt dar:

$$\int_L \delta u (EA u'' + n) dx = 0 \quad (12) \quad \text{und} \quad \int_L \delta w (EI w'''' - p) dx = 0 \quad (13)$$

Um die Anforderungen an die Ansatzfunktion u bzw. w der Näherungslösung hinsichtlich der Höhe ihrer Ordnung zu reduzieren werden beide Gleichungen nun mittels partieller Integration in geeigneter Art und Weise umgeformt.

$$\int_L \delta u (EA u'' + n) dx = \int_L \delta u EA u'' dx + \int_L \delta u n dx$$

$$\text{mit } \int_L \delta u EA u'' dx = - \int_L \delta u EA u' dx + \left[\underbrace{\delta u EA u'}_N \right]_A^B$$

Der durch die partielle Integration entstandene Randausdruck ist bei Ausschluß des gleichzeitigen Auftretens von Kraft- und Verschiebungsgrößen in den Randpunkten des Gesamtlösungsgebietes gleich Null und kann daher ersatzlos entfallen. So ist bei einer eingepprägten Lagerung der Skalarfaktor a_i am betreffenden Punkt von vornherein bekannt und somit nach der Definition (11) die Gewichtsfunktion δu gleich Null (a_i ist ein bekannter konstanter Wert und somit seine Ableitung gleich Null). Bei einem freien Rand ist die Schnittgröße N aus Gleichgewichtsgründen immer gleich Null.

(Bei Federlagerung oder LF Auflagersenkung in den Randknotenpunkten ist der Ausdruck zu berücksichtigen, da hierbei keine der beiden Größen δu bzw N zu Null wird.)

Es ergeben sich somit aus (12) n Gleichungen der Form:

$$\boxed{\int_L \delta u' EA u' dx - \int_L \delta u n dx = 0} \quad (14)$$

In analoger Weise wird nun mit Gleichung (13) verfahren. Hier ist jedoch eine zweifache Anwendung der partiellen Integration erforderlich, um eine gleichmäßige Verteilung der Ableitungen auf die Größen w und δw zu erreichen. Die Randausdrücke entfallen aufgrund gleicher Überlegungen.

$$\int_L \delta w (EI w'''' - p) dx = \int_L \delta w EI w'''' dx - \int_L \delta w p dx$$

$$\text{mit } \int_L \delta w EI w'''' dx = - \int_L \delta w' EI w''' dx + \left[\underbrace{\delta w EI w'''}_Q \right]_A^B$$

$$\text{und } \int_L \delta w' EI w''' dx = - \int_L \delta w'' EI w'' dx + \left[\underbrace{\delta w'}_{\delta \varphi} \underbrace{EI w''}_M \right]_A^B$$

Es ergeben sich somit aus (13) n Gleichungen der Form:

$$\boxed{\int_L \delta w'' EI w'' dx - \int_L \delta w p dx = 0} \quad (15)$$

Gleichung (14) und (15) bilden die Grundlage aller weiteren Betrachtungen. Es werden nun die Ansätze im finiten Element betrachtet und durch Einsetzen dieser in das jeweils erste Integral in (14) und (15) die Elementsteifigkeitsmatrizen hergeleitet. Das jeweils zweite Integral führt nach dem Einsetzen der Elementansätze zum Elementlastvektoren, dieser wird jedoch nur am Rande betrachtet, da er für das zu entwickelnde Plattenbalkenprogramm nur in der Testphase von Bedeutung ist. Beim entgültigen Programmpaket werden die Lasten in Form von Elementlastvektoren der Plattenelemente aufgebracht. Durch die sich daraus ergebenden Verformungen der Plattenelemente ergeben sich über die Kopplung der Freiwerte von Platte, exzentrischem Balken und

Scheibe Verzerrungen des Balkenelementes, welches die Aktivierung der Balkensteifigkeit zur Folge hat. Die aus der Dehnsteifigkeit des Balkens entstehenden Verschiebungen wiederum aktivieren den Scheibenanteil in der Stahlbetonplatte. (Durch die direkte Kopplung der Knotenfreiwerte beeinflussen sich alle Tragwirkungen gegenseitig und treten nicht so wie hier beschrieben in einer streng getrennten Abfolge auf). Eine explizite Lasteinleitung in das Stabelement ist somit nicht erforderlich.

Gleichung (14) und (15) beschreiben noch das gesamte Problem. Um nun auf die Elementbetrachtung übergehen zu können müssen beide Gleichungen diskretisiert werden, d.h. die Integrale über das Gesamtlösungsgebiet L werden durch die Summe über alle Elemente der Integrale über die Elementlänge S ersetzt. Dieser Übergang setzt allerdings eine C₀-Stetigkeit des Ansatzes voraus, da sonst noch gesonderte Beiträge auf den Rändern der Elemente berücksichtigt werden müssen.

2. Dehnstab

2.1 Ausgangselement

Nach der Diskretisierung durch finite Elemente kann die Systemgleichung (14) in der Form:

$$\boxed{\sum_e \int_s \delta u' EA u' dx - \sum_e \int_s \delta u n dx = 0} \quad (16)$$

geschrieben werden.

Es soll nun das Integral über die Länge eines Elementes ausgewertet werden. Dafür wird in einem lokalen normierten z-Koordinatensystem folgender isoparametrischer Ansatz gewählt:

Geometrie:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} x_{1A} & x_{1B} \\ x_{2A} & x_{2B} \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{2}(1-z) \\ \frac{1}{2}(1+z) \end{pmatrix}$$

$$\vec{x} = \underline{X} \cdot \vec{s} \quad (17)$$

Physik:

$$u(z) = (u_A, u_B) \cdot \begin{pmatrix} \frac{1}{2}(1-z) \\ \frac{1}{2}(1+z) \end{pmatrix} = \underbrace{\frac{1}{2}(1-z)}_1 u_A + \underbrace{\frac{1}{2}(1+z)}_2 u_B$$

$$u(z) = \vec{u}_e^T \cdot \vec{s} \quad (18)$$

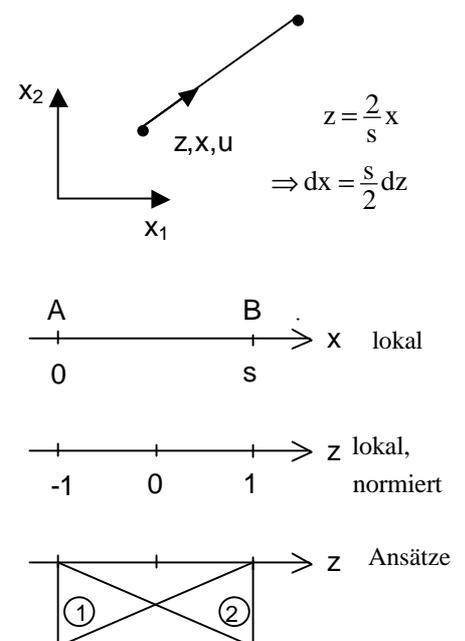


Bild 5: Koordinatensysteme und Formfunktionen

Es wird somit sowohl die Geometrie als auch der Verlauf der Längsverschiebung zwischen zwei Knoten linear interpoliert.

Bestimmung der Ableitungen:

$$\mathbf{u}' = \frac{d\mathbf{u}}{dz} \frac{dz}{dx} = \bar{\mathbf{u}}_e^T \cdot \bar{\mathbf{s}}_z \cdot \frac{2}{s} \quad \text{mit} \quad \bar{\mathbf{s}}_z = \begin{pmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} \quad (19)$$

$$\delta \mathbf{u} = \frac{d\mathbf{u}}{d\mathbf{u}_i} = \bar{\mathbf{e}}_i^T \cdot \bar{\mathbf{s}} \quad (20)$$

$$\delta \mathbf{u}' = \frac{d\mathbf{u}'}{d\mathbf{u}_i} = \bar{\mathbf{e}}_i^T \cdot \bar{\mathbf{s}}_z \cdot \frac{2}{s} \quad (21)$$

(Der Elementverschiebungsvektor $\bar{\mathbf{u}}_e$ ist ein Teilvektor des Systemvektors, dessen Dimension gleich der Anzahl der Knoten mal Anzahl der Verschiebungsfreiwerte pro Knoten ist. $\bar{\mathbf{e}}_i$ kann in gleicher Weise interpretiert werden und übernimmt so das richtige Einsortieren der Elemente in die Systemgleichung)

Einsetzen der Ansätze in das erste Integral von (16) ergibt:

$$\int_s \delta \mathbf{u}' \mathbf{EA} \mathbf{u}' dx = \int_s \bar{\mathbf{e}}_i^T \cdot \bar{\mathbf{s}}_z \cdot \frac{2}{s} \cdot \mathbf{EA} \cdot \bar{\mathbf{u}}_e^T \cdot \bar{\mathbf{s}}_z \cdot \frac{2}{s} \cdot \frac{s}{2} dz = \bar{\mathbf{e}}_i^T \cdot \left[2 \frac{\mathbf{EA}}{s} \int_s \bar{\mathbf{s}}_z \cdot \bar{\mathbf{s}}_z^T dz \right] \cdot \bar{\mathbf{u}}_e \quad (22)$$

Durchführung der Integration:

$$[..] = 2 \frac{\mathbf{EA}}{s} \int_s \begin{pmatrix} \frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{4} & \frac{1}{4} \end{pmatrix} dz = \frac{\mathbf{EA}}{s} \int_{-1}^1 \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{pmatrix} dz = \frac{\mathbf{EA}}{s} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} = \mathbf{K}_{eD} \quad (23)$$

Somit ist das erste Integral von (16) analytisch exakt ausgewertet.

\mathbf{K}_{eD} ist die Elementsteifigkeit des eindimensionalen Dehnstabes.

Auf analoge Weise wird mit dem zweiten Integral von (16) verfahren:

$$\int_s \delta \mathbf{u} \mathbf{n} dx = \int_s \bar{\mathbf{e}}_i^T \cdot \bar{\mathbf{s}} \cdot \mathbf{n} \cdot \frac{s}{2} dz = \bar{\mathbf{e}}_i^T \cdot \left[\frac{\mathbf{ns}}{2} \int_{-1}^1 \begin{pmatrix} \frac{1}{2}(1-z) \\ \frac{1}{2}(1+z) \end{pmatrix} dz \right] = \bar{\mathbf{e}}_i^T \cdot \underbrace{\frac{\mathbf{ns}}{2} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix}}_{\bar{\mathbf{q}}_{eD}} \quad (24)$$

$\bar{\mathbf{q}}_{eD}$ ist der Elementlastvektor.

Durch Einsetzen von (22), (23) und (24) in die Systemgleichung (16) kann diese nun als Summe der Elementbeiträge in folgender Form geschrieben werden:

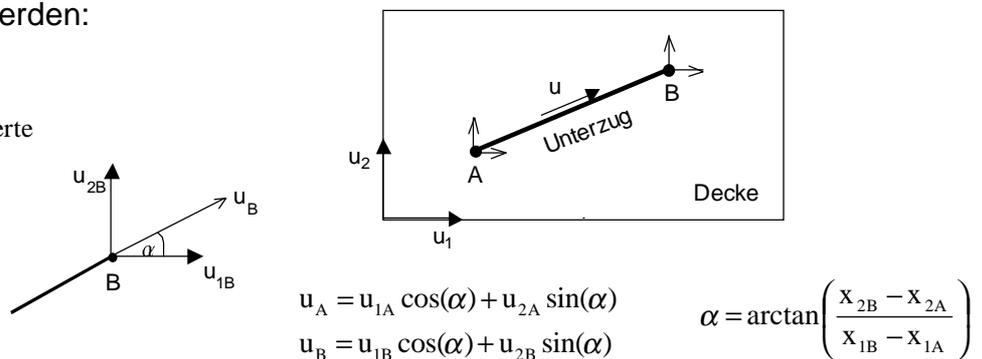
$$\sum_e \bar{\mathbf{e}}_i^T \left[\frac{\mathbf{EA}}{s} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \bar{\mathbf{u}}_e - \begin{pmatrix} \frac{\mathbf{ns}}{2} \\ \frac{\mathbf{ns}}{2} \end{pmatrix} \right] = 0 \quad (25)$$

2.2 Transformation in die Grundrißebene

Gleichung (25) gilt für den eindimensionalen Fall, d.h. sowohl Belastung als auch Verformungen treten nur in Stablängsachse (die dem der Betrachtung zugrunde liegendem Koordinatensystem entspricht) auf. Für die Verwendung dieses Dehnstabes zur rechnerischen Erfassung des Dehntagverhaltens eines Unterzuges von monolithischen Decken-Balken-Systemen ist jedoch eine Beschreibung des Verformungsverhaltens in der Grundrißebene der Deckenplatte notwendig.

Dazu müssen die lokalen Verschiebungen u_A und u_B des Balkens durch die Verschiebungen u_{1A} , u_{2A} und u_{1B} , u_{2B} des globalen Koordinatensystems der Platte ausgedrückt werden:

Bild 6: Transformation der Verschiebungsfreiwerte



$$\begin{aligned} u_A &= u_{1A} \cos(\alpha) + u_{2A} \sin(\alpha) \\ u_B &= u_{1B} \cos(\alpha) + u_{2B} \sin(\alpha) \end{aligned} \quad \alpha = \arctan\left(\frac{x_{2B} - x_{2A}}{x_{1B} - x_{1A}}\right)$$

Somit gilt:

$$\begin{pmatrix} u_A \\ u_B \end{pmatrix} = \begin{pmatrix} \cos(\alpha) & \sin(\alpha) & 0 & 0 \\ 0 & 0 & \cos(\alpha) & \sin(\alpha) \end{pmatrix} \cdot \begin{pmatrix} u_{1A} \\ u_{2A} \\ u_{1B} \\ u_{2B} \end{pmatrix}$$

$$\bar{u}_e = \underline{D} \cdot \hat{u}_e \quad (26)$$

Wird Gleichung (26) in Gleichung (18) bis (21) substituiert, so erhält man:

$$u(z) = \hat{u}_e^T \cdot \underline{D}^T \cdot \vec{s} \quad (18')$$

und für die Ableitungen:

$$u' = \frac{du}{dz} \frac{dz}{dx} = \hat{u}_e^T \cdot \underline{D}^T \cdot \vec{s}_z \cdot \frac{2}{s} \quad (19')$$

$$\delta u = \frac{du}{du_i} = \vec{e}_i^T \cdot \underline{D}^T \cdot \vec{s} \quad (20')$$

$$\delta u' = \frac{du'}{du_i} = \vec{e}_i^T \cdot \underline{D}^T \cdot \vec{s}_z \cdot \frac{2}{s} \quad (21')$$

Die Änderung des \vec{e}_i Vektors in (18')-(21') gegenüber (18)-(21) wird hier und nachfolgend zur Wahrung der Übersichtlichkeit zeichentechnisch nicht erfaßt, da sie nur aus einer Erhöhung der Dimension besteht.

Analog zu (22) werden die Ansätze (18') bis (21') in das erste Integral von (16) eingesetzt:

$$\int_s \delta \underline{\hat{u}}^T \underline{EA} \underline{u}' dx = \int_s \underline{\hat{e}}_i^T \cdot \underline{D}^T \cdot \underline{\hat{s}}_z \cdot \frac{2}{s} \cdot \underline{EA} \cdot \underline{\hat{u}}_e^T \cdot \underline{\hat{s}}_z \cdot \frac{2}{s} \cdot \frac{s}{2} dz = \underline{\hat{e}}_i^T \cdot \left[2 \frac{\underline{EA}}{s} \cdot \underline{D}^T \cdot \int_s \underline{\hat{s}}_z \cdot \underline{\hat{s}}_z^T dz \right] \cdot \underline{\hat{u}}_e$$

$$= \underline{\hat{e}}_i^T \left[\frac{\underline{EA}}{s} \cdot \underline{D}^T \cdot 2 \int_s \underline{\hat{s}}_z \cdot \underline{\hat{s}}_z^T dz \cdot \underline{D} \right] \cdot \underline{\hat{u}}_e = \underline{\hat{e}}_i^T \underbrace{\left[\frac{\underline{EA}}{s} \cdot \underline{D}^T \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \cdot \underline{D} \right]}_{\underline{K}'_{eD}} \cdot \underline{\hat{u}}_e$$

$$\boxed{\underline{K}'_{eD} = \underline{D}^T \cdot \underline{K}_{eD} \cdot \underline{D}} \quad (27)$$

Wie man leicht sieht, ergibt sich die neue Steifigkeitsmatrix \underline{K}'_e in einfacher Weise durch eine Links- bzw. Rechtsmultiplikation der alten Steifigkeitsmatrix mit der transponierten Drehmatrix \underline{D}^T bzw. mit der Drehmatrix \underline{D} selbst, so daß sich die neue Steifigkeitsmatrix unter Berücksichtigung der Transformation in die Grundrißebene wie folgt darstellt:

$$(28) \quad \begin{pmatrix} \cos(\alpha) & 0 \\ \sin(\alpha) & 0 \\ 0 & \cos(\alpha) \\ 0 & \sin(\alpha) \end{pmatrix} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \\ -\cos(\alpha) & \sin(\alpha) \end{pmatrix} \underbrace{\begin{pmatrix} \cos(\alpha) & \sin(\alpha) & 0 & 0 \\ 0 & 0 & \cos(\alpha) & \sin(\alpha) \\ \cos^2(\alpha) & \sin(\alpha)\cos(\alpha) & -\cos^2(\alpha) & -\sin(\alpha)\cos(\alpha) \\ \sin(\alpha)\cos(\alpha) & \sin^2(\alpha) & -\sin(\alpha)\cos(\alpha) & -\sin^2(\alpha) \\ -\cos^2(\alpha) & -\sin(\alpha)\cos(\alpha) & \cos^2(\alpha) & \sin(\alpha)\cos(\alpha) \\ -\sin(\alpha)\cos(\alpha) & -\sin^2(\alpha) & \sin(\alpha)\cos(\alpha) & \sin^2(\alpha) \end{pmatrix}}_{\underline{K}'_{eD}}$$

Hinweis:

Verschiebungen senkrecht zur unverformten Balkenachse erzeugen keine Kräfte. Dies ist durch die bloße Übertragung des eindimensionalen Stabelements in eine zweidimensionale Grundrißebene bedingt. So wird folgende Längenänderung eines Stabelementes, die ausschließlich durch in der Grundrißebene senkrecht zur Stabachse angetragene Verschiebungen der Elementknoten entstanden ist, von dem hier verwendeten Stabelement nicht erfaßt und somit einer solchen Verzerrung kein Widerstand entgegengesetzt.

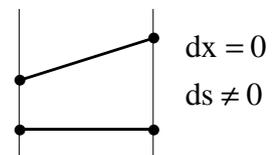


Bild 7: zwängungsfreie Verformung

Der Lastvektor besteht in elementarer Weise an jedem der beiden Knoten aus 2 zu den globalen Koordinatenachsen parallel wirkenden Lastanteilen. Diese erhält man in analoger Weise aus:

$$\underline{\hat{q}}'_{eD} = \underline{D}^T \cdot \underline{\hat{q}}_{eD}$$

(wie schon erwähnt ist ein solcher Lastvektor, der eine längs der Stabachse wirkende Streckenlast erfaßt für die spätere Anwendung von keinerlei Bedeutung)

2.3 Berücksichtigung der Exzentrizität

Berechnungsebene des monolithischen Decken-Balken-Systems ist die Systemebene von Platte und Scheibe, d.h. die Mittelebene der Stahlbetonplatte. Alle verwendeten Verschiebungsfreiwerte beziehen sich auf die in dieser Ebene liegenden Knoten. Der Steg des Plattenbalkenquerschnittes soll bei der Berechnung durch den bisher betrachteten Stab abgebildet werden.

Die Berechnungsachse dieses Stabes ist jedoch seine eigene Schwerachse, die, beim Unterhängen des Stabes unter die Platte/Scheibe, um die Exzentrizität e unterhalb der Berechnungsebene von Platte und Scheibe liegt. Um die Verträglichkeit der Verformungen von Platte/Scheibe auf der einen und Stab auf der anderen Seite zumindest in den Knoten des FEM-Modells zu sichern, müssen die jeweiligen Verschiebungsfreiwerte in Abhängigkeit von der Größe der Exzentrizität zueinander in Beziehung gesetzt werden.

Aus der reinen Plattenbiegung wird zunächst folgende Horizontalverschiebung in der Balkenachse induziert:

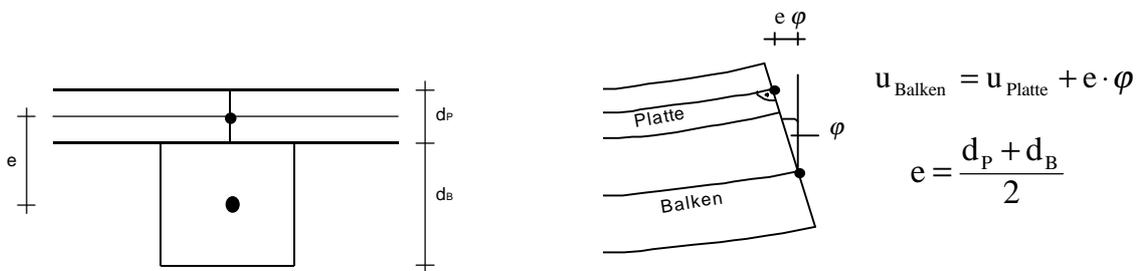


Bild 8: Kopplung der Balken- und Plattenverschiebungsfreiwerte

Nach der für die Platte verwendeten Kirchhoffschen Plattentheorie sind die Horizontalverschiebungen aller Punkte der Mittelebene bei reiner Biegebeanspruchung gleich Null.

Der Winkel φ der Querschnittsverdrehung bezüglich der unverformten Lage kann mittels der ersten Ableitungen der vertikalen Verschiebungsfunktion der Plattenbiegung nach der jeweiligen Koordinatenrichtung berechnet werden. Die vertikale Verschiebungsordinate der Plattenbiegung wird dabei mit u_3 und nicht wie bisher mit w benannt um damit den Übergang vom platteneigenen Koordinatensystem zum gemeinsamen Koordinatensystem von Platte, Scheibe und Balken zu kennzeichnen. Die Bezeichnungen α_1 und α_2 sind um eine Kompatibilität mit den Vorgaben zu sichern, aus [6] übernommen:

$$\begin{aligned} u_{1 \text{ Balken}} &= e \cdot u_{3,1} = -e \cdot \alpha_2 \\ u_{2 \text{ Balken}} &= e \cdot u_{3,2} = +e \cdot \alpha_1 \end{aligned} \quad \text{mit } \alpha_1 = u_{3,2} \text{ und } \alpha_2 = -u_{3,1}$$

Zusätzlich sind noch Verschiebungen aus dem Scheibenanteil der Stahlbetonplatte vorhanden, so daß sich ergibt:

$$\begin{aligned} u_{1 \text{ Balken}} &= u_{1 \text{ Scheibe}} + e \cdot u_{3,1} = u_{1 \text{ Scheibe}} - e \cdot \alpha_2 \\ u_{2 \text{ Balken}} &= u_{2 \text{ Scheibe}} + e \cdot u_{3,2} = u_{2 \text{ Scheibe}} + e \cdot \alpha_1 \end{aligned} \quad (29)$$

Gleichung (29) läßt sich noch in Matrixschreibweise unter Verwendung aller vorhandener Scheiben- und Plattenverschiebungsgrößen wie folgt darstellen:

$$\begin{pmatrix} u_{1A} \\ u_{2A} \\ u_{1B} \\ u_{2B} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & -e & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & e & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -e \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & e & 0 \end{pmatrix} \cdot \begin{pmatrix} u_{1A} \\ u_{2A} \\ u_{3A} \\ \alpha_{1A} \\ \alpha_{2A} \\ u_{1B} \\ u_{2B} \\ u_{3B} \\ \alpha_{1B} \\ \alpha_{2B} \end{pmatrix}$$

$$\hat{\underline{u}}_e = \underline{V} \cdot \underline{u}_{se} \quad (30)$$

Analog zu Abschnitt 2.2 erhält man mittels Substitution in (18') und dann in (18)-(21) die neuen Ausdrücke für die Ansatzfunktion und ihre Ableitungen:

$$u(z) = \underline{u}_{se}^T \cdot \underline{V}^T \cdot \underline{D}^T \cdot \underline{s} \quad (18'')$$

und für die Ableitungen:

$$u' = \frac{du}{dz} \frac{dz}{dx} = \underline{u}_{se}^T \cdot \underline{V}^T \cdot \underline{D}^T \cdot \underline{s}_z \cdot \frac{2}{s} \quad (19'')$$

$$\delta u = \frac{du}{du_i} = \underline{e}_i^T \cdot \underline{V}^T \cdot \underline{D}^T \cdot \underline{s} \quad (20'')$$

$$\delta u' = \frac{du'}{du_i} = \underline{e}_i^T \cdot \underline{V}^T \cdot \underline{D}^T \cdot \underline{s}_z \cdot \frac{2}{s} \quad (21'')$$

Eine weitere systematische Vorgehensweise, analog zu der in Abschnitt 2.2 verfolgten zeigt, daß sich die um die Exzentrizität der Balkenachse zur Berechnungsebene des Gesamtsystems erweiterte Steifigkeitsmatrix aus einer erneuten Links- bzw. Rechtsmultiplikation in diesem Fall jedoch mit der transponierten Versatzmatrix \underline{V}^T bzw. mit \underline{V} selbst ergibt:

$$\underline{K}_{eD}'' = \underline{V}^T \cdot \underline{K}_{eD}' \cdot \underline{V} \quad (31)$$

Die anschließende Durchführung der Berechnung erfolgt mit dem Programmsystem DERIVE. Zur vollständigen Darstellung wird nochmals von der Dehnsteifigkeitsmatrix \underline{K}_{eD} des eindimensionalen Stabelementes ausgegangen und aus dieser mit Hilfe der Gleichungen (27) und (31) die Dehnsteifigkeitsmatrix des verwendeten Stabelementes unter Berücksichtigung der Transformation in die Grundrißebene des Decken-Balken-Systems und der Exzentrizität berechnet.

DERIVE-Protokoll:

$$\underline{K}_{eD}(EA, s) := \begin{pmatrix} / & 1 & -1 & \backslash \\ & & & \\ \backslash & -1 & 1 & / \end{pmatrix} \cdot \frac{EA}{s}$$

$$\underline{D}(\alpha) := \begin{pmatrix} / & \cos(\alpha) & \sin(\alpha) & 0 & 0 & \backslash \\ & & & & & \\ \backslash & 0 & 0 & \cos(\alpha) & \sin(\alpha) & / \end{pmatrix}$$

$$\underline{V}(e) := \begin{pmatrix} / & 1 & 0 & 0 & 0 & -e & 0 & 0 & 0 & 0 & 0 & \backslash \\ & & & & & & & & & & & \\ \backslash & 0 & 1 & 0 & e & 0 & 0 & 0 & 0 & 0 & 0 & / \\ & & & & & & & & & & & \\ \backslash & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -e & / \\ & & & & & & & & & & & \\ \backslash & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & e & 0 & / \end{pmatrix}$$

$$\underline{K}_{eD}''(EA, e, s, \alpha) := \underline{V}^T \cdot \underline{D}^T \cdot \underline{K}_{eD} \cdot \underline{D} \cdot \underline{V} = \tag{32}$$

$$\begin{pmatrix} / & \cos(\alpha)^2 & \sin(\alpha) \cos(\alpha) & 0 & e \sin(\alpha) \cos(\alpha) & -e \cos(\alpha)^2 & -\cos(\alpha)^2 & -\sin(\alpha) \cos(\alpha) & 0 & -e \sin(\alpha) \cos(\alpha) & e \cos(\alpha)^2 & \backslash \\ & & & & & & & & & & & \\ \backslash & \sin(\alpha) \cos(\alpha) & \sin(\alpha)^2 & 0 & e \sin(\alpha)^2 & -e \sin(\alpha) \cos(\alpha) & -\sin(\alpha) \cos(\alpha) & -\sin(\alpha)^2 & 0 & -e \sin(\alpha)^2 & e \sin(\alpha) \cos(\alpha) & / \\ & & & & & & & & & & & \\ \backslash & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & / \\ & & & & & & & & & & & \\ \backslash & e \sin(\alpha) \cos(\alpha) & e \sin(\alpha)^2 & 0 & e^2 \sin(\alpha)^2 & -e^2 \sin(\alpha) \cos(\alpha) & -e \sin(\alpha) \cos(\alpha) & -e \sin(\alpha)^2 & 0 & -e^2 \sin(\alpha)^2 & e^2 \sin(\alpha) \cos(\alpha) & / \\ & & & & & & & & & & & \\ \backslash & -e \cos(\alpha)^2 & -e \sin(\alpha) \cos(\alpha) & 0 & -e^2 \sin(\alpha) \cos(\alpha) & e^2 \cos(\alpha)^2 & e \cos(\alpha)^2 & e \sin(\alpha) \cos(\alpha) & 0 & e^2 \sin(\alpha) \cos(\alpha) & -e^2 \cos(\alpha)^2 & / \\ & & & & & & & & & & & \\ \backslash & -\cos(\alpha)^2 & -\sin(\alpha) \cos(\alpha) & 0 & -e \sin(\alpha) \cos(\alpha) & e \cos(\alpha)^2 & \cos(\alpha)^2 & \sin(\alpha) \cos(\alpha) & 0 & e \sin(\alpha) \cos(\alpha) & -e \cos(\alpha)^2 & / \\ & & & & & & & & & & & \\ \backslash & -\sin(\alpha) \cos(\alpha) & -\sin(\alpha)^2 & 0 & -e \sin(\alpha)^2 & e \sin(\alpha) \cos(\alpha) & \sin(\alpha) \cos(\alpha) & \sin(\alpha)^2 & 0 & e \sin(\alpha)^2 & -e \sin(\alpha) \cos(\alpha) & / \\ & & & & & & & & & & & \\ \backslash & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & / \\ & & & & & & & & & & & \\ \backslash & -e \sin(\alpha) \cos(\alpha) & -e \sin(\alpha)^2 & 0 & -e^2 \sin(\alpha)^2 & e^2 \sin(\alpha) \cos(\alpha) & e \sin(\alpha) \cos(\alpha) & e \sin(\alpha)^2 & 0 & e^2 \sin(\alpha)^2 & -e^2 \sin(\alpha) \cos(\alpha) & / \\ & & & & & & & & & & & \\ \backslash & e \cos(\alpha)^2 & e \sin(\alpha) \cos(\alpha) & 0 & e^2 \sin(\alpha) \cos(\alpha) & -e^2 \cos(\alpha)^2 & -e \cos(\alpha)^2 & -e \sin(\alpha) \cos(\alpha) & 0 & -e^2 \sin(\alpha) \cos(\alpha) & e^2 \cos(\alpha)^2 & / \end{pmatrix} \begin{matrix} EA \\ S \end{matrix}$$

3. Biegebalken

3.1 Ausgangselement

Nach der Diskretisierung durch finite Elemente kann die Systemgleichung (15) in der Form:

$$\sum_e \int_s \delta w'' EI w'' dx - \sum_e \int_s \delta w p dx = 0 \quad (33)$$

geschrieben werden.

Um diese analog zu Abschnitt 2.1 auswerten zu können werden zunächst wieder Ansätze für die Geometrie und die Physik im lokalen, normierten Koordinatensystem gewählt. Der Ansatz für die Geometrie ist dabei mit der Gleichung (17) identisch. Für die Physik ist allerdings ein höherwertiger Ansatz notwendig. An den Elementknoten des Biegebalkens stehen mit w und $\varphi = w'$ jeweils 2 Freiwerte, insgesamt somit 4 Freiwerte zur Verfügung. Demnach ist mindestens ein kubischer Ansatz notwendig.

Geometrie:

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} X_{1A} & X_{1B} \\ X_{2A} & X_{2B} \end{pmatrix} \cdot \begin{pmatrix} \frac{1}{2}(1-z) \\ \frac{1}{2}(1+z) \end{pmatrix} \quad (17)$$

$$\vec{x} = \underline{X} \cdot \vec{s}$$

Physik:

$$w(z) = (w_A, \varphi_A, w_B, \varphi_B) \cdot \begin{pmatrix} \frac{1}{4}(z^3 - 3z + 2) \\ \frac{s}{8}(z^3 - z^2 - z + 1) \\ -\frac{1}{4}(z^3 - 3z - 2) \\ \frac{s}{8}(z^3 + z^2 - z - 1) \end{pmatrix}$$

$$w(z) = \vec{u}_e^T \cdot \vec{s} \quad (34)$$

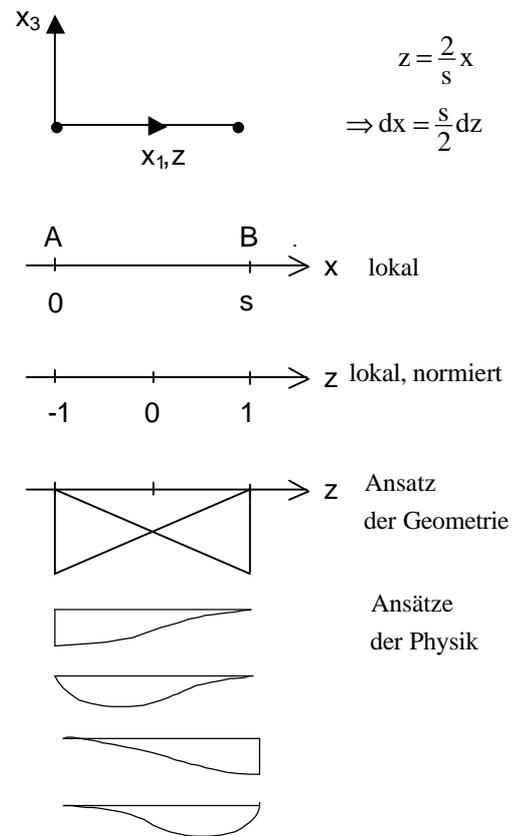


Bild 9: Koordinatensysteme und Formfunktionen

Die Herleitung des Formvektors \vec{s} kann dem DERIVE-Protokoll der nachfolgenden Seite entnommen werden.

Bestimmung der Ableitungen:

$$w' = \frac{dw}{dz} \frac{dz}{dx} = \vec{u}_e^T \cdot \vec{s}_z \cdot \frac{2}{s} \quad (35)$$

$$w'' = \frac{dw'}{dz} \frac{dz}{dx} = \vec{u}_e^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} \quad (36)$$

$$\delta w = \frac{dw}{du_i} = \vec{e}_i^T \cdot \vec{s} \quad (37)$$

$$\delta w'' = \frac{dw''}{du_i} = \vec{e}_i^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} \quad (38)$$

$$\vec{s}_{zz} = \begin{pmatrix} \frac{3}{2}z \\ \frac{s}{4}(3z-1) \\ -\frac{3}{2}z \\ \frac{s}{4}(3z+1) \end{pmatrix}$$

Wobei $\frac{d}{du_i}$ die Ableitung nach dem i-ten Element von \vec{u}_e und somit nach

w_A, φ_A, w_B oder φ_B bedeutet.

Einsetzen der Ansätze in das erste Integral von (33):

$$\int_s \delta w'' EI w'' dx = \int_s \vec{e}_i^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} \cdot EI \cdot \vec{u}_e^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} dx$$

$$= \vec{e}_i^T \cdot \frac{16}{s^4} \cdot EI \int_{-1}^1 \vec{s}_{zz} \cdot \vec{s}_{zz}^T \cdot \vec{u}_e \cdot \frac{s}{2} dz = \vec{e}_i^T \cdot \underbrace{\frac{8EI}{s^3} \int_{-1}^1 \vec{s}_{zz} \cdot \vec{s}_{zz}^T dz}_{\underline{K}_{eB}} \cdot \vec{u}_e \quad (39)$$

Die Matrizenmultiplikation sowie die Integration werden wiederum mit DERIVE durchgeführt. Ebenfalls ist die Herleitung der Formfunktionen gezeigt. Zu beachten ist dabei der Umrechnungsfaktor $s/2$ für die Ableitungen, der den Zusammenhang zwischen normiertem und nicht normiertem lokalem Koordinatensystem erfaßt.

DERIVE-Protokoll:

$$w(z) := a_0 + a_1 z + a_2 z^2 + a_3 z^3$$

$$w'(z) := 3 a_3 z^2 + 2 a_2 z + a_1$$

$$\left\{ \begin{array}{l} p(-1) = w_A, p(1) = w_B, p'(-1) = \frac{\varphi_A s}{2}, p'(1) = \frac{\varphi_B s}{2} \end{array} \right\}$$

$$\left\{ \begin{array}{l} a_0 := \frac{\varphi_A s - \varphi_B s + 4 w_B + 4 w_A}{8}, a_1 := -\frac{\varphi_A s + \varphi_B s + 6 (w_A - w_B)}{8}, \\ a_2 := \frac{s (\varphi_B - \varphi_A)}{8}, a_3 := \frac{\varphi_A s + \varphi_B s + 2 (w_A - w_B)}{8} \end{array} \right\}$$

$$w(z) = \frac{w_A (z^3 - 3z + 2)}{4} + \frac{s \varphi_A (z^3 - z^2 - z + 1)}{8} - \frac{w_B (z^3 - 3z - 2)}{4} + \frac{s \varphi_B (z^3 + z^2 - z - 1)}{8}$$

$$\frac{d^2}{dz^2} w(z) = \frac{3z w_A}{2} + \frac{s \phi_A (3z - 1)}{4} - \frac{3z w_B}{2} + \frac{s \phi_B (3z + 1)}{4}$$

$$s_{zz} := \begin{pmatrix} \frac{3z}{2} \\ s(3z - 1) \\ \frac{3z}{2} \\ s(3z + 1) \end{pmatrix}$$

$$\underline{M} := s_{zz} \cdot s_{zz}^T$$

$$\underline{M} := \begin{pmatrix} \frac{9z^2}{4} & \frac{3sz(3z-1)}{8} & \frac{9z^2}{4} & \frac{3sz(3z+1)}{8} \\ \frac{3sz(3z-1)}{8} & \frac{s^2(3z-1)^2}{16} & \frac{3sz(1-3z)}{8} & \frac{s^2(3z+1)(3z-1)}{16} \\ \frac{9z^2}{4} & \frac{3sz(1-3z)}{8} & \frac{9z^2}{4} & \frac{3sz(3z+1)}{8} \\ \frac{3sz(3z+1)}{8} & \frac{s^2(3z+1)(3z-1)}{16} & \frac{3sz(3z+1)}{8} & \frac{s^2(3z+1)^2}{16} \end{pmatrix}$$

$$\underline{MM} := \int_{-1}^1 \underline{M} dz$$

$$\underline{MM} := \begin{pmatrix} \frac{3}{2} & \frac{3s}{4} & \frac{3}{2} & \frac{3s}{4} \\ \frac{3s}{4} & \frac{s^2}{2} & \frac{3s}{4} & \frac{s^2}{4} \\ \frac{3}{2} & \frac{3s}{4} & \frac{3}{2} & \frac{3s}{4} \\ \frac{3s}{4} & \frac{s^2}{4} & \frac{3s}{4} & \frac{s^2}{2} \end{pmatrix}$$

$$\underline{K}_{eB}(EI, s) := \frac{8EI}{3s} \underline{MM}$$

$$\underline{K}_{eB} = \frac{EI}{3s} \begin{pmatrix} 12 & 6s & -12 & 6s \\ 6s & 4s^2 & -6s & 2s^2 \\ -12 & -6s & 12 & -6s \\ 6s & 2s^2 & -6s & 4s^2 \end{pmatrix} \quad (40)$$

3.2 Transformation in die Grundrißebene

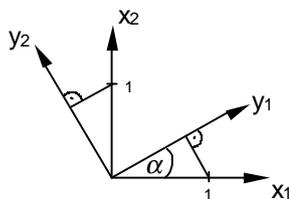
Analog zum Dehnstab ist auch beim Biegebalken für die gewünschte Anwendung eine Transformation des Grundelementes in die Grundrißebene notwendig. Dabei wird zunächst davon ausgegangen, daß die x_1 -Achse des globalen Koordinatensystems Parallel zur Stabachse verläuft.

Mit $\varphi = u_{3,1} = -\alpha_2$ ergibt sich die Zuordnung der Knotenfreiwerte zwischen globalem und lokalem System in trivialer Weise zu:

$$\begin{pmatrix} w_A \\ \varphi_A \\ w_B \\ \varphi_B \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} u_{1A} \\ u_{2A} \\ u_{3A} \\ \alpha_{1A\parallel} \\ \alpha_{2A\parallel} \\ u_{1B} \\ u_{2B} \\ u_{3B} \\ \alpha_{1B\parallel} \\ \alpha_{2B\parallel} \end{pmatrix} \quad (41)$$

$$\vec{u}_e = \underline{T} \cdot \vec{u}_{e\parallel} \quad (42)$$

Ist nun die Stabachse um einen Winkel α gegen die x_1 -Achse des globalen Systems geneigt, so muß der Stützwertevektor um die x_3 -Achse rotiert werden. Da die Freiwerte u_1 und u_2 beim Biegebalken entfallen (siehe Gl. (41)) betrifft die Rotation nur die beiden Verdrehungen α_1 und α_2 .



Basisvektoren des globalen Systems beschrieben im lokalen System:

$$\vec{x}_1 = \begin{pmatrix} \cos(\alpha) \\ -\sin(\alpha) \\ 0 \end{pmatrix} \quad \vec{x}_2 = \begin{pmatrix} \sin(\alpha) \\ \cos(\alpha) \\ 0 \end{pmatrix} \quad \vec{x}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Bild 10: Rotation des Koordinatensystems

Damit kann die Drehung wie folgt beschrieben werden:

$$\vec{u}_{e\parallel} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & \vdots & & & \vdots & & \vdots & & & \\ 0 & 0 & 0 & \cos(\alpha) & \sin(\alpha) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\sin(\alpha) & \cos(\alpha) & 0 & 0 & 0 & 0 & 0 \\ & \vdots & & & \vdots & & \vdots & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cos(\alpha) & \sin(\alpha) & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\sin(\alpha) & \cos(\alpha) & \end{pmatrix} \cdot \begin{pmatrix} u_{1A} \\ u_{2A} \\ u_{3A} \\ \alpha_{1A} \\ \alpha_{2A} \\ u_{1B} \\ u_{2B} \\ u_{3B} \\ \alpha_{1B} \\ \alpha_{2B} \end{pmatrix}$$

$$\vec{u}_{e||} = \underline{\mathbf{R}} \cdot \vec{u}_{se} \quad (43)$$

Die vollständige Transformation ist somit:

$$\vec{u}_e = \underline{\mathbf{T}} \cdot \underline{\mathbf{R}} \cdot \vec{u}_{se} \quad (44)$$

Alle nachfolgenden Ausführungen ergeben sich analog zu Abschnitt 2.2. Einsetzen von (44) in (34)-(38) liefert:

$$w(z) = \vec{u}_e^T \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \vec{s} \quad (34')$$

$$w'' = \frac{dw'}{dz} \frac{dz}{dx} = \vec{u}_e^T \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} \quad (36')$$

$$\delta w = \frac{dw}{du_i} = \vec{e}_i^T \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \vec{s} \quad (37')$$

$$\delta w'' = \frac{dw''}{du_i} = \vec{e}_i^T \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} \quad (38')$$

Einsetzen von (45) und (34')-(38') in (39):

$$\begin{aligned} \int_s \delta w'' EI w'' dx &= \int_s \vec{e}_i^T \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} \cdot EI \cdot \vec{u}_e^T \cdot \vec{s}_{zz} \cdot \frac{4}{s^2} dx \\ &= \vec{e}_i^T \cdot \frac{16}{s^4} \cdot EI \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \int_{-1}^1 \vec{s}_{zz} \cdot \vec{s}_{zz}^T \cdot \vec{u}_e \cdot \frac{s}{2} dz = \vec{e}_i^T \cdot \frac{8EI}{s^3} \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \int_{-1}^1 \vec{s}_{zz} \cdot \vec{s}_{zz}^T dz \cdot \vec{u}_e \\ &= \vec{e}_i^T \cdot \underbrace{\frac{8EI}{s^3} \cdot \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \int_{-1}^1 \vec{s}_{zz} \cdot \vec{s}_{zz}^T dz \cdot \underline{\mathbf{T}} \cdot \underline{\mathbf{R}}}_{\underline{\mathbf{K}}'_{eB}} \cdot \vec{u}_{se} \end{aligned}$$

Somit gilt:

$$\boxed{\underline{\mathbf{K}}'_{eB} = \underline{\mathbf{R}}^T \cdot \underline{\mathbf{T}}^T \cdot \underline{\mathbf{K}}_{eB} \cdot \underline{\mathbf{T}} \cdot \underline{\mathbf{R}}} \quad (45)$$

Die exzentrische Lage des Balkens bei der Anordnung als Steg eines Plattenbalkenquerschnittes bezüglich der Berechnungsebene hat beim Biegebalken keinen Einfluß auf die Gestalt der Elementmatrix, da der Biegebalken die Freiwerte u_1 und u_2 nicht enthält. Die Freiwerte u_3, α_1, α_2 werden von der Exzentrizität nicht beeinflusst.

DERIVE-Protokoll:

$$\underline{K}_{\text{eb}}(\text{EI}, s) := \frac{\text{EI}}{s^3} \begin{pmatrix} / 12 & 6s & -12 & 6s \backslash \\ 6s & 4s^2 & -6s & 2s^2 \\ -12 & -6s & 12 & -6s \\ 6s & 2s^2 & -6s & 4s^2 \end{pmatrix} \quad (40)$$

$$\underline{T} := \begin{pmatrix} / 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \backslash \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \backslash 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

$$\underline{R}(\alpha) := \begin{pmatrix} / 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \backslash \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \text{COS}(\alpha) & \text{SIN}(\alpha) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\text{SIN}(\alpha) & \text{COS}(\alpha) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \text{COS}(\alpha) & \text{SIN}(\alpha) \\ \backslash 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\text{SIN}(\alpha) & \text{COS}(\alpha) \end{pmatrix}$$

$$\underline{K}'_{\text{eb}}(\text{EI}, s, \alpha) := \underline{R}(\alpha)^T \cdot \underline{T}^T \cdot \underline{K}_{\text{eb}}(\text{EI}, s) \cdot \underline{T} \cdot \underline{R}(\alpha)$$

$$\underline{K}'_{\text{eb}} = \begin{pmatrix} / 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \backslash \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{12 \text{ EI}}{s^3} & \frac{6 \text{ EI SIN}(\alpha)}{s^2} & -\frac{6 \text{ EI COS}(\alpha)}{s^2} & 0 & 0 & -\frac{12 \text{ EI}}{s^3} & \frac{6 \text{ EI SIN}(\alpha)}{s^2} & -\frac{6 \text{ EI COS}(\alpha)}{s^2} \\ 0 & 0 & \frac{6 \text{ EI SIN}(\alpha)}{s^2} & \frac{4 \text{ EI SIN}(\alpha)^2}{s} & -\frac{4 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} & 0 & 0 & -\frac{6 \text{ EI SIN}(\alpha)}{s^2} & \frac{2 \text{ EI SIN}(\alpha)^2}{s} & -\frac{2 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} \\ 0 & 0 & \frac{6 \text{ EI COS}(\alpha)}{s^2} & -\frac{4 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} & \frac{4 \text{ EI COS}(\alpha)^2}{s} & 0 & 0 & \frac{6 \text{ EI COS}(\alpha)}{s^2} & -\frac{2 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} & \frac{2 \text{ EI COS}(\alpha)^2}{s} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{12 \text{ EI}}{s^3} & -\frac{6 \text{ EI SIN}(\alpha)}{s^2} & \frac{6 \text{ EI COS}(\alpha)}{s^2} & 0 & 0 & \frac{12 \text{ EI}}{s^3} & -\frac{6 \text{ EI SIN}(\alpha)}{s^2} & \frac{6 \text{ EI COS}(\alpha)}{s^2} \\ 0 & 0 & \frac{6 \text{ EI SIN}(\alpha)}{s^2} & \frac{2 \text{ EI SIN}(\alpha)^2}{s} & -\frac{2 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} & 0 & 0 & -\frac{6 \text{ EI SIN}(\alpha)}{s^2} & \frac{4 \text{ EI SIN}(\alpha)^2}{s} & -\frac{4 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} \\ 0 & 0 & \frac{6 \text{ EI COS}(\alpha)}{s^2} & -\frac{2 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} & \frac{2 \text{ EI COS}(\alpha)^2}{s} & 0 & 0 & \frac{6 \text{ EI COS}(\alpha)}{s^2} & -\frac{4 \text{ EI SIN}(\alpha) \text{COS}(\alpha)}{s} & \frac{4 \text{ EI COS}(\alpha)^2}{s} \end{pmatrix} \quad (46)$$

Zur leichteren Erfassung der Struktur von \underline{K}'_{eB} kann dies noch verkürzt geschrieben werden:

$$\underline{K}'_{eB} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a & b & c & 0 & 0 & -a & b & c \\ 0 & 0 & b & d & e & 0 & 0 & -b & \frac{d}{2} & \frac{e}{2} \\ 0 & 0 & c & e & f & 0 & 0 & -c & \frac{e}{2} & \frac{f}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -a & -b & -c & 0 & 0 & a & -b & -c \\ 0 & 0 & b & \frac{d}{2} & \frac{e}{2} & 0 & 0 & -b & d & e \\ 0 & 0 & c & \frac{e}{2} & \frac{f}{2} & 0 & 0 & -c & e & f \end{pmatrix} \quad (47)$$

mit

$$a = \frac{12 EI}{s^3} \quad b = \frac{6 EI \sin(\alpha)}{s^2} \quad c = -\frac{6 EI \cos(\alpha)}{s^2}$$

$$d = \frac{4 EI \sin(\alpha)^2}{s} \quad e = -\frac{4 EI \sin(\alpha) \cos(\alpha)}{s} \quad f = \frac{4 EI \cos(\alpha)^2}{s}$$

4. Gesamtelementmatrix

Wie bereits anfangs erwähnt, soll der Steg eines Plattenbalkenquerschnittes im Rahmen einer FEM-gestützten Tragwerksanalyse durch ein exzentrisch angeschlossenes Stabelement abgebildet werden.

Dieses Stabelement soll alle wesentlichen Tragwirkungen (Dehnung und Biegung) erfassen. Weiterhin soll eine beliebige Anordnung im Grundriß des Deckensystems möglich sein.

Zu beachten ist, daß die Berechnungsebene des Gesamtsystem in der Plattenmittelfläche und somit außerhalb des Balkenquerschnittes liegt.

Mit Hilfe der in den Abschnitten 2. und 3. hergeleiteten Elementmatrizen läßt sich mittels einfacher Summation die Elementmatrix eines Stabelementes berechnen, die all die genannten Einflüsse berücksichtigt.

$$\underline{K}_e = \underline{K}''_{eD} + \underline{K}'_{eB} \quad (48)$$

Es sei ausdrücklich darauf hingewiesen, daß die so gewonnene Elementmatrix im Bezug zu einem allg. üblichen mathematischen Koordinaten System steht. Dies bedeutet insbesondere, daß die x_3 -Achse nach oben und nicht wie in der klassischen Baustatik nach unten weist. Hieraus ergeben sich weitere Konsequenzen hinsichtlich der Nachlaufrechnung zur Bestimmung der Schnittgrößen, da diese zur gewohnten Weiterverarbeitung im klassischen System benötigt werden. Die Matrix kann dem nachfolgendem Protokoll entnommen werden.

DERIVE-Protokoll:

$$\underline{K}_e(EA, EI, e, s, \alpha) := \underline{K}_{ed}(EA, e, s, \alpha) + \underline{K}_{es}(EI, s, \alpha) =$$

$$\left(\begin{array}{cccccccc}
 \frac{EA \cos(\alpha)}{B} & \frac{EA \sin(\alpha) \cos(\alpha)}{B} & 0 & \frac{e EA \cos(\alpha)}{B} & \frac{EA \sin(\alpha) \cos(\alpha)}{B} & 0 & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{e EA \cos(\alpha)}{B} \\
 \frac{EA \sin(\alpha) \cos(\alpha)}{B} & \frac{EA \sin(\alpha)}{B} & 0 & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{EA \sin(\alpha) \cos(\alpha)}{B} & \frac{EA \sin(\alpha)}{B} & \frac{e EA \sin(\alpha)}{B} & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} \\
 0 & 0 & \frac{12 EI}{B} & \frac{6 EI \sin(\alpha)}{B} & \frac{6 EI \cos(\alpha)}{B} & 0 & \frac{12 EI}{B} & \frac{6 EI \cos(\alpha)}{B} \\
 \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{e EA \sin(\alpha)}{B} & \frac{6 EI \sin(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{e EA \sin(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA + 2 EI}{B} \sin(\alpha) \cos(\alpha) \\
 \frac{e EA \cos(\alpha)}{B} & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{6 EI \cos(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{e EA \cos(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA + 2 EI}{B} \cos(\alpha) \\
 \frac{EA \cos(\alpha)}{B} & \frac{EA \sin(\alpha) \cos(\alpha)}{B} & 0 & \frac{e EA \cos(\alpha)}{B} & \frac{EA \sin(\alpha) \cos(\alpha)}{B} & 0 & \frac{e EA \cos(\alpha)}{B} & \frac{e EA \cos(\alpha)}{B} \\
 \frac{EA \sin(\alpha) \cos(\alpha)}{B} & \frac{EA \sin(\alpha)}{B} & 0 & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{EA \sin(\alpha) \cos(\alpha)}{B} & \frac{EA \sin(\alpha)}{B} & \frac{e EA \sin(\alpha)}{B} & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} \\
 0 & 0 & \frac{12 EI}{B} & \frac{6 EI \cos(\alpha)}{B} & \frac{6 EI \sin(\alpha)}{B} & 0 & \frac{12 EI}{B} & \frac{6 EI \sin(\alpha)}{B} \\
 \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{e EA \sin(\alpha)}{B} & \frac{6 EI \sin(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{e EA \sin(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA + 2 EI}{B} \sin(\alpha) \cos(\alpha) \\
 \frac{e EA \cos(\alpha)}{B} & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{6 EI \cos(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA \sin(\alpha) \cos(\alpha)}{B} & \frac{e EA \cos(\alpha)}{B} & \frac{e EA + 4 EI}{B} \sin(\alpha) \cos(\alpha) & \frac{e EA + 2 EI}{B} \cos(\alpha)
 \end{array} \right) \quad (49)$$

5. Verwendete Koordinatensysteme

Alle verwendeten Koordinatensysteme sind karthesische Rechtshandsysteme. Ihre Orientierung wird so gewählt, daß jeweils die x_1 - x_2 -Ebene in der Mittelfläche der Platte liegt. Die x_3 -Achse weist nach oben und somit zu der, von der Plattenmittelebene aus betrachtet, dem Unterzug abgewandten Seite. Somit unterscheiden sich die Koordinatensysteme nur um den jeweiligen Drehwinkel α um die x_3 -Achse.

5.1. Globales Koordinatensystem (Typ 1)

Das globale Koordinatensystem verläuft parallel zu den Bildschirmachsen und ist das Referenzkoordinatensystem der FEM-Berechnung. Es ist in der linken unteren Ecke des Darstellungsbereiches angeordnet. Seine x_1 -Achse verläuft von links nach rechts, die x_2 -Achse von unten nach oben. Alle Elementmatrizen, die eingepprägten Lasten, sowie die Beschreibung der Geometrie, insbesondere die Beschreibung der Lage der Knoten, beziehen sich auf dieses Koordinatensystem. Auch die Darstellung der Ergebnisse für die Weg- und Scheiben-/Plattenkraftgrößen orientiert sich an diesem Koordinatensystem.

5.2. Lokales Koordinatensystem der Lager (Typ 2)

Werden an bestimmten Knoten Randbedingungen eingepragt, die nicht nur die vertikale Verschiebung betreffen, so soll diese Einprägung in einem lokalen Koordinatensystem erfolgen. Dies ist insbesondere bei Linienlagern, wie sie sich beispielsweise bei der Auflagerung der Decke auf eine Wand ergeben sinnvoll, um die Verdrehungen und Verschiebungen parallel bzw. senkrecht zur Wand auch dann sperren zu können, wenn die Wand nicht parallel zu den globalen Koordinatenachsen verläuft.

Die Orientierung dieser Koordinatensysteme wird im allg. so gewählt, daß die x_1 -Achse jeweils senkrecht zur Wand verläuft.

5.3. Lokales Koordinatensystem des Unterzuges (Typ 3)

Dieses Koordinatensystem wird zur Berechnung der Balkenschnittgrößen des Unterzuges verwendet. Seine x_1 -Achse verläuft längs des Unterzuges.

Beachte: Dies ist nicht exakt das Koordinatensystem, in dem die Schnittkräfte des Balkens visualisiert werden. (siehe dazu Abschnitt 6.)

5.4. Berechnungsablauf

Vor der Berechnung liegen folgende Modellelemente vor:

- Knotenkoordinaten im globalen System
- Knotenfreiwerte im globalen System, bei Lagerung ggf. im System 2
- Lager im lokalen System
- Lasten im globalen System
- Elementmatrizen von Platte, Scheibe sowie exzentrischem Stab im globalen System

Um für die Berechnung konforme Koordinatensysteme zu erhalten werden bei Aufbau der Systemgleichung folgende Transformationen vorgenommen:

- wenn an einem Knoten ein Koordinatensystem vom Typ 2 vorhanden ist:
 - Transformation des betreffenden Lastvektorenanteils in das Koordinatensystem 2
 - Transformation der betreffenden Elementmatrizenanteile der Stab-, Scheiben- und Plattenelemente in das System 2

Somit bestimmt das Vorhandensein eines lokalen Systems vom Typ 2 an einem Knoten das verwendete Berechnungssystem in der Systemgleichung.

System 2 vorhanden \Rightarrow alle Berechnungsgrößen des betreffenden Knotens befinden sich im System 2

System 2 nicht vorhanden \Rightarrow alle Berechnungsgrößen des betreffenden Knotens befinden sich im globalen System

Nach Lösen der Systemgleichung liegen die Ergebnisse in dem jeweiligen am Knoten vorhandenen Koordinatensystem vor.

Da jedoch der größte Teil der Berechnungsergebnisse bezüglich des globalen Systems visualisiert werden soll, werden die Knotenfreiwerte dort, wo sie nicht ohnehin schon im globalen System liegen, beim zurückspeichern wieder in das globale System zurücktransformiert.

Die Nachlaufrechnung zur Berechnung der Scheiben- und Plattenkraftgrößen kann sich somit unmittelbar auf die abgespeicherten Weggrößen stützen.

Für die Berechnung der Stabkraftgrößen ist noch eine Transformation in das Koordinatensystem vom Typ 3 notwendig, um die lokalen Schnittgrößen berechnen zu können. Dies geschieht aber temporär während der Nachlaufrechnung und beeinflusst nicht den vorrätigen Datenbestand.

5.5. Durchführung der Transformationen

Die Transformation erfolgt prinzipiell analog zu der unter Gl.(43), nur daß bis auf u_3 alle Freiwerte transformiert werden. Die Teilmatrix \underline{I}_R ist der wesentliche Teil der Rotationsmatrix, wobei der Index R die Richtung der Rotation anzeigt:

GL = Transformation vom globalen Koordinatensystem in das lokale

LG = Transformation vom lokalen Koordinatensystem in das globale

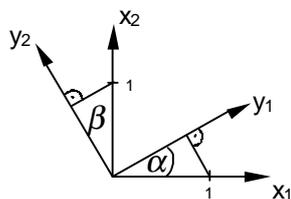
Es bestehen folgende Zusammenhänge:

$$\underline{\bar{u}}_e = \begin{pmatrix} \underline{T}_R & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \underline{T}_R & 0 & 0 & 0 \\ 0 & 0 & 0 & \underline{T}_R & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \underline{T}_R \end{pmatrix} \cdot \begin{pmatrix} u_{1A} \\ u_{2A} \\ u_{3A} \\ \alpha_{1A} \\ \alpha_{2A} \\ u_{1B} \\ u_{2B} \\ u_{3B} \\ \alpha_{1B} \\ \alpha_{2B} \end{pmatrix} \quad (50)$$

$$\underline{T}_{LG} = \underline{T}_{GL}^{-1} \quad (51)$$

$$\underline{T}_{GL} = \begin{pmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{pmatrix} \quad (52)$$

Im allgemeinen Fall der affinen Transformation, wie sie bei zwei nicht rechtwinklig aufeinandertreffenden Linienlagern vorkommt :



$$\underline{T}_{GL} = \begin{pmatrix} \cos(\alpha) & \sin(\beta) \\ -\sin(\alpha) & \cos(\beta) \end{pmatrix} \quad (53)$$

Bild 11: Zusammenhang bei affinen Transformationen

6. Nachlaufrechnung

Ziel der Nachlaufrechnung ist die Bestimmung der lokalen Balkenschnittkräfte. Zur Berechnung soll der Einfachheit halber die Steifigkeitsmatrix des eindimensionalen Stabes genutzt werden. Diese setzt sich in einfacher Weise aus den beiden schon hergeleiteten Teilmatrizen für die Dehnung (Gl.(23)) und Biegung (Gl.(40)) zusammen:

$$\underline{K}_{e1D} = \begin{pmatrix} \frac{EA}{s} & 0 & 0 & -\frac{EA}{s} & 0 & 0 \\ 0 & 12\frac{EI}{s^3} & 6\frac{EI}{s^2} & 0 & -12\frac{EI}{s^3} & 6\frac{EI}{s^2} \\ 0 & 6\frac{EI}{s^2} & 4\frac{EI}{s} & 0 & -6\frac{EI}{s^2} & 2\frac{EI}{s} \\ -\frac{EA}{s} & 0 & 0 & \frac{EA}{s} & 0 & 0 \\ 0 & -12\frac{EI}{s^3} & -6\frac{EI}{s^2} & 0 & 12\frac{EI}{s^3} & -6\frac{EI}{s^2} \\ 0 & 6\frac{EI}{s^2} & 2\frac{EI}{s} & 0 & -6\frac{EI}{s^2} & 4\frac{EI}{s} \end{pmatrix} \quad (54)$$

Wie bereits erwähnt werden zunächst die vorgehaltenen Berechnungsergebnisse der Verschiebungsgrößen vom globalen in das lokale Koordinatensystem parallel zur Balkenachse transformiert.

Die so erhaltenen Verschiebungsgrößen sind diejenigen der Knoten in der Mittelebene der Platte. Die Verschiebungen, der vor der Belastung direkt darunter auf der Stabachse liegenden Punkte, ergeben sich aus den lokalen Weggrößen in der Plattenebene wie folgt:

$$\begin{pmatrix} u \\ w \\ \varphi \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & -e \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} \quad (55) \quad \text{Siehe dazu auch Erläuterungen zu Gl.(29) und Gl.(41)}$$

Die Freiwerte u_2 und α_1 können einfach entfallen, da sie aufgrund der vorher ausgeführten Transformation senkrecht zur vom Stabelement erfaßten Tragrichtung wirken.

Zur Ermittlung der Schnittkräfte wird nun das aus dem System freigeschnittene Stabelement betrachtet und ihm die, mit Hilfe der Transformation und der Gl.(55) aus der Gesamtberechnung erhaltenen, Weggrößen eingeprägt. Es ergibt sich somit eine „Miniatursystemgleichung“ dessen dualer Vektor die Lagerkräfte enthält, welche den gesuchten Schnittkräften entsprechen.

Da der Lastvektor dabei als Nullvektor anzusetzen ist, ergeben sich die Schnittkräfte in einfacher Weise durch ein Matrizen-Vektor-Multiplikation der Art:

$$\boxed{\vec{k}_m = \underline{K}_{eID} \cdot \vec{u}_{e \text{ Balken}}} \quad (56)$$

$$\text{mit } \vec{k}_m^T = (N_A, Q_A, M_A, N_B, Q_B, M_B)$$

Wobei der Index m kennzeichnen soll, daß die so erhaltenen Schnittgrößen im mathematischen Koordinatensystem zu interpretieren sind.

Um eine ingenieurmäßige Interpretation der Ergebnisse zu erleichtern, soll nun eine Umrechnung in das klassische Koordinatensystem der Baustatik erfolgen.

Da die Orientierung der x_1 -Achse und der Winkel in beiden Koordinatensystemen gleich ist, betreffen die Änderungen nur die Kräfte in Richtung der x_3 -Achse. Hier muß eine Vorzeichenumkehr vorgenommen werden.

Die Kraftgrößen am ersten Knoten des Stabelementes befinden sich an einem linken Schnittufer und besitzen daher ein anderes Vorzeichen. Um nun auch diese anzupassen werden die Vorzeichen der betreffenden Elemente umgekehrt.

Die so errechneten Schnittgrößen sind im klassischen Sinne zu interpretieren:

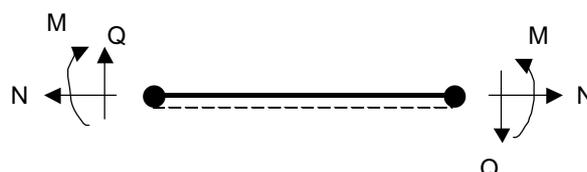


Bild 12: Vorzeichenkonvention im klassischen Koordinatensystem

Kapitel C	Implementierung des Stabelementes
-----------	-----------------------------------

1. Interface

Um eine Austauschbarkeit des Stabelementes zu ermöglichen, wird ein Interface namens `IcEccentricBeam` zur Verfügung gestellt, welches wie alle Interfaces des Programmsystems zum Paket `iface` gehört. Alle programminternen Zugriffe auf das Stabelement erfolgen über dieses Interface, welches die zur Verwendung des Stabelementes benötigten Methoden enthält:

```
package iface      ;
import  service.* ;

public interface IcEccentricBeam

....set().....

{   void  setNodeId(String k1, String k2) ;
    // Sets the identifiers of the nodes of the element.

    void  setNodeCoordinates(int k,double x1,double x2) ;
    // Sets the coordinates of node k of the element to x1,x2.

    void  setMaterialId(String materialId) ;
    // Sets the identifier of the material of the element.

    void  setHeight(double value) ;
    // Sets the height of the beam to value.

    void  setWidth(double value) ;
    // Sets the width of the beam to value.

    void  setThicknessOfPlate(double value) ;
    // Sets the thickness of the plate to value.

    void  setReferences() throws FemException ;
    // Replaces all object identifiers by object references.

....get().....

    int  getNumberOfNodes() ;
    // Returns the number of nodes in the element.

    String[]  getNodeId() ;
    // Returns the identifiers of the nodes of the element.

    IcNode[]  getNodeReferences() ;
    // Returns the refernece array of the nodes in the element.

    String  getMaterialId() ;
    // Returns the identifier of the material of the element.

    IcMaterial  getMaterialReference() ;
    // Returns the reference to the material object

    double  getHeight() ;
    // Returns the height of the element.

    double  getWidth() ;
    // Returns the width of the element.
```

```
double[] getNodeCoordinates(int index) ;
    // Returns the coordinates(x1,x2) of node with localindex.

double[][] getNodeCoordinates() ;
    // Returns the coordinates of all nodes of the element.

int [] getSystemIndices() ;
    // Returns the system indices of the nodal variables in the element
    // displacement vector

double[] getglobalDisplacementVector();
    // Returns the displacement vector of the beam containing the
    // nodal displacement vectors in the order of the node indices
    // in the global coordinate system, read from the current
    // displacements at the nodes

double[] getlokalDisplacementVector();
    // Return the displacement vector in the lokal coordinate system

//....compute().....

double[][] computeStiffnessMatrix() ;
    // Computes the element stiffness matrix.

double[][] rotateStiffnessMatrix() ;
    // Rotates the element stiffness matrix to the system vector us

//....computeForces in lokal System.....

double[] computeForces();
    // Computes the forces at both points

....print().....

void printProperties() ;
    // Prints the node and the material identifiers of the element :
    // format to agree with the heading printed in class model

void printDisplacements();
    // Prints the id of the Element and all its displacements
    // The Koordinatesystem dependance on which method is called before
    // getglobalDisplacementVector() -> global system
    // getlokalDisplacementVector() -> lokal system

void printForces();
    // Prints the forces at both points
}
```

2. Elementklasse

Die Klasse des Stabelementes selbst heißt `EccentricBeam` und ist Teil des Paketes `standard`.

Ihre Umsetzung erfolgte insgesamt sehr nah an der hergeleiteten Theorie. `EccentricBeam` implementiert alle vom Interface gelisteten Methoden und stellt folgenden Konstruktor zur Verfügung, der das Setzen der Systemkennwerte sowie die Registrierung bei der Klasse `Model` des Paketes `plate` übernimmt:

```
EccentricBeam(String id, String k1, String k2,  
              String materialId, double h, double b, double p)
```

id	Elementname	h	Höhe des Unterzuges
k1	Name von Knoten A	b	Breite des Unterzuges
k2	Name von Knoten B	p	Höhe der Platte
materialId	Name des Baustoffs		

Der wesentliche Teil der Klasse ist die Methode zur Erzeugung der Elementmatrix und zur Berechnung der Balkenschnittgrößen.

Die Elementmatrix wird wie unter Abschnitt 4. des Kapitels B beschrieben aus den beiden Teilmatrizen der Dehnung (Gl.(32)) und der Biegung (Gl.(47)) zusammengesetzt (Gl.(48)). Zuvor werden noch alle benötigten Werte (EI, EA, e, l, α) aus der Geometrie und den gesetzten Systemwerten berechnet.

Es werden zweckmäßige Abkürzungen eingeführt:

```
public double[][] computeStiffnessMatrix()  
{  
    double EI, EA, e, e2, l, c, s, c2, s2, sc, alpha, dx, dy, f ;  
  
    getNodeCoordinates() ;  
    EI = material.getModulusOfElasticity() ;  
    EI *= b*h*h*h/12. ;  
    EA = material.getModulusOfElasticity() ;  
    EA *= h*b ;  
    e = (plateH + h)/2. ;  
    e2 = e*e ;  
    dx = coor[1][0]-coor[0][0] ;  
    dy = coor[1][1]-coor[0][1] ;  
    alpha= Math.atan(dy/dx) ;  
    if(dx<0. && dy>0.) alpha += Math.PI ;  
    if(dx<0. && dy<0.) alpha += Math.PI ;  
    l = Math.sqrt(dx*dx+dy*dy) ;  
    c = Math.cos(alpha) ;  
    s = Math.sin(alpha) ;  
    sc = s*c ;  
    c2 = c*c ;  
    s2 = s*s ;  
    f = EA/l ;  
}
```

```
//Stiffnessmatrix of beam stretching

double stretch[][] =

{{
    c2*f,   sc*f,   0.,   e*sc*f,   -e*c2*f,   -c2*f,   -sc*f,   0.,   -e*sc*f,   e*c2*f },
 {
    sc*f,   s2*f,   0.,   e*s2*f,   -e*sc*f,   -sc*f,   -s2*f,   0.,   -e*s2*f,   e*sc*f },
 {
    0.,     0.,     0.,     0.,     0.,     0.,     0.,     0.,     0.,     0. },
 {
    e*sc*f,   e*s2*f,   0.,   e2*s2*f,   -e2*sc*f,   -e*sc*f,   -e*s2*f,   0.,   -e2*s2*f,   e2*sc*f },
 {
    -e*c2*f,   -e*sc*f,   0.,   -e2*sc*f,   e2*c2*f,   e*c2*f,   e*sc*f,   0.,   e2*sc*f,   -e2*c2*f },
 {
    -c2*f,   -sc*f,   0.,   -e*sc*f,   e*c2*f,   c2*f,   sc*f,   0.,   e*sc*f,   -e*c2*f },
 {
    -sc*f,   -s2*f,   0.,   -e*s2*f,   e*sc*f,   sc*f,   s2*f,   0.,   e*s2*f,   -e*sc*f },
 {
    0.,     0.,     0.,     0.,     0.,     0.,     0.,     0.,     0.,     0. },
 {
    -e*sc*f,   -e*s2*f,   0.,   -e2*s2*f,   e2*sc*f,   e*sc*f,   e*s2*f,   0.,   e2*s2*f,   -e2*sc*f },
 {
    e*c2*f,   e*sc*f,   0.,   e2*sc*f,   -e2*c2*f,   -e*c2*f,   -e*sc*f,   0.,   -e2*sc*f,   e2*c2*f }};

matrix = stretch ;

//Stiffnessmatrix of beam bending

double A,B,C,D,E,F ;
A = 12.*EI/(l*1*1) ;
B = 6.*EI*s/(l*1) ;
C = -6.*EI*c/(l*1) ;
D = 4.*EI*s2/l ;
E = -4.*EI*sc/l ;
F = 4.*EI*c2/l ;

matrix[2][2] += A ; matrix[2][3] +=B ; matrix[2][4] +=C ; matrix[2][7] +=-A ; matrix[2][8] +=B ; matrix[2][9] +=C ;
matrix[3][2] += B ; matrix[3][3] +=D ; matrix[3][4] +=E ; matrix[3][7] +=-B ; matrix[3][8] +=D/2.; matrix[3][9] +=E/2.;
matrix[4][2] += C ; matrix[4][3] +=E ; matrix[4][4] +=F ; matrix[4][7] +=-C ; matrix[4][8] +=E/2.; matrix[4][9] +=F/2.;

matrix[7][2] +=-A ; matrix[7][3] +=-B ; matrix[7][4] +=-C ; matrix[7][7] += A ; matrix[7][8] +=-B ; matrix[7][9] +=-C ;
matrix[8][2] += B ; matrix[8][3] +=D/2.; matrix[8][4] +=E/2.; matrix[8][7] +=-B ; matrix[8][8] += D ; matrix[8][9] += E ;
matrix[9][2] += C ; matrix[9][3] +=E/2.; matrix[9][4] +=F/2.; matrix[9][7] +=-C ; matrix[9][8] += E ; matrix[9][9] += F ;

return matrix ;
}
```

Die Berechnung der Balkenschnittgrößen erfolgt über eine Nachlaufrechnung streng nach Abschnitt 6. des Kapitels B:

```
public double[] computeForces()
{
    double ei,ea,e,s,s2,s3,dx,dy ;

    getNodeCoordinates() ;
    ei = material.getModulusOfElasticity() ;
    ei *= b*h*h*h/12. ;
    ea = material.getModulusOfElasticity() ;
    ea *= h*b ;
    e = (plateH + h)/2. ;
    dx = coor[1][0]-coor[0][0] ;
    dy = coor[1][1]-coor[0][1] ;
    s = Math.sqrt(dx*dx+dy*dy) ;
    s2 = s*s ;
    s3 = s*s*s ;

    double m[][]=
    {{
        ea/s , 0. , 0. , -ea/s, 0. , 0. },
     {
        0. , 12.*ei/s3, 6*ei/s2, 0. , -12*ei/s3, 6*ei/s2 },
     {
        0. , 6.*ei/s2, 4*ei/s, 0. , - 6*ei/s2, 2*ei/s },
     {
        -ea/s , 0. , 0. , ea/s, 0. , 0. },
     {
        0. , -12.*ei/s3, -6*ei/s2, 0. , 12*ei/s3, - 6*ei/s2 },
     {
        0. , 6.*ei/s2, 2*ei/s , 0. , - 6*ei/s2, 4*ei/s }};

    double ul[] = getlokalDisplacementVector() ;
    double u[] = new double[6] ;
}
```

```
int counter = 0 ;
for(int j=0; j<2; j++)
{
    u[counter]= ul[j*5]-e*ul[j*5+4] ;
    counter++ ;
    u[counter]= ul[j*5+2] ;
    counter++ ;
    u[counter]=-ul[j*5+4] ;
    counter++ ;
}
Algebra.matrixVectorProduct(f,m,u) ;

f[1] *= -1 ;
f[4] *= -1 ;

for(int z=0; z<3; z++) f[z] *= -1.

return f ;
}
```

Wobei die Transformation der globalen Koordinaten in das Koordinatensystem des Typs 3 von der Methode `getlocalDisplacementVector()` übernommen wird:

```
public double[] getlokalDisplacementVector()
{
    double c,s,alpha,dx,dy ;

    getNodeCoordinates() ;
    dx = coor[1][0]-coor[0][0] ;
    dy = coor[1][1]-coor[0][1] ;
    alpha= Math.atan(dy/dx) ;
    if(dx<0. && dy>0.) alpha += Math.PI ;
    if(dx<0. && dy<0.) alpha += Math.PI ;
    c = Math.cos(alpha) ;
    s = Math.sin(alpha) ;

    int i,k,counter ;
    double u[],rot[][] ;
    counter = 0 ;
    for (i = 0; i < 2; i++)
    { u = node[i].getNodalDisplacements() ;
      double c1,c2,c3,c4 ;
      c1 = u[0]* c + u[1]* s ;
      c2 = u[0]*(-s) + u[1]* c ;
      c3 = u[3]* c + u[4]* s ;
      c4 = u[3]*(-s) + u[4]* c ;
      nodeDisplacement[counter ] = c1 ;
      nodeDisplacement[counter+1] = c2 ;
      nodeDisplacement[counter+2] = u[2] ;
      nodeDisplacement[counter+3] = c3 ;
      nodeDisplacement[counter+4] = c4 ;
      counter += 5 ;
    }
    return nodeDisplacement ;
}
```

An den Knoten des Deckenrandes werden zur Einprägung der Randbedingungen der Scheibe/Platte senkrecht bzw. parallel zum Rand lokale Koordinatensysteme vom Typ 2 verwendet. Um diese Randbedingungen auch für das Stabelement nutzen zu können, muß wie bereits in Kapitel B Abschnitt 5 beschrieben, an den betreffenden Knoten der zugeordnete Teil der Elementmatrix in das lokale System gedreht werden.

Die Rotation erfolgt vollkommen analog zu Gl.(45) bzw. Gl.(50). Der wesentliche Teil \underline{I}_R der Transformationsmatrix \underline{I} wird dabei von der Methode `getRotationMatrix()` des lokalen Koordinatensystems zur Verfügung gestellt:

```
public double[][] rotateStiffnessMatrix()
{ int      i,index,k,m,n                                ;
  double  c1,c2,c3,c4,rot[][] ,mat[][]                ;
  IcLocal local                                        ;
  mat = matrix                                        ;

  //Rt*mat
  for (index = 0; index < 2; index++)
  { if (node[index].hasLocalSystem())
    { local = node[index].getLocalCoordinateSystem()    ;
      rot  = local.getRotationMatrix()                  ;
      k    = 5 * index                                  ;
      for(i = 0; i < 10; i++)
      {
        c1 = rot[0][0] * mat[k][i]  + rot[1][0] * mat[k+1][i] ;
        c2 = rot[0][1] * mat[k][i]  + rot[1][1] * mat[k+1][i] ;
        c3 = rot[0][0] * mat[k+3][i] + rot[1][0] * mat[k+4][i] ;
        c4 = rot[0][1] * mat[k+3][i] + rot[1][1] * mat[k+4][i] ;
        mat[k][i]    = c1                                ;
        mat[k+1][i]  = c2                                ;
        mat[k+3][i]  = c3                                ;
        mat[k+4][i]  = c4                                ;
      }
    }
  }

  //(Rt*mat)*R
  for (index = 0; index < 2; index++)
  { if (node[index].hasLocalSystem())
    { local = node[index].getLocalCoordinateSystem()    ;
      rot  = local.getRotationMatrix()                  ;
      k    = 5 * index                                  ;
      for(i = 0; i < 10; i++)
      {
        c1 = rot[0][0] * mat[i][k]   + rot[1][0] * mat[i][k+1] ;
        c2 = rot[0][1] * mat[i][k]   + rot[1][1] * mat[i][k+1] ;
        c3 = rot[0][0] * mat[i][k+3] + rot[1][0] * mat[i][k+4] ;
        c4 = rot[0][1] * mat[i][k+3] + rot[1][1] * mat[i][k+4] ;
        mat[i][k]    = c1                                ;
        mat[i][k+1]  = c2                                ;
        mat[i][k+3]  = c3                                ;
        mat[i][k+4]  = c4                                ;
      }
    }
  }

  return mat ;
}
```

3. Berechnung auf Systemebene

Der Zusammenbau der algebraischen Systemgleichung geschieht durch Aufruf der Methode `computeSystemMatrix()` der Klasse `Analysis` im Paket `plate`. Diese Methode wurde so erweitert, daß sie die Elementmatrizen der im System verwendeten Stabelemente mit erfaßt. Dahinter steht eine einfache Addition der Steifigkeitsanteile von Scheibe/Platte und Stab zur Systemsteifigkeitsmatrix des System:

```
public void computeSystemMatrix() throws FemException
{
    :

    iter = model.elementSet.iterator() ;
    while (iter.hasNext())
    { element = (IcElement)iter.next() ;
      index   = element.getSystemIndices() ;
      element.computeStiffnessMatrix() ;
      elementMatrix = element.rotateStiffnessMatrix() ;
      system.addMatrix(index,elementMatrix) ;
    }

    iter = model.beamSet.iterator() ;
    while (iter.hasNext())
    { beam    = (IcEccentricBeam)iter.next() ;
      index   = beam.getSystemIndices() ;
      beam.computeStiffnessMatrix() ;
      elementMatrix = beam.rotateStiffnessMatrix() ;
      system.addMatrix(index,elementMatrix) ;
    }
}
```

Die Aufstellung des Lastvektors bleibt wie schon in Kapitel B Abschnitt 1. erwähnt vom Prinzip her unverändert. Es wird weiterhin der Lastvektor der Platte verwendet.

Um Lasten jedoch immer im globalen System einprägen zu können wird der Lastvektor beim Vorhandensein eines lokalen Koordinatensystems am Lastknoten in das lokale System gedreht. Dies war im vorgegebenen Programm nicht notwendig, da nur vertikale Plattenlasten vorgesehen waren, deren Richtung in beiden Koordinatensystemen gleich ist. Nötig wird die Drehung jedoch bei eingepprägten Stab- oder Scheibenlängskräften, die zu Testzwecken benötigt werden.

Die Drehung wird von der Methode `rotateLoadvector()` der Klasse `NodeLoad` vollzogen:

```
private double[] rotateLoadvector()
{
    if(node.hasLocalSystem())
    {
        double rot[][] =
node.getLocalCoordinateSystem().getInverseRotationMatrix();
        double localforce[] = new double[5] ;
        double c1,c2,c3,c4 ;
    }
}
```

```
//rot*localforce
    c1 = rot[0][0] * force[0] + rot[0][1] * force[1] ;
    c2 = rot[1][0] * force[0] + rot[1][1] * force[1] ;
    c3 = rot[0][0] * force[3] + rot[0][1] * force[4] ;
    c4 = rot[1][0] * force[3] + rot[1][1] * force[4] ;
    localforce[0] = c1 ;
    localforce[1] = c2 ;
    localforce[2] = force[2] ;
    localforce[3] = c3 ;
    localforce[4] = c4 ;
    return localforce ;
}
return force;
}
```

Nach dem Lösen der Systemgleichung sind alle Knotenverschiebungsfreiwerte unter Berücksichtigung des Einflusses des Unterzuges auf das Tragverhalten des Deckensystems bekannt. Sie liegen in den jeweiligen Koordinatensystemen der Knoten vor, d.h. bei Randknoten im lokalen, bei allen anderen Knoten im globalen System. Um die globale Darstellung der Scheiben-/Plattenschnittkräfte zu ermöglichen werden beim Zurückspeichern alle Verschiebungsergebnisse in das globale System gedreht. Die Reaktionskräfte in den Lagern verbleiben im knoteneigenen System:

```
public void saveResult()
{ double primal[] = system.getPrimal() ;
  double dual[] = system.getDual() ;
  iter = model.nodeSet.iterator() ;
  while (iter.hasNext())
  { node = (IcNode)iter.next() ;
    index = node.getSystemIndices() ;
    node.setNodalDisplacements(primal,index[0]) ;
    node.invertRotation() ;
  }

  iter = model.supportSet.iterator() ;
  while (iter.hasNext())
  { support = (IcSupport)iter.next() ;
    node = support.getNodeReference() ;
    index = node.getSystemIndices() ;
    support.setReactions(dual,index) ;
  } }
```

Die so erhaltenen Verschiebungsgrößen werden bis zu einer erneuten Berechnung im Programmsystem vorgehalten. Alle weiteren Berechnungen zur Bestimmung der Kraftgrößen greifen auf diese Weggrößen zu, verändern sie jedoch nicht. Die weitere Drehung der Weggrößen in ein lokales System zur Bestimmung der Balkenschnittgrößen ist, wie bereits gezeigt, so angelegt, daß sie keinen Einfluß auf den Datenbestand hat.

1. Test des Stabelementes

1.1. Handrechnungen

Nachfolgend werden die mit dem nach Kapitel B und C implementierten Stabelement erzielten Ergebnisse auf ihre Richtigkeit hin untersucht. Besonderes Augenmerk wird dabei auf die Richtungsunabhängigkeit der lokalen Rechenergebnisse bezüglich der Lage der Stabachse gelegt. Dazu werden die lokalen und globalen Verschiebungsgrößen auf der Shell ausgegeben. Die Kraftgrößen werden nur im lokalen System ermittelt.

Um absolut sichere Aussagen treffen zu können werden zunächst die beiden Traganteile Dehnung und Biegung getrennt von einander getestet. Dies geschieht durch Herausnehmen der jeweilig nicht betrachteten Teilmatrix in der Methode `computeStiffnessMatrix()` der Klasse `EccentricBeam`.

Es wird ein System aus zwei Stabelementen mit einer jeweiligen Länge von 1m verwendet. Die Querschnittsabmaße werden mit $h=0,5\text{m}$ und $b=0,4\text{m}$ gewählt. Die Plattenhöhe wird zu $p=0,3\text{m}$ und der Elastizitätsmodul zu $E=30000\text{KN/m}^2$ gewählt. Die Lagerung ergibt sich je nach zu untersuchender Tragwirkung.

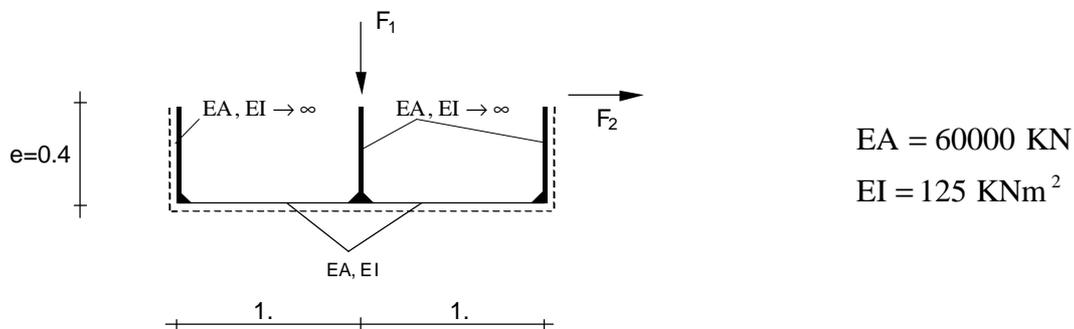


Bild 13: verwendetes Testbeispiel

1.1.1. Dehnanteil

Um den Dehnanteil getrennt von der Biegung testen zu können werden an den drei Vertikalstäben Einspannungen angeordnet, zwei davon horizontal verschieblich.

Um eine Singularität der Systemmatrix zu verhindern müssen außerdem die Verschiebungen senkrecht zur Zeichenebene, sowie alle Torsionsverdrehungen gesperrt werden. Als Belastung wird $F_1=0$ und $F_2=100 \text{ KN}$ gewählt.

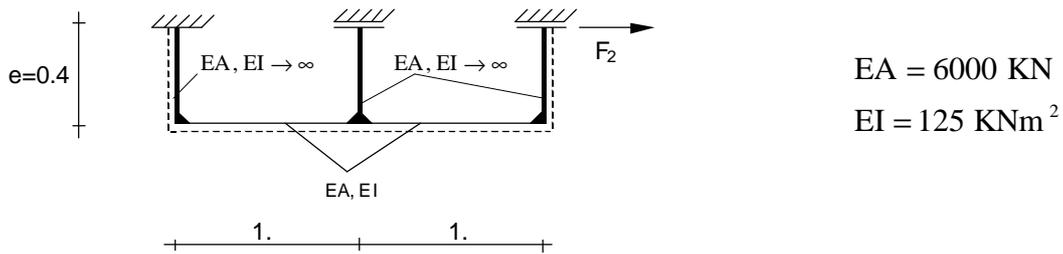


Bild 14: Lagerung und Belastung zum Test des Dehnanteils

Das zu erwartende Ergebnis am Lastangriffspunkt ist somit:

$$N = +100 \text{ KN}$$

lokal :

$$\sigma = E \cdot \varepsilon \Leftrightarrow \varepsilon = \frac{\Delta l}{l} = \frac{\sigma}{E} = \frac{N}{EA} \Rightarrow u = \Delta l = \frac{N}{EA} \cdot l = \frac{100}{6000} \cdot 2 = 0,0\bar{3}$$

global :

	u_1	u_2
0°	$0,0\bar{3}$	0
30°	$0,0\bar{3} \cdot \cos(30^\circ) = 0,02887$	$0,0\bar{3} \cdot \sin(30^\circ) = 0,01\bar{6}$
45°	$0,0\bar{3} / \sqrt{2} = 0,02357$	0,02357
90°	0	$0,0\bar{3}$
180°	$-0,0\bar{3}$	0
-45°	0,02357	$-0,02357$

1.1.2. Biegeanteil

Es wird ein beidseitig eingespannter Balken konstruiert. Als Belastung wird $F_1=100 \text{ KN}$ und $F_2=0$ gewählt. Alle Torsionsverdrehungen sowie die Verschiebungen senkrecht zur Zeichenebene werden gesperrt.

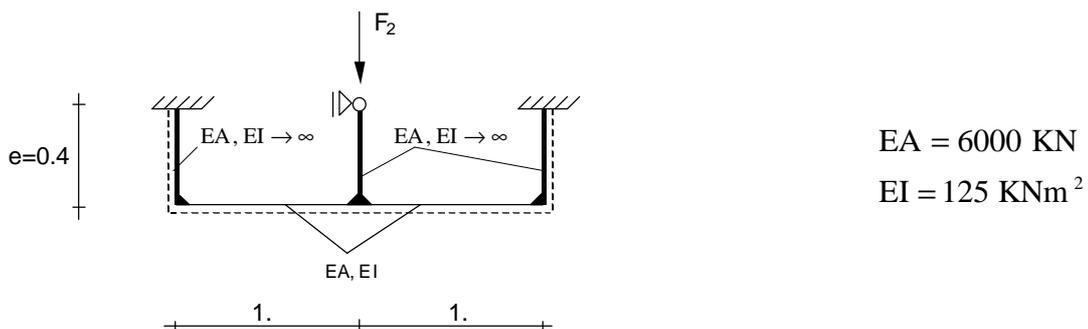


Bild 15: Lagerung und Belastung zum Test des Biegeanteils

Es werden folgende Schnitt- und Weggrößen erwartet:

PvK:

$$w = 4 \cdot \frac{1}{125} \cdot \frac{1}{3} \cdot 0,5 \cdot 25 \cdot 0,25 = 0,0\bar{3}$$

$$\varphi = 0 \text{ (an den Knoten)}$$

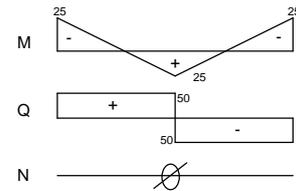


Bild 16: erwarteter Schnittgrößenverlauf

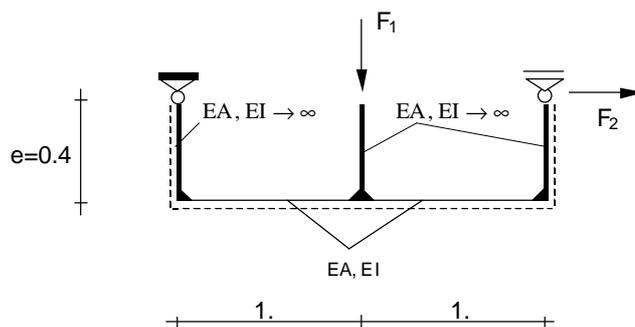
Die lokalen Weggrößen entsprechen den globalen. (für φ nur hier wegen $\varphi = 0$)

1.1.3. Gesamttragverhalten

Nun werden beide Tragverhalten kombiniert und die volle Elementsteifigkeitsmatrix verwendet.

LF1: $F_1=100$ $F_2=0$

LF2: $F_1=0$ $F_2=100$



$$EA = 6000 \text{ KN}$$

$$EI = 125 \text{ KNm}^2$$

Bild 17: System zum Test des Gesamttragverhaltens

Die zu erwartenden Ergebnisse wurden mit Schnittprinzip und PvK ermittelt. Es werden jeweils die Werte w , M , Q und N unterhalb der Last F_1 sowie der Wert u bei der Last F_2 angegeben.

LF1: $u = -0,16$ $w = 0,1\bar{3}$ $M = +50$ $Q_1 = +50$ $Q_r = -50$ $N = 0$

LF2: $u = 0,289\bar{3}$ $w = -0,16$ $M = -40$ $Q = 0$ $N = +100$

1.2. Programmresultate

Zum Erzeugen der Testbeispiele wurde in der Klasse Generator des Paketes plate eine Methode constructTestBeam() vorgesehen.

Es werden dort zwei Stabelemente, ein lokales Koordinatensystem, drei Lagerungen sowie die jeweilige Last erzeugt. Die Richtung der Stabachse sowie die Höhe der Lasten wird über die Oberfläche gesteuert:

```
public void constructTestBeam()
{
    double alpha = angle                ;
    double rad   = (Math.PI*alpha)/180.0 ;

    model.add(new Local("lokalsys",    alpha)) ;

    model.add (new Node("Knoten_1", 0,0.,0.)) ;
    model.add (new Node("Knoten_2", 5,Math.cos(rad),Math.sin(rad))) ;
    model.add (new Node("Knoten_3",10,2*Math.cos(rad),2*Math.sin(rad))) ;
    IcNode node;
        node = (IcNode)App.getObject("Knoten_1") ;
        node.setLocalCoordinateSystem("lokalsys") ;
        node = (IcNode)App.getObject("Knoten_2") ;
        node.setLocalCoordinateSystem("lokalsys") ;
        node = (IcNode)App.getObject("Knoten_3") ;
        node.setLocalCoordinateSystem("lokalsys") ;
    model.add(new EccentricBeam("Balken_A","Knoten_1","Knoten_2",
                                "iso", beamHeight, beamWidth, thickness)) ;
    model.add(new EccentricBeam("Balken_B","Knoten_2","Knoten_3",
                                "iso", beamHeight, beamWidth, thickness)) ;

    boolean state1[] = { true , true, true, true, false };
    boolean state2[] = { false, true, true, true, false };
    boolean eben [] = { false, true, false, true, false };
    double value[] = { 0.0, 0.0, 0.0, 0.0, 0.0 } ;

    model.add(new Support("Lager_1","Knoten_1",state1,value)) ;
    model.add(new Support("Lager_2","Knoten_3",state2,value)) ;
    model.add(new Support("Ebenesproblem","Knoten_2",eben,value));

    model.add(new NodeLoad("Mittellast","Knoten_2",0.0,0.0,uniformLoad,
                            0.0,0.0)) ;
    model.add(new NodeLoad("Endlast","Knoten_3",
                            pointLoad*Math.cos(rad),pointLoad*Math.sin(rad),0.0,0.0,0.0)) ;
}
```

Für die Ausgabe aller benötigter Werte wurde eine Methode `balkenTest()` zur alphanumerischen Ausgabe benutzt. Sie befindet sich in der Klasse `Control` des Paketes `graphik`.

Beim Vergleich der Ergebnisse ist darauf zu achten, daß die Handrechnung im klassischen System der Baustatik erfolgte, wohingegen die Verschiebungsergebnisse des FEM-Programms im mathematischen Koordinatensystem vorliegen. (Gleiches gilt für das Vorzeichen der eingepprägten Last)

Bei allen unter Abschnitt 3.1 Angegebenen Testbeispielen ergaben sich Sowohl für die Weg- als auch für die Kraftgrößen exakt die erwarteten Ergebnisse. Diese zeigten sich desweiteren absolut richtungsunabhängig, so daß von einer richtigen Implementierung gesprochen werden kann.

2. Test der Membranwirkung

2.1. Handrechnung

Es wurden zwei Beispielhandrechnungen durchgeführt.

2.1.1. Kragsscheibe unter Zugeinwirkung

Es wird eine Kragsscheibe unter Zugeinwirkung infolge einer angreifenden Streckenlast betrachtet, die aus vier Elementen zusammengesetzt wird. Da der LF Streckenlast nicht implementiert wurde, wird die Streckenlast auf die Knoten aufgeteilt:

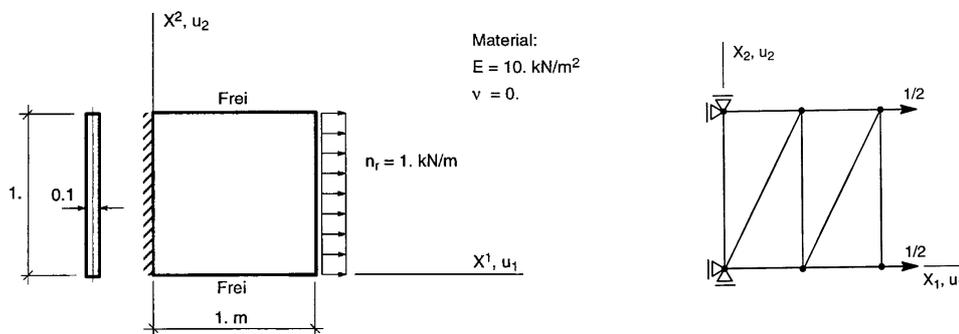


Bild 18: Kragsscheibe unter Zugeinwirkung (System und Elementierung)

Zu erwarten ist ein konstanter Kraftgrößen- sowie ein linearer Verschiebungsverlauf der Art:

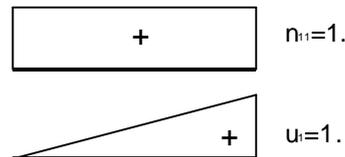
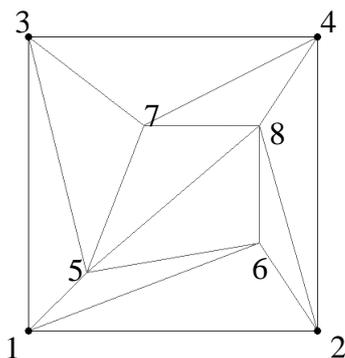


Bild 19: zu erwartender Kraftgrößen- und Verschiebungsverlauf

Alle anderen Größen müssen sich aufgrund der Wahl von $\nu = 0$ identisch Null ergeben.

2.1.2. Scheibe mit eingprägter Verzerrung



Es wird folgende Scheibe betrachtet:

Knoten	1	2	3	4	5	6	7	8
x_1	0.	10.	0.	10.	2.	8.	4.	8.
x_2	0.	0.	10.	10.	2.	3.	7.	7.

Tabelle 1: Knotenkoordinaten

Die eingprägten Verschiebungen betragen:

$$u_1(4)=u_2(4)= 1., \quad u_1(3)= 1., \quad u_2(2)= 1.$$

alle anderen Verschiebungen werden zu Null gesetzt

Bild 20: Elementierung der Scheibe

Die Weggrößen verteilen sich somit linear von Punkt 1 aus. u_1 nimmt parallel zur Kante 1-2, u_2 parallel zu Kante 1-3 zu.

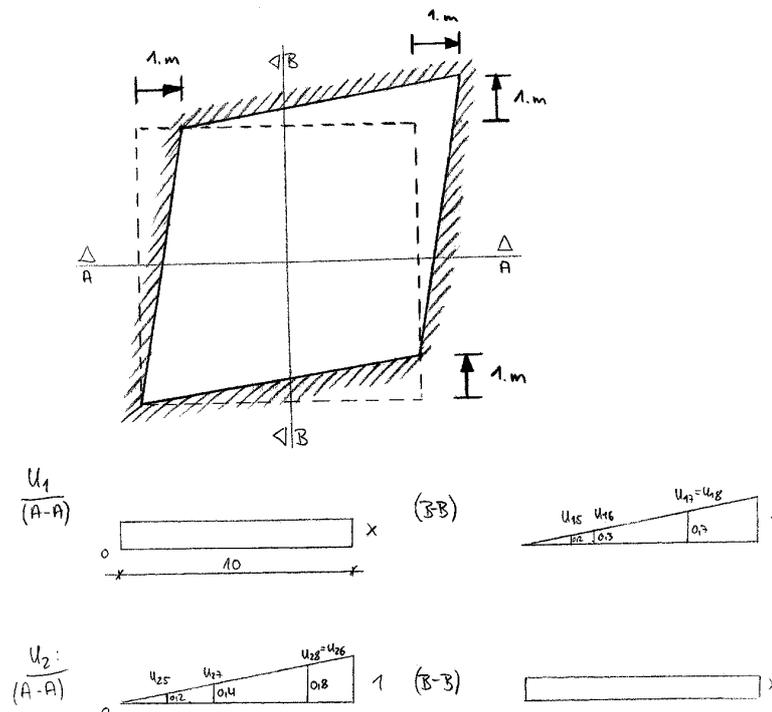


Bild 21: Verformungsbild der betrachteten Scheibe

Für die Dehnungen ergibt sich somit:

$$\varepsilon_{11} = \frac{\partial u_1}{\partial x_1} = 0 \quad ; \quad \varepsilon_{22} = \frac{\partial u_2}{\partial x_2} = 0 \quad ; \quad \varepsilon_{12} = \frac{\partial u_1}{\partial x_2} + \frac{\partial u_2}{\partial x_1} = \frac{1}{10} + \frac{1}{10} = 0,2$$

Die zu erwartenden Kraftgrößen werden wie folgt berechnet:

$$\#1: \begin{bmatrix} n_{11} \\ n_{22} \\ n_{12} \end{bmatrix} = \frac{E \cdot t}{1 - \nu} \cdot \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1 - \nu}{2} \end{bmatrix} \cdot \begin{bmatrix} \alpha_{11} \\ \alpha_{22} \\ \alpha_{12} \end{bmatrix}$$

#2: [E := 30000, ν := 0.3, t := 0.3]

#3: [α_{11} := 0, α_{22} := 0, α_{12} := 0.2]

#4: [n11 = 0, n22 = 0, n12 = 692.307]

Auf die gleiche Art und Weise können weitere einfache Verschiebungslastfälle konstruiert werden.

2.2. Programmresultate

Im Programmpaket werden die Beispiele analog zu der in Abschnitt 1.3 gezeigten Methode erzeugt. Die Ergebnisse können unmittelbar mit Hilfe der dargestellten Isolinien überprüft werden. Es stellen sich die erwarteten Ergebnisse ein.

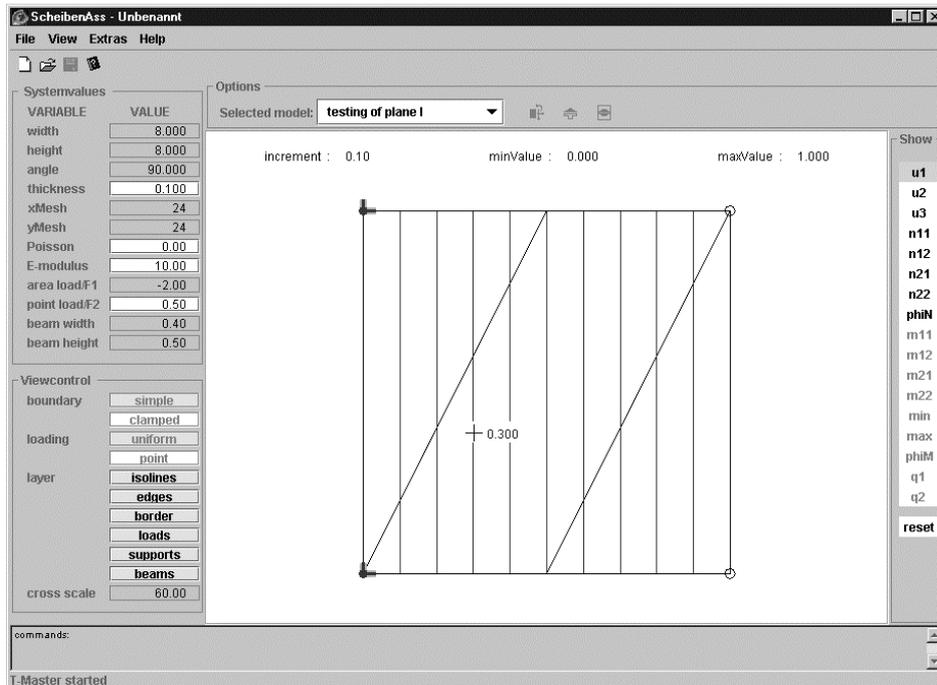


Bild 23: u_1 in Bsp. 2.1.1.

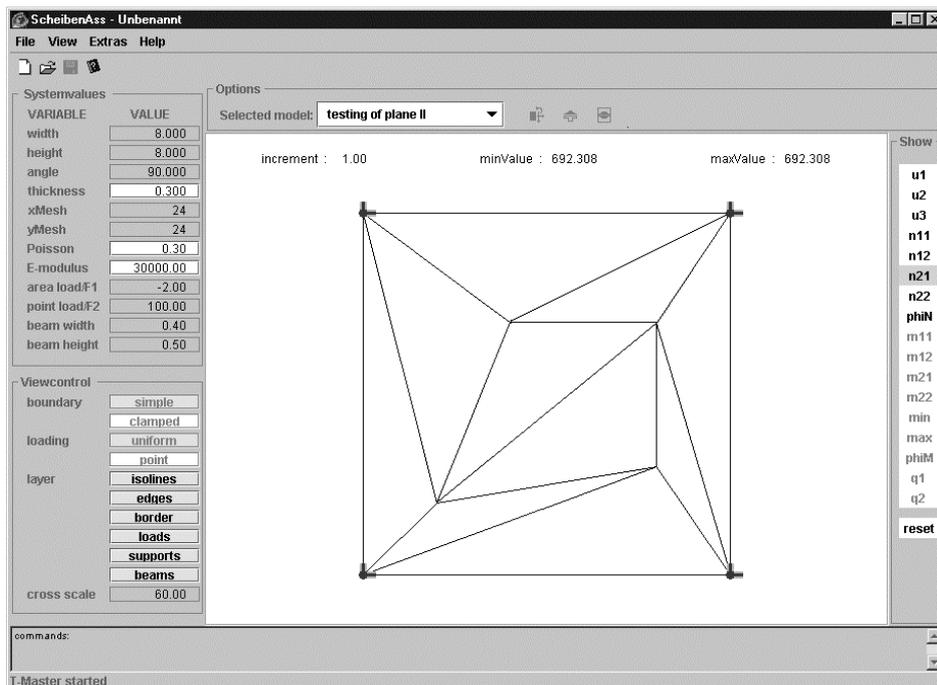


Bild 22: n_{21} in Bsp. 2.1.2

3. Test der Plattentragwirkung

Obwohl keine Veränderungen am Plattentraganteil vorgenommen wurden ist es sinnvoll nach so umfangreichen Arbeiten am Programmpaket (durch die Anhebung der Anzahl der Freiwerte wurde ein Großteil aller vorhandenen Klassen geändert) zu überprüfen, ob die alte Funktionalität uneingeschränkt erhalten geblieben ist.

Dazu wurden Vergleichsrechnungen an einfachen Deckensystemen mit dem Ursprungsprogramm bei nicht Vorhandensein eines Unterzuges durchgeführt. Dabei wurde das nicht Vorhandensein des Unterzuges durch Nullsetzen einer seiner beiden Abmessungen realisiert.

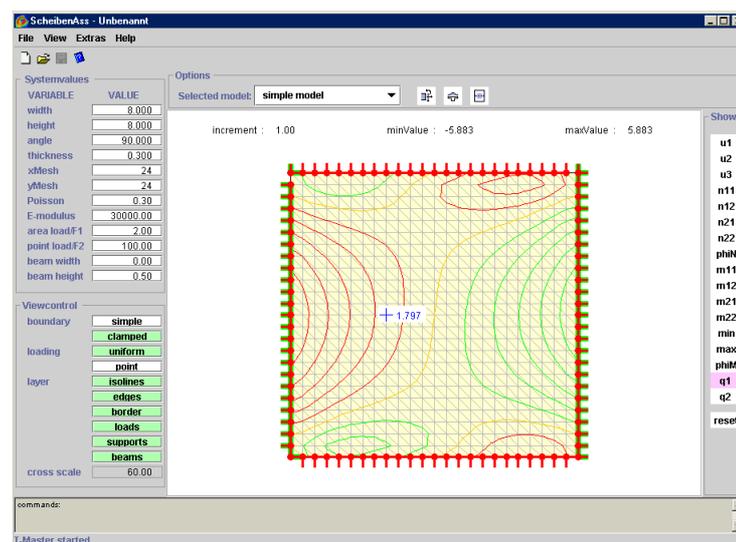
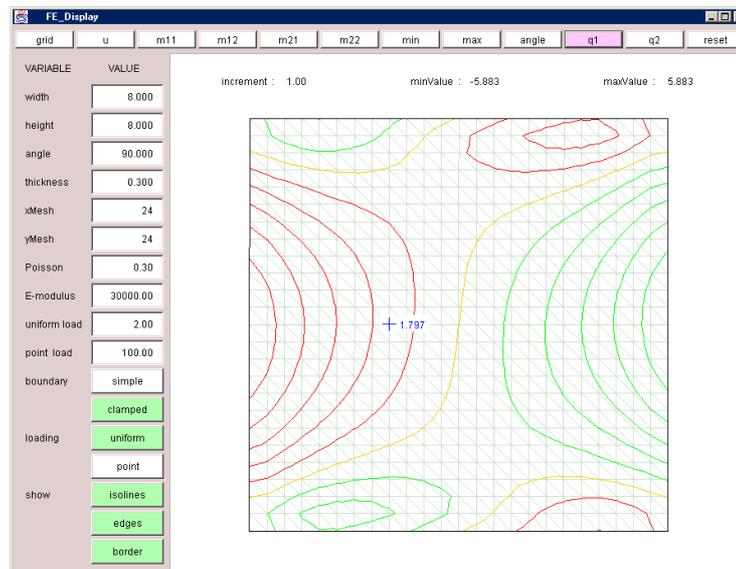


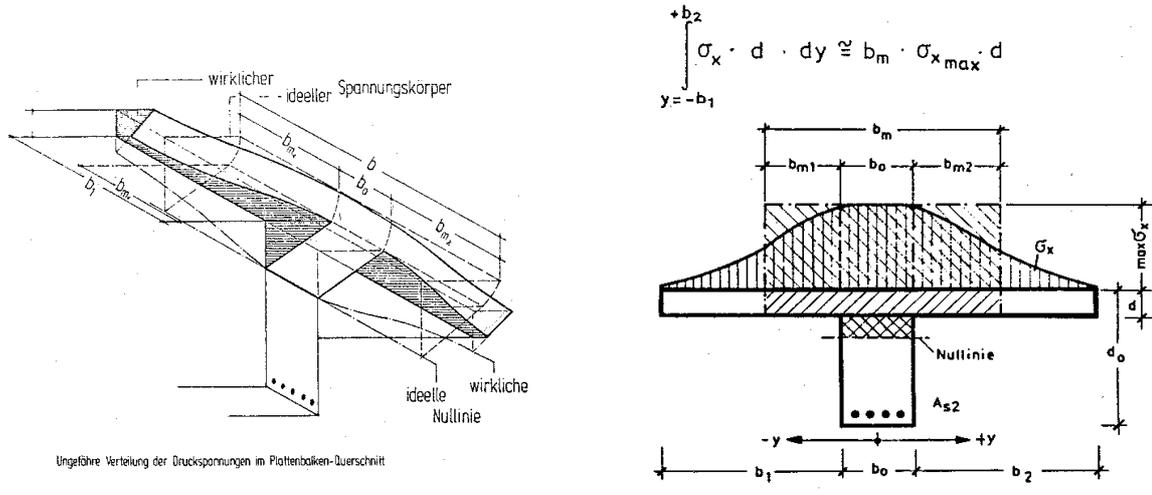
Bild 24: Bsp. für eine Vergleichsrechnung, oben: vorgegebenes Programm, unten: erweitertes Programm

Es konnten dabei keine Unterschiede erkannt werden. Des weiteren wurden ausgewählte Beispiele mit den Czerny-Tafeln überprüft. Hier konnte eine Konvergenz gegen die Tafelwerte erkannt werden.

Kapitel E Bestimmung der mitwirkenden Plattenbreite

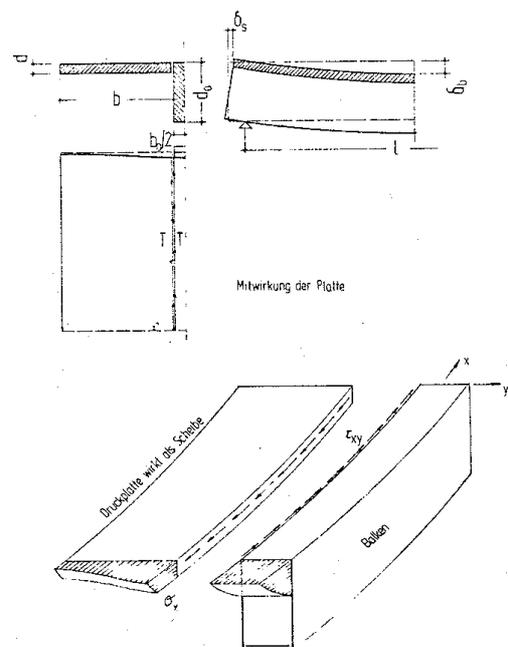
1. Konzept

Das Bemessungskonzept der mitwirkenden Plattenbreite des konstruktiven Stahlbetonbaus beruht auf der Ermittlung einer rechnerischen Ersatzbreite der Druckplatte. Sie ergibt sich, wenn man die tatsächliche, über die Breite veränderliche Spannungsverteilung in der Druckplatte durch einen idealisierten, gleichförmigen Spannungsblock mit dem selben Maximalwert der Spannung ersetzt:



Im vorliegenden Berechnungsmodell werden die aus dem Balken in die Platte eingeleiteten Druckspannungen aus der **Scheibenverformung** der Stahlbetonplatte errechnet.

Die Scheibenverformungen der Platte ergeben sich aus der kinematischen Kopplung der horizontalen Verschiebungsfreiwerte von Platte und Balken und bilden somit die Schubübertragung im Balkenanschnitt des monolithischen Tragwerks ab. Zu beachten ist jedoch, daß durch die Erfassung des Balkenquerschnittes mit Hilfe nur eines Balkenelementes im Modell eine punktuelle Schubeinleitung erfolgt, und nicht wie im wirklichen Tragwerk eine auf rechten und linken Balkenanschnitt verteilte. Dieser Unterschied von Modell und wirklichem Tragwerk hat unmittelbaren Einfluß auf das Verformungsbild der Platte und somit auf die Verteilung der daraus berechneten Plattendruckspannungen.



2. Algorithmus

Wie aus dem vorhergegangenen Abschnitt zu ersehen ist, ist für die Ermittlung der mitwirkenden Plattenbreite zunächst eine genaue Kenntnis der Spannungsverteilung in der Platte normal zu einem senkrecht zum Balken verlaufenden Schnitt notwendig.

Diese Spannungsverteilung kann unmittelbar aus den vorgehaltenen Berechnungsergebnissen gewonnen werden.

2.1. Bestimmung der Spannungsverteilung

Zu bestimmen ist die Plattendruckspannungsverteilung längs eines Schnittes. Da die Bestimmung der Spannungsverteilung auf Grundlage der Ergebnisse der FEM-Analyse erfolgt, unterliegt sie den gleichen Näherungen. Eine übertriebene Genauigkeit bei ihrer Bestimmung ist somit nicht sinnvoll.

Es wurden folgende Vereinbarungen getroffen:

- Die Bestimmung der Spannungsverteilung ist nur in einem Schnitt je Balkenelement und zwar in dessen Mitte möglich
- Der Abstand der Spannungsauswertungspunkte ds längs eines solchen Schnittes beträgt die Hälfte der Balkenelementlänge dl .

Die Mitte des Balkenelementes wurde als Auswertungsort gewählt um eine eindeutige Zuordnung des Schnittes zu einem Balkenelement zu gewährleisten. Dies ist insbesondere von Bedeutung wenn die Unterzugsachse im Grundriß an den Knoten Knicke aufweist. Weiterhin ist zur Mitte schon eine gewisse Ausbreitung der von den Knoten des zum Schnitt gehörenden Balkenelementes in die Platte eingeleiteten Last erfolgt. Auch die Probleme, die die Eigenschaft eines FEM-Modells an den Elementgrenzen un stetige Kraftgrößen sprünge aufzuweisen mit sich bringt, können auf dies Weise vermindert werden.

Eine geeignete Festlegung der Schrittweite ist dahingegen schwierig, da für beliebige FE-Netze keine allgemeingültig vorteilhafte Wahl getroffen werden kann. Sinnvoll scheint es sich in der Feinheit der Spannungsauswertung zumindest am Abstand der Lasteinleitungspunkte des Balkens in die Platte zu orientieren. Andererseits ist eine Auswertung in jedem der vom Schnitt durchfahrenen Elemente wünschenswert, wobei die Lage der Auswertungsstelle aus dem schon erwähnten Phänomen der Unstetigkeit der Kraftgrößen an den Elementrändern möglichst zentral im Elementinnern liegen sollte. Um einen Anhaltspunkt zu bekommen wird von einer gleichmäßigen quadratischen Maschenweite ausgegangen, wobei aus einer Masche durch Teilung entlang einer Diagonalen zwei Dreieckselemente entstehen. Die Erzeugung eines solchen Netzes ist mit dem Generator des vorgegebenen Programms für einfache Geometrien möglich.

Für das beschriebene Netz ergibt sich eine vorteilhafte Wahl der Schrittweite von der Hälfte der Balkenlänge, wie nachfolgendes Bild veranschaulicht:

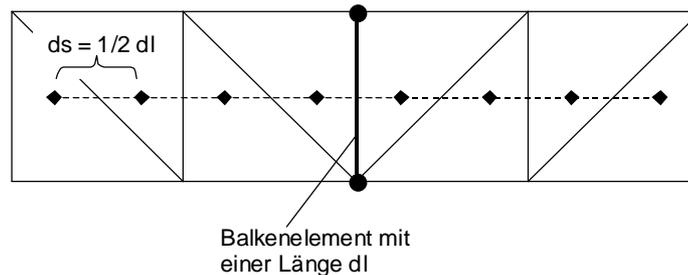


Bild 27: Wahl der Auswertungsstellen bei einem gleichmäßigen symmetrischen Netz

Die getroffene Wahl ist bei symmetrischem und nicht symmetrischem Netz gleichermaßen vorteilhaft. Für beliebige Netze kann keine genaue Aussage getroffen werden. Jedoch ist bei im Grundriß zusammenfallender Balkenelementachse mit Scheibenelementkante eine vorwiegende Auswertung im Elementinnern wahrscheinlich. Das Zusammenfallen im Grundriß von Scheibenelementkante und Balkenelement wird gewählt um im Bereich des Balkenelementes keine un stetigen Kraftsprünge sowie geknickte Verschiebungsverläufe zu haben, da mit dem gekoppelten Elementen ein monolithischer T-Querschnitt erfaßt werden soll.

Unter den gewählten Voraussetzungen wird die Spannungsverteilung wie folgt bestimmt:

Zunächst wird vom Nutzer die Angabe eines Balkenelementes erwartet, durch welches der Schnitt verlaufen soll. Sodann wird aus den Koordinaten der Endpunkte dieses spezifizierten Balkenelementes der Mittelpunkt bestimmt. Ausgehend von diesem Punkt wird entlang einer gedachten Linie senkrecht zur Balkenachse in Plattenebene zunächst nach links mit der Hälfte der aus der Vereinbarung vorgegebenen Schrittweite ($\Rightarrow 0,5 ds = 0,25 \cdot \text{Balkenelementlänge}$) vorangeschritten. Für den so gewonnenen neuen Punkt wird überprüft, ob er innerhalb eines im Modell enthaltenen ebenen Faltwerkelementes enthalten ist. Ist dies nicht der Fall, so ist die Ermittlung der Spannungsverteilung links vom Balken abgeschlossen, da über den Plattenrand hinweggegriffen wurde.

Wurde jedoch ein Element gefunden, daß den neuen Punkt enthält, so wird die senkrecht zum Schnitt wirkende Spannung ermittelt.

Dies geschieht wie in Kapitel A erläutert durch Drehen der vorgehaltenen Ergebnisse der Verschiebungsfreiwerte an den Knoten in ein zur Balkenachse parallel verlaufendes Koordinatensystem und der anschließenden Auswertung der auf dieses System bezogenen n_{11} -Spannung des betreffenden Punktes. Die entsprechende Methode `computeN` wird vom Element bereitgestellt (siehe Kapitel A Abschnitt 3).

Aufgrund der Reduktion der Stahlbetonplatte auf seine Mittelebene sind die so erhaltenen Spannungen schon integrale Werte über die Plattendicke.

Auf gleiche Art und Weise wird nun mit der definierten Schrittweite ($ds=0,5 \cdot \text{Balkenelementlänge}$) fortgeschritten und an jedem Punkt die Spannung ausgewertet, bis das Ende der Platte erreicht wird. Danach wird die Berechnung in analoger Weise auf der rechten Seite der Balkenachse fortgesetzt.

Die Zwischenwerte werden zu einer stetigen Kurve linear interpoliert. Das nachfolgende Bild veranschaulicht die Vorgehensweise.

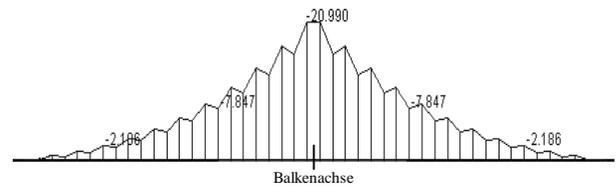
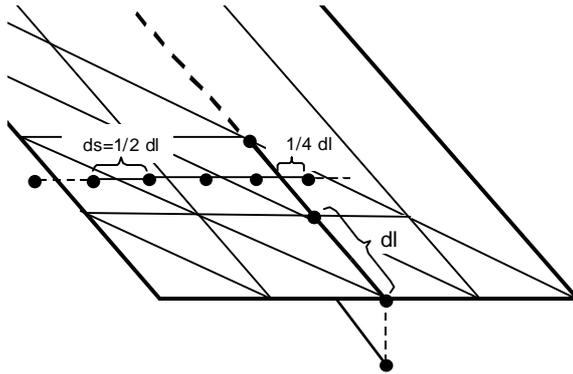


Bild 28: Spannungsverlauf

Bild 29: Auswertungsstellen bei der Spannungsermittlung

bestimme Mittelpunkt	
Mittelpunkt in einem Faltwerkselement enthalten	
+	-
Bestimme nächsten linken Punkt im Abstand $0,5 ds$	Ende, keine Platte vorhanden
solange Punkt in einem Faltwerkselement enthalten	
Bestimme N und füge vorne in den Vektor ein	
Länge += ds	
Bestimme nächsten linken Punkt im Abstand ds	
Länge == 0	
+	-
Bestimme N am Mittelpunkt	Länge -= $0,5 ds$
Mitte = Länge	
Bestimme ersten rechten Punkt im Abstand $0,5 ds$	
solange Punkt in einem Faltwerkselement enthalten	
Bestimme N und füge hinten in den Vektor ein	
Länge += ds	
Bestimme nächsten rechten Punkt im Abstand ds	
Mitte == Länge	
+	-
Bestimme N am Mittelpunkt	Länge -= $0,5 ds$

Bild 30: Ablaufstruktogramm zur Spannungsermittlung

Die Spannungswerte an den ausgewerteten Punkten werden in einem Vektor verwaltet.

Die einzelnen Streckenlängen werden nicht gespeichert, lediglich die Gesamtlänge und die Länge des linken Flügels des Auswertungsbereichs werden gespeichert, da eine konstante Schrittweite ds vorausgesetzt wurde.

Es sei darauf hingewiesen, daß der Auswertungsbereich nicht exakt der Gesamtplattenbreite an der betrachteten Stelle entspricht. Der Auswertungsbereich kann die wirkliche Breite bis annähernd ein ds unterschreiten.

Um eine, an die Fenstergröße angepasste, skalierte Darstellung zu ermöglichen, wird parallel zur Ermittlung des Spannungsverlaufes der betragsgrößte Spannungswert im Schnitt ermittelt. Zur aussagekräftigen Beschriftung und zur späteren Ermittlung der mitwirkenden Plattenbreite wird auch der absolut kleinste Wert im Integrationsbereich benötigt. Dieser wird jedoch erst während der Integration ermittelt, um sicherzustellen, daß er innerhalb des Integrationsbereichs liegt.

2.2. Bestimmung der mitwirkenden Plattenbreite

Wie in Abschnitt 1. dieses Kapitels bereits beschrieben ergibt sich die mitwirkende Plattenbreite aus der Umwandlung der Spannungsfläche in ein flächengleiches Rechteck, dessen Höhe dem auftretenden Maximum entspricht.

2.2.1. Bestimmung der Spannungsfläche

Die Spannungsfläche ist aufgrund ihrer schrittweisen Ermittlung schon in Trapezflächen unterteilt. Die Bestimmung der Gesamtfläche erfolgt somit als Summe der einzelnen Teilflächen. Die Höhen der Einzeltrapeze sind bereits aus der Spannungsermittlung bekannt und wurden in der Reihenfolge von links nach recht abgespeichert. Die Breite aller Trapeze beträgt d_s .

Zu beachten ist jedoch, das im Integrationsbereich Nulldurchgänge liegen können oder es sich um einen Randunterzug handelt, so daß die Breite des ersten Trapezes neben dem Balken nur $0,5 d_s$ beträgt. Die Integration geschieht dabei, wie schon bei der Spannungsermittlung, von der Mitte ausgehend nacheinander für beide Seiten. Zur Bestimmung der mitwirkenden Plattenbreite kommen dabei nur die negativen (Druck-) Spannungen bis zum jeweils ersten Nulldurchgang in Betracht.

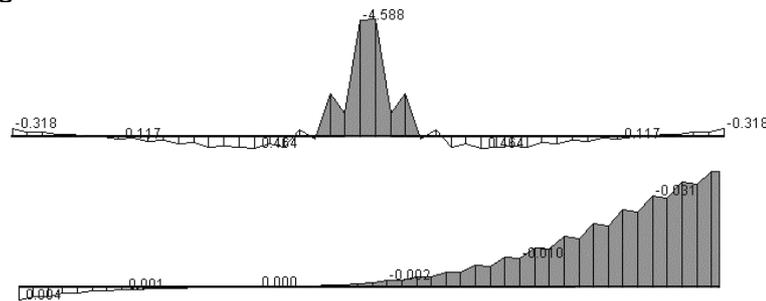
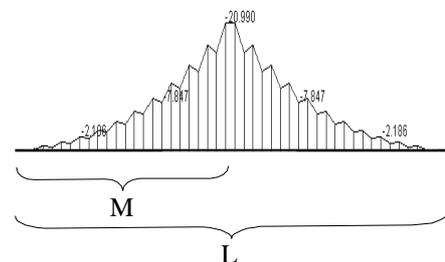
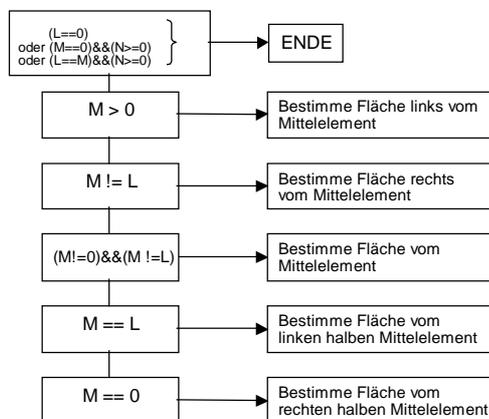


Bild 31: mögliche Integrationsgrenzen

Den Ablauf der Integration veranschaulicht nachfolgendes Diagramm:



N: Spannungswert über der Balkenachse

$$\text{Fläche} = \sum \text{Teilflächen}$$

Bild 32: Ablaufdiagramm

Innerhalb der Integrationsgrenzen erfolgt die Teilflächenermittlung dabei nach folgendem Schema:

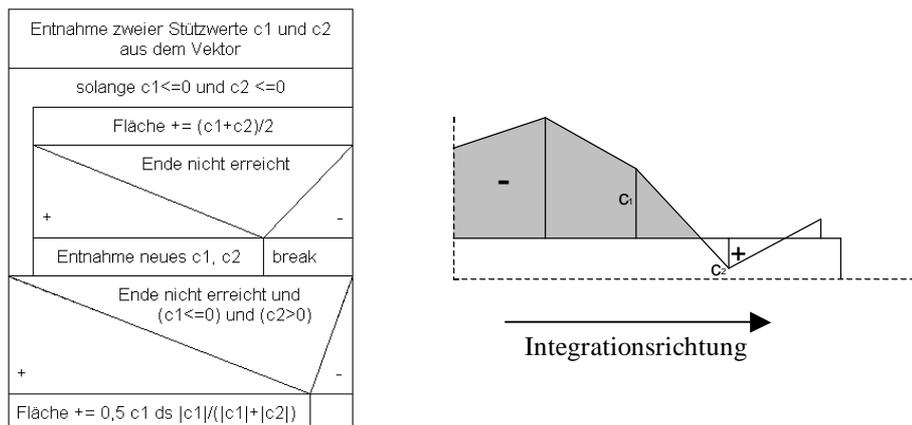


Bild 33: Prinzip der Teilflächenermittlung

Für die Integration in die andere Richtung sind die Bedingungen entsprechend anzupassen. Wie schon erwähnt erfolgt während der Integration gleichzeitig die Bestimmung des absolut kleinsten Spannungswertes (Druck). Dazu werden c1 und c2 jedes Mal mit dem aktuell kleinsten Wert verglichen und gegebenenfalls gespeichert.

2.2.2. Bestimmung der mitwirkenden Breite

Die mitwirkende Plattenbreite wird aus der Umwandlung der Spannungsfläche in einen Spannungsblock mit dem Maximalwert der Plattendruckspannung ermittelt:

$$b_m = \frac{\text{Fläche}}{\text{absolutes Spannungswert minimum im Integrationsbereich}}$$

Sie wird im Schnitt zusammen mit der Spannungsverteilung visualisiert, wobei darauf zu achten ist, daß sie je nach geometrischen Gegebenheiten eine exzentrische Lage bezüglich der Balkenachse besitzen kann.

Wird eine mitwirkende Plattenbreite erreicht, die kleiner ist als die Unterzugsabmessung, so wird ein Hinweis ausgegeben.

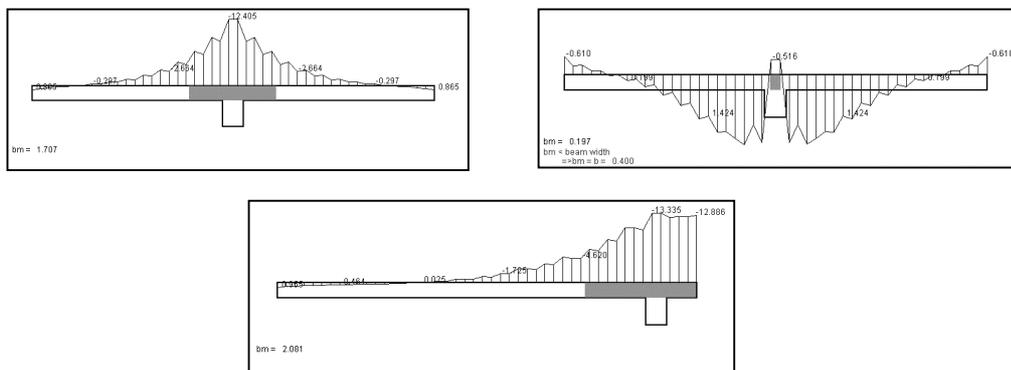


Bild 34: mögliche mitwirkende Plattenbreiten

2.3. Grundrißansicht

Um einen besseren Überblick über den Verlauf der mitwirkenden Plattenbreite längs des Unterzuges zu erhalten wird aus allen möglichen Schnitten eine Grundrißansicht generiert und in einem weiteren Fenster visualisiert. Zur besseren Orientierung können sowohl die Balkenelemente, als auch die Plattenelemente hinterlegt werden.

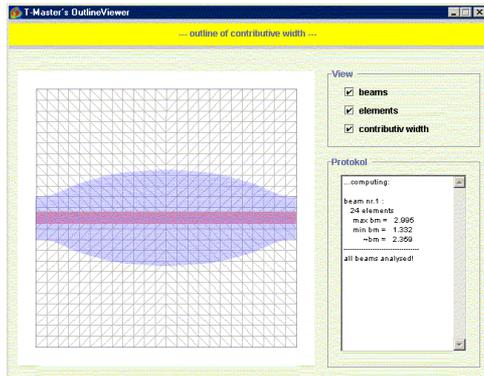


Bild 35: mitwirkender Bereich im Feld

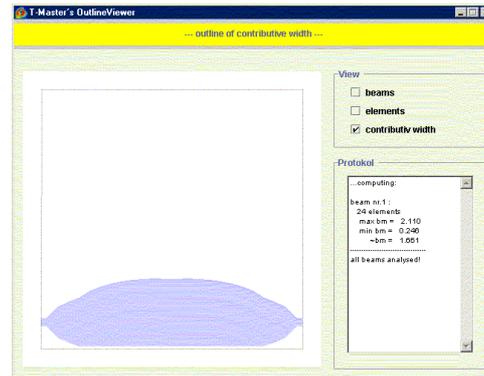


Bild 36: Randberührung des mitwirkenden Bereichs

Die Fläche der mitwirkenden Plattenbreite entsteht, indem die in der Mitte jedes Balkenelementes ausgewerteten Breiten aufgetragen und linear interpoliert werden. Am Anfangs- und Endelement wird über die halbe Elementlänge eine konstante mitwirkende Breite angetragen.

Auf diese Weise werden die in jedem Schnitt berechneten mitwirkenden Plattenbreiten angezeigt, unabhängig davon ob sie die vorhandene Unterzugsbreite unterschreiten oder nicht. Um ein markanteres Bild zu erhalten erfolgt keine Anhebung auf die Unterzugsbreite.

In Querrichtung zur Balkenlängsachse erfolgt die Darstellung der Fläche immer bis zum letzten Auswertungspunkt und nicht bis zum Plattenrand, woraus sich bei Randberührung des mitwirkenden Bereichs ein schmaler Reststreifen ergibt. (siehe Bild 35)

3. Implementierung

Die Darstellung des Schnittes und der Grundrißansicht des mitwirkenden Bereichs soll wie auch die der Balkenschnittgrößen in einem jeweils eigenständigen Fenster erfolgen. Für diese Fenster wurde eine abstrakte Klasse `SectionViewer` geschaffen. Sie stellt ein Fenster mit Überschriftsleiste und einen Zeichenbereich zur Verfügung. Die Algorithmen der Spannungsermittlung und der Berechnung der mitwirkenden Breite, sowie die einzelnen Zeichenmethoden sind in den von dieser Klasse abgeleiteten Klassen `ForceViewer` (Balkenschnittkräfte), `WidthViewer` (Schnitt) und `OutlineViewer` (Grundrißansicht) implementiert. Alle Klassen gehören zum Paket `graphik`.

Bei Aufruf über den entsprechenden Knopf in der Oberfläche wird das jeweilige Objekt erzeugt, daß selbstständig alle Berechnungen durchführt. Der `OutlineViewer` greift dabei auf die Berechnungsmethode des `WidthViewer` zurück.

Kapitel FFEM-Analyse einfacher Deckensystemen

Nachfolgend sollen die Ergebnisse des erstellten FEM-Programms anhand von Untersuchungen an einfachen Deckensystemen beurteilt werden.

1. Allgemeines

Betrachtet man die Ergebnisse bei Vorhandensein eines Unterzuges und vergleicht diese mit denen einer einfachen Deckenplatte, so stellen sich die zu erwartenden Unterschiede ein. Die Durchbiegung wird über dem Balken verringert, es bilden sich Stützmomente aus und die Druckeinleitung des Balkens in die Platte erzeugt ein ovales Spannungsfeld. Nachfolgend ist dies für eine allseitig einfach aufgelagerte Platte mit minimaler Scheibenlagerung gezeigt.

Systemvalues	
VARIABLE	VALUE
width	8.000
height	8.000
angle	90.000
thickness	0.300
xMesh	24
yMesh	24
Poisson	0.30
E-modulus	30000.00
area load/F1	-2.00
point load/F2	100.00
beam width	0.60
beam height	0.80

Bild 37: Systemwerte, Berechnung für Flächenlast

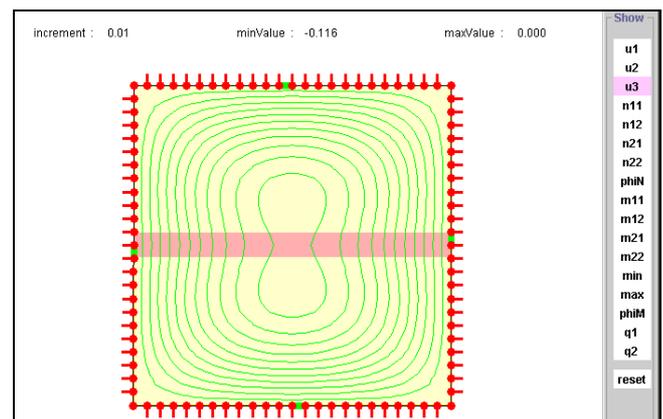


Bild 38: Durchbiegung

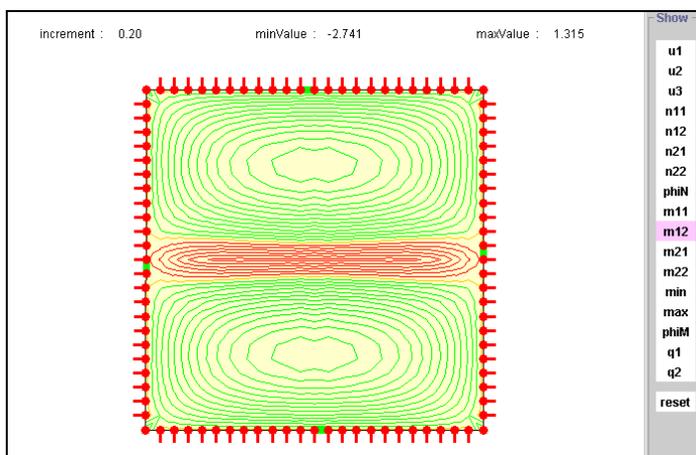


Bild 39: Stützmoment über dem Balken

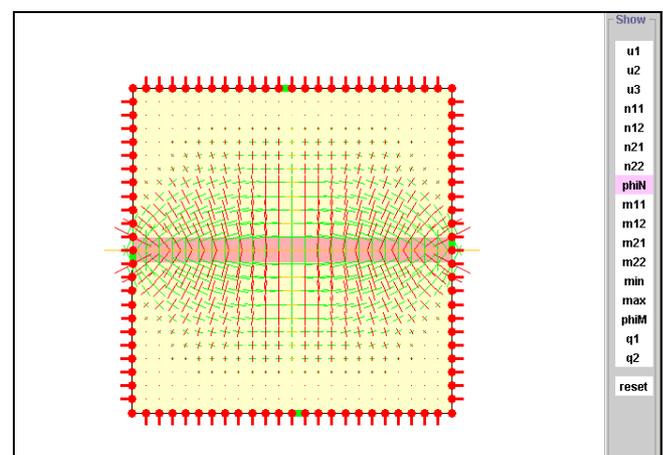


Bild 40: Hauptspannungsfeld

Für alle anderen Größen ergeben sich ähnlich plausible Ergebnisse!

Die mitwirkende Plattenbreite erstreckt sich bei einfacher Auflagerung über die ganze Länge der Decke und verringert sich zu den Auflagern hin.

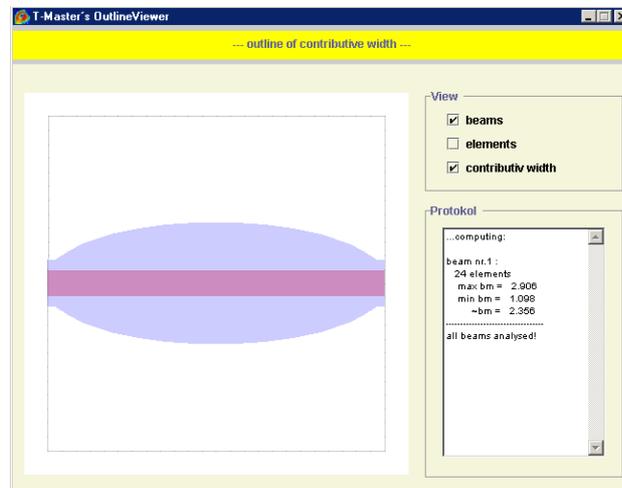


Bild 41: Grundrißansicht des mitwirkenden Bereichs

Bei der Berechnung solcher Deckensysteme ist besonders auf eine minimale Scheibenlagerung zu achten, da sonst unbeabsichtigte Zwängungen entstehen, die zusammen mit den getroffenen Näherungen (Berechnung bezüglich der Plattenmittelebene) zu unerwünschten Effekten führen.

Bei voller Scheibenlagerung an den beiden Stirnseiten des Systems ergeben sich nachfolgend gezeigte Unterschiede in den Horizontalverschiebungen:

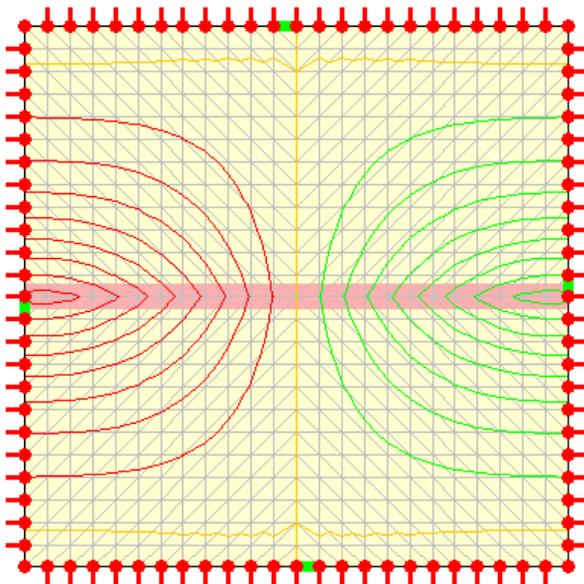


Bild 42: u_1 bei minimaler Scheibenlagerung

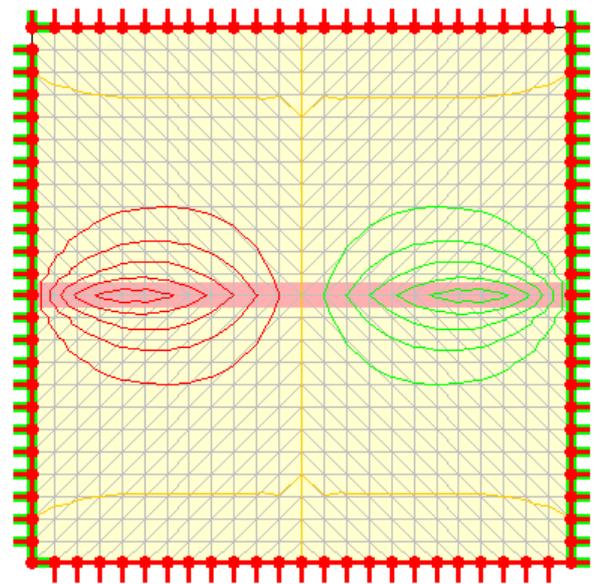


Bild 43: u_1 bei voller Scheibenlagerung an den Stirnseiten

rot: pos. Verschiebung

grün: neg. Verschiebung

orange: Nulllinie

Dies hat zur Folge, daß die Scheibenspannungen in den Punkten mit maximaler Verschiebung einen Nulldurchgang haben und somit der mitwirkende Bereich der Platte abreißt.

Sinnvoll wäre eine volle Scheibenlagerung ausschließlich in den beiden Balkenendknoten in der wirklichen Systemachse. Dies ist jedoch nicht möglich.

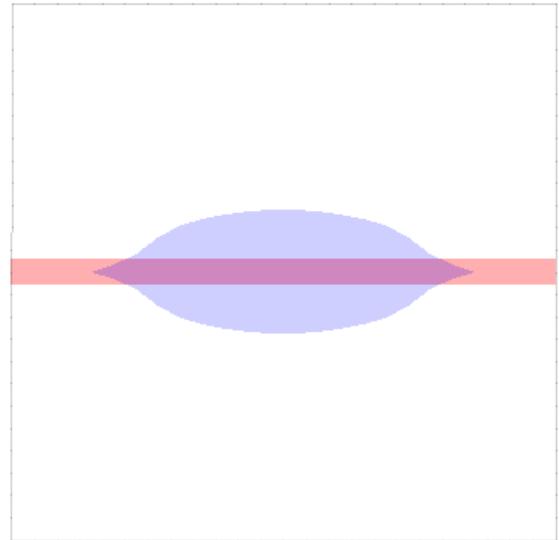
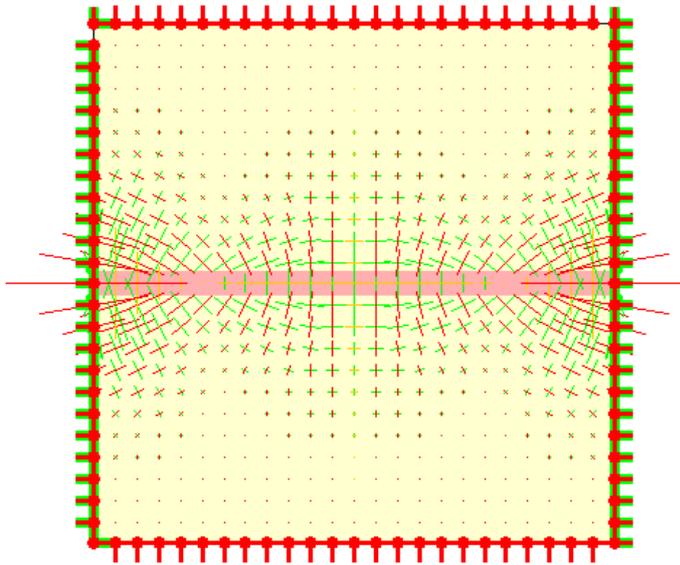


Bild 44: Hauptspannungsfeld bei voller Scheibenlagerung an den Stirnseiten

Bild 45: Abreißen des mitwirkenden Bereichs

Betrachtet man die Spannungsverteilung der Plattendruckspannungen in einem Schnitt senkrecht zur Balkenachse, so fällt auf, daß sich nicht die erwartete Glockenform sondern eine dreiecksförmige Spannungsverteilung ergibt. Dies ist darauf zurückzuführen, daß der Balken über seine Breite nur in einem Knoten mit dem FE-Netz der Platte verbunden ist, und sich somit eine Spannungsspitze im Einleitungspunkt ergibt. Bei der Verwendung mehrerer Balken hingegen ergibt sich eine Ausrundung, auch wenn der Einfluß nur sehr begrenzt erscheint.

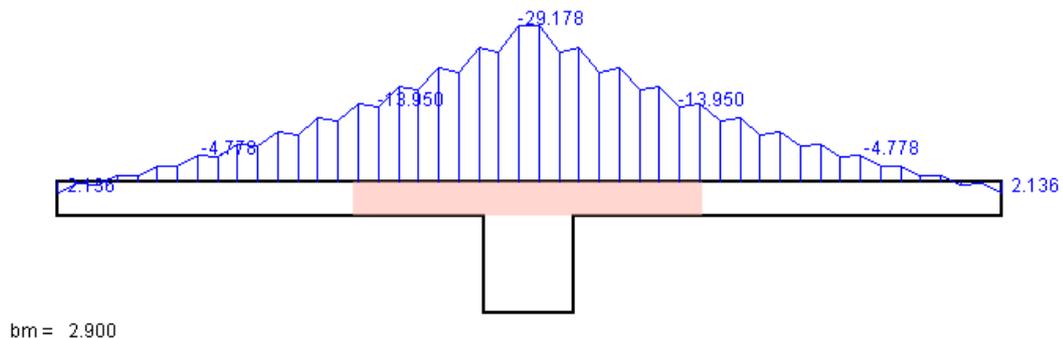


Bild 46: Spannungsverteilung einem Balken 75cm x 80cm

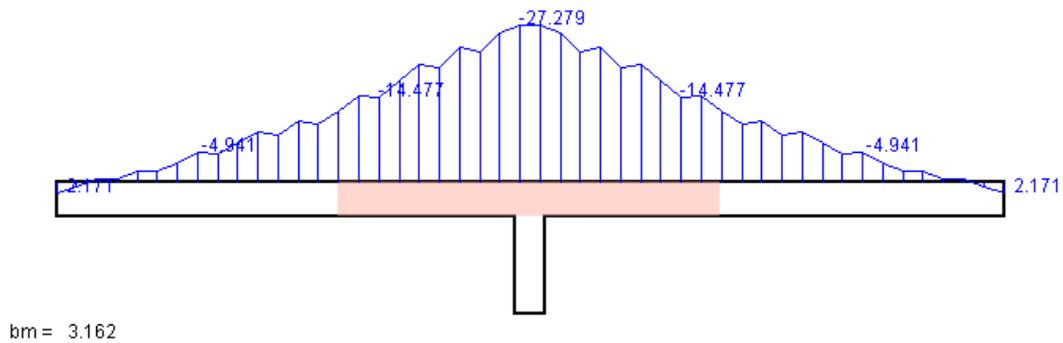


Bild 47: Spannungsverteilung bei drei Balken 25cm x 80cm

Bei den Balkenschnittgrößen, die auf die Balkenachse bezogen sind, ergibt sich ein folgender Effekt. Die Verläufe von N, Q und M weisen unstetige Sprünge an den Balkenelementgrenzen auf, deren Größe erst bei starker Netzverfeinerung vernachlässigbar klein wird.

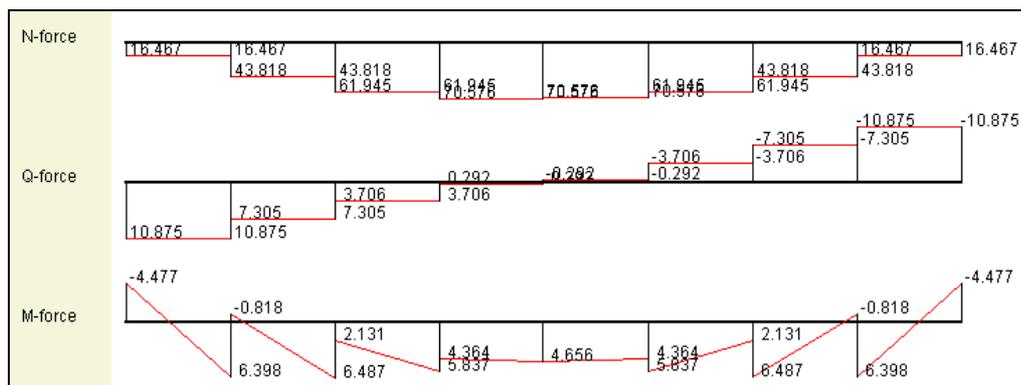


Bild 48: beispielhafte Verläufe der Balkenschnittkräfte bei minimaler Scheibenlagerung, Flächenlast und einem FE-Netz von 8x8

Dies ist auf die punktuelle Kopplung von Platten-/Scheibenverformung auf der einen und Balkenverformung auf der anderen Seite zurückzuführen.

Der Einfluß der Platten-/und Scheibenwirkung verhindert, daß sich im Balken die bei reiner Balkenbiegung vorhandenen Querschnittsverdrehungen in vollem Maße einstellen, wodurch jeweils an den Kopplungsstellen ein Moment eingeleitet wird. Nimmt der Einfluß der Plattensteifigkeit im Verhältnis zur Balkensteifigkeit ab, so ist dieser Effekt geringer.

Beispielhaft wird der Einfluß der Scheibenwirkung auf die Kraftgrößenverläufe des exzentrischen Balkens an zwei Extrembeispielen unter Einzellast gezeigt. Es wird ein exzentrischer Balken berechnet. Zunächst ist dieser an den beiden Stirnseiten fest gelagert, alle anderen Knoten erhalten eine minimale Lagerung. Im zweiten Berechnungsschritt wird eine starre Scheibe durch eine feste Horizontallagerung aller Punkte simuliert:

Das System wird zunächst aus vier Balkenelementen zusammengesetzt, die eine Höhe von 50cm sowie eine Breite von 40cm besitzen. Die Gesamtlänge des Systems beträgt 2m. Die Balkenachse besitzt bezüglich der Berechnungsebene eine Exzentrizität von 40cm. Aufgebracht wird eine Last von 100.

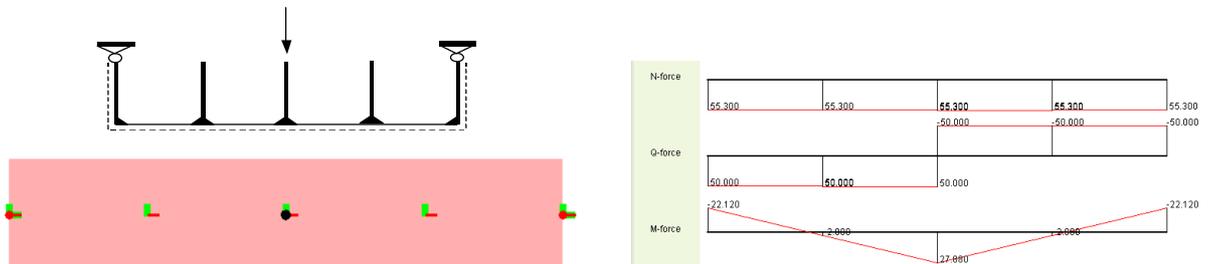


Bild 49: horizontal feste Lagerung der Stirnseiten

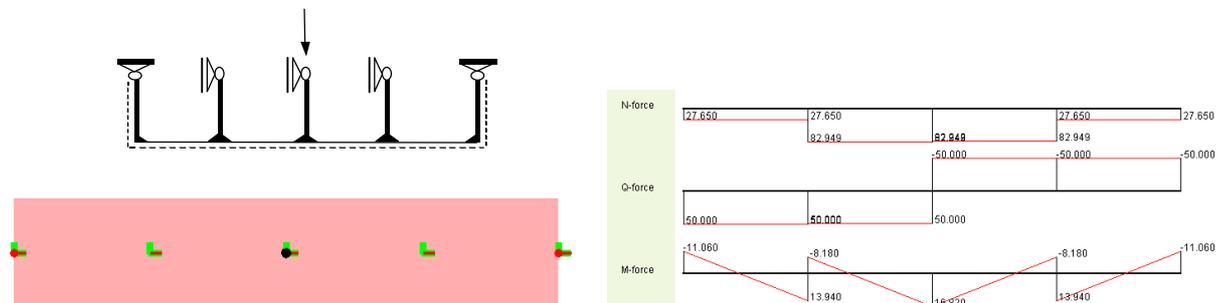


Bild 50: vollständig horizontal feste Lagerung

Bei dem Beispiel aus Bild 48 stellt sich der erwartete Momentenverlauf ein. Bei Bild 49 treten bedingt durch die Zwängungen aus den horizontal fest gelagerten Knoten Momentensprünge an den Elementgrenzen des Balkens auf. Mit einer erhöhten Elementanzahl wird eine Annäherung an den in Wirklichkeit vorhandenen kontinuierlichen Verbund erreicht und der Betrag der Momentensprünge verringert sich. (siehe nachfolgende Bilder)

Bei sehr feiner Unterteilung ergeben sich bei der Lagerung aus Bild 49 für M und Q die Qualitativen Verläufe des beidseitig fest gelagerten, einfach unbestimmten Balkens auf zwei Stützen.

Die Normalkraft ist aufgrund der kontinuierlichen horizontalen Stützung und der Kopplung des Dehnanteils mit dem Biegeanteil über die Exzentrizität linear verteilt. Dies ist aus folgender Überlegung heraus sinnvoll: Der Momenten- und somit auch der Krümmungsverlauf sind linear, woraus sich ein quadratischer Verlauf der Querschnittsverdrehung φ ergibt. Durch die exzentrische Lage bezüglich der Lagerung folgt hieraus mit $u = \varphi \cdot e$ ein quadratischer Verschiebungsverlauf. Die hieraus berechneten Dehnungen sind durch ihre Definition als Ableitung der Verschiebungen linear verteilt woraus sich unmittelbar eine lineare Normalkraftverteilung ergibt. An den Enden ergibt sich die Normalkraft annähernd zu Null, da sie entlang des ganzen Balkens in die Lager eingetragen wird.

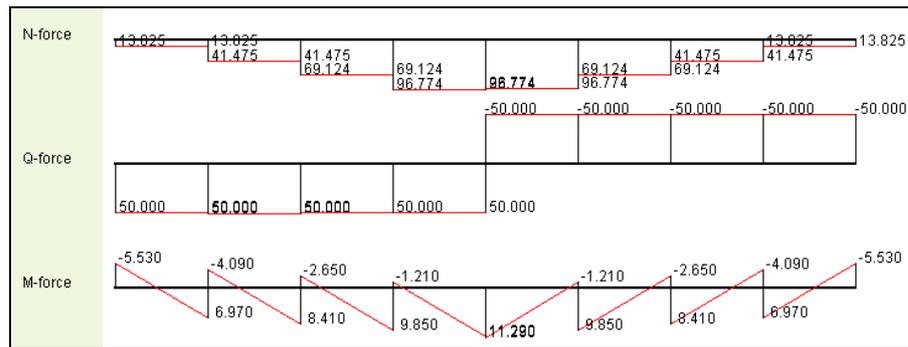


Bild 51: Balkenschnittkräfte bei 8 Elemente

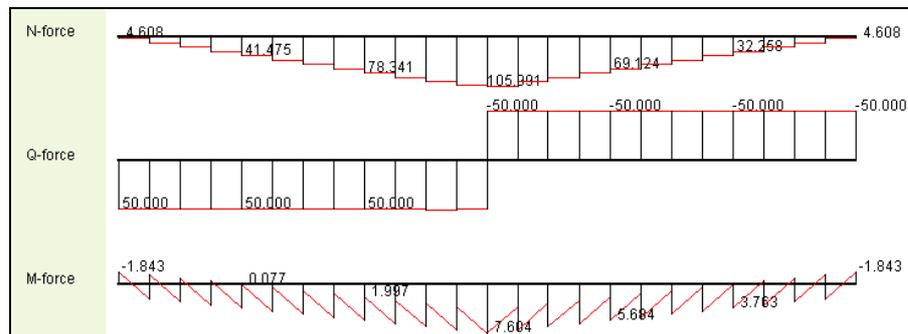


Bild 52: Balkenschnittkräfte bei 24 Elemente

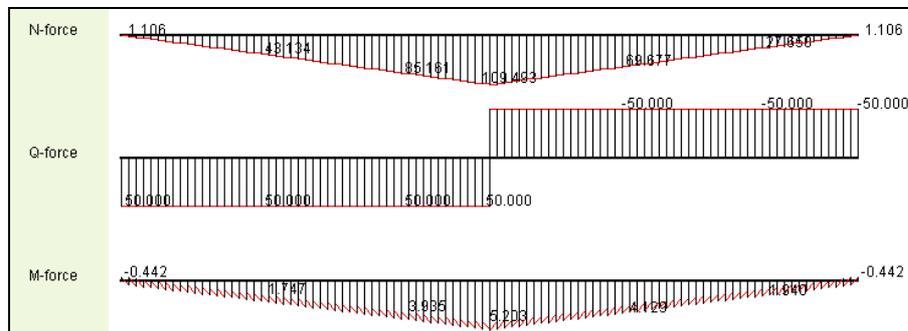


Bild 53: Balkenschnittkräfte bei 100 Elemente

Dieser Effekt ist vorerst als negativ zu beurteilen, da er mit einem Steifigkeitsabfall verbunden ist, welches wiederum direkten Einfluß auf die erzielten Ergebnisse für die mitwirkende Plattenbreite hat.

Das Anwachsen der Balkennormalkraft mit zunehmender Elementanzahl bei vorhandener horizontaler Lagerung läßt auf eine Steigerung der Plattendruckkräfte schließen, die bei qualitativ gleichem Verlauf zu einer größeren mitwirkenden Plattenbreite führen würden.

Eine Einordnung der mit diesen Näherungen erzielten Ergebnisse bezüglich der in Heft 220 DAfStb angegebenen mitwirkenden Plattenbreiten soll nun anhand von Betrachtungen an einfachen Deckensystemen erfolgen. Ausgewählt wurden die vier einfachsten Systeme Kragarm, einfach gelagerter Balken und zwei- bzw. dreifeldriger Durchlaufträger, jeweils mit T-Querschnitt. Weiterhin wurde der Einfluß der Art der Belastung sowie der Maschengröße des FE-Netzes verfolgt. Dem Programm entsprechend wurden Einzel- und Steckenlast gewählt.

2. Einfach gelagerter Balken

2.1. Flächenlast

Es wird eine an den Stirnseiten einfach gelagerte Deckenplatte mit mittig angeordnetem Unterzug und minimaler Scheibenlagerung betrachtet. Die Breite der Stirnseiten wird auf 2m festgelegt. Die Systemlänge wird im Bereich von 1m bis 8m in Meterschritten variiert. Zunächst erfolgt die Belastung mit einer Flächenlast. Die Systemwerte für alle nachfolgenden Beispiele sind Bild 53 zu entnehmen.

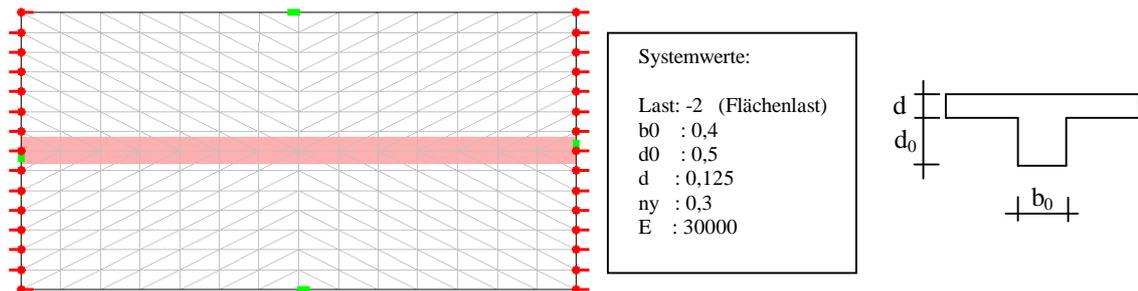


Bild 54: System des einfach gelagerten Balkens

Es wird ein doppelt symmetrisches Netz verwendet, dessen Maschenweite später variiert wird. Für die erste Untersuchung wird eine konstante Maschenanzahl von 24x24 gewählt. Dies bedeutet, daß sich mit der Variation der Systemlänge ein unterschiedlich stark verzerrtes Netz entsteht. Ebenso ändert sich durch die Kopplung der Schrittweite der Auswertung der Spannungen im Schnitt die Genauigkeit der Berechnung der mitwirkenden Plattenbreite.

Je untersuchtes System wurden zwei Vergleichswerte ermittelt. Der eine wurde mit der in Heft 220 DAfStb angegebenen, auf der sicheren Seite befindlichen, Regelung von $b_m = 1/3 L$ berechnet. Der andere wurde in Abhängigkeit des Systems und der Systemwerte nach den in Heft 220 DAfStb angegebenen Tabellen ermittelt.

Die Berechnungen mit dem entwickelten FEM-Programmpaket wurden sowohl mit einem Balken der Breite 0,4m als auch mit 5 Balken der Breite 0,08m durchgeführt. Um den Einfluß der Netzverzerrung auf die Ergebnisse der mitwirkenden Plattenbreite aufzuzeigen, wurde anschließend für die Berechnung mit einem Balken eine Vergleichsrechnung durchgeführt, bei der die Anzahl der Elemente so angepaßt wurde, daß sich immer ein gleichmäßiges Netz ergibt.

Geometrie	1 Balken		5 Balken		1 Balken	
	1/3 L	mit b = 0,4	mit b = 0,08	nach Heft 220	angepaßtes Netz	Netz
2m x 1m	0,333	0,379	0,602	0,864	0,380	24x12
2m x 2m	0,666	0,709	0,922	1,280	0,709	24x24
2m x 3m	1,000	1,044	1,194	1,530	1,045	24x36
2m x 4m	1,333	1,398	1,448	1,760	1,328	24x48
2m x 5m	1,666	1,627	1,655	1,862	1,520	24x60
2m x 6m	2,000	1,650	1,669	1,936	1,642	24x72
2m x 7m	2,000	1,723	1,737	1,968	1,720	24x84
2m x 8m	2,000	1,702	1,862	2,000	1,773	24x96

Tabelle 2: mitwirkende Plattenbreiten beim einfach gelagertem Balken

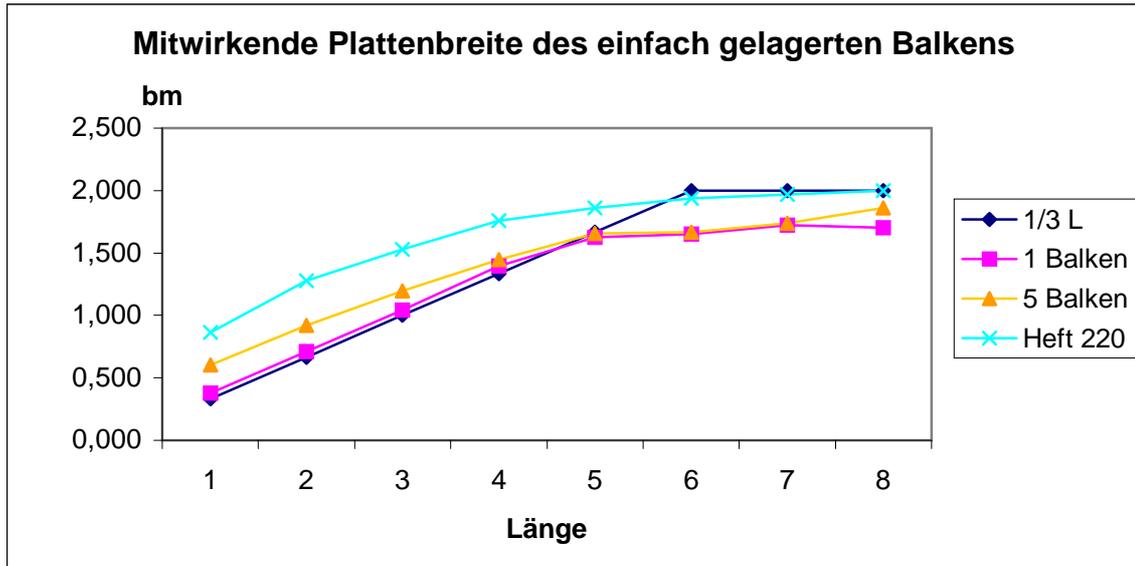


Diagramm 1: Berechnungsergebnisse für den einfach gelagerten Balken

Zu erkennen ist, daß bis zur Systemlänge 5m die beiden nach Heft 220 DAfStb berechneten Vergleichswerte als obere und untere Schranke angesehen werden können, wobei die erhaltenen Werte dichter an der nach der 1/3-Regel ermittelten Kurve liegen. Ebenfalls ist zu erkennen, daß sich bei der Berechnung mit 5 Balken im Bereich $L < 5m$ eine größere mitwirkende Breite ergibt. Dies ist darauf zurückzuführen, daß sich bei der Verwendung von nur einem Balken in der Mitte eine Spannungskonzentration ergibt. Das Erreichen der Systembreite als mitwirkende Plattenbreite geschieht nicht, da sich nie eine konstante Spannungsverteilung im Schnitt ergibt. Die errechneten mitwirkenden Breiten liegen deshalb ab $L > 5m$ unterhalb der Vergleichswerte.

Der Einfluß der Netzverzerrung ist wie Diagramm 2 zeigt nur sehr gering. Zum Vergleich ist nochmals die Kurve für die Berechnung mit einem Balken bei verzerrtem Netz aufgetragen.

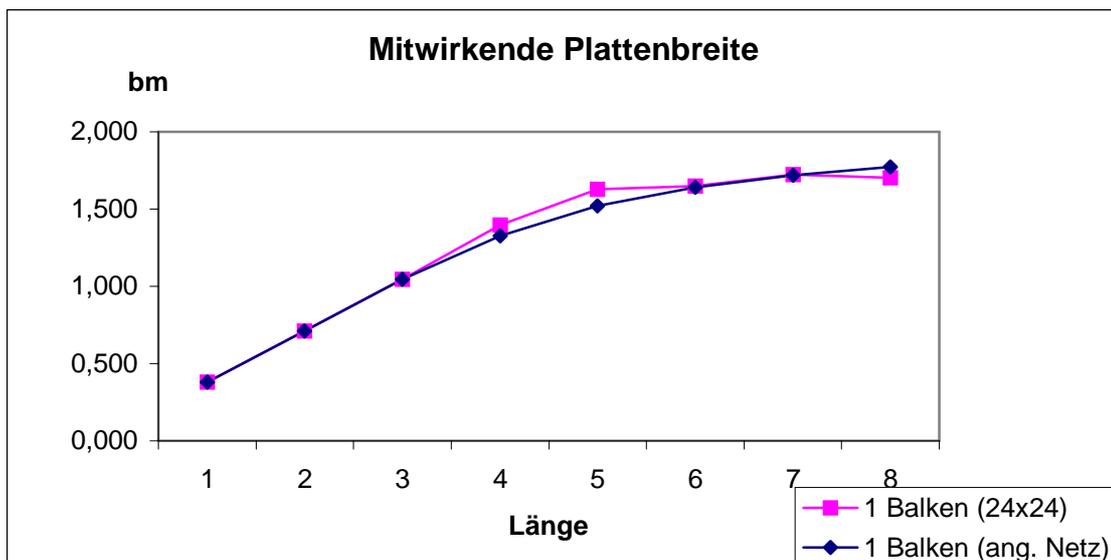


Diagramm 2: Vergleich verzerrtes/unverzerrtes Netz

Die größte Abweichung ergibt sich für die Geometrie 2mx5m. Für diese Geometrie wurden weitere Untersuchungen angestellt.

Als erstes wurde bei festgehaltener Geometrie eine in beide Richtungen gleichmäßige Netzverfeinerung vorgenommen. Die Ergebnisse sind in Tabelle 3 und Diagramm 3 dargestellt. Zu erkennen ist ein streng monotoner Anstieg der mitwirkenden Plattenbreite, die gegen einen Wert von ca. 1.53 konvergiert.

Netz	bm
4x10	1,381
8x20	1,457
12x30	1,488
16x40	1,504
20x50	1,514
24x60	1,520
28x70	1,525
32x80	1,529

Tabelle 3: gleichmäßige Netzverfeinerung

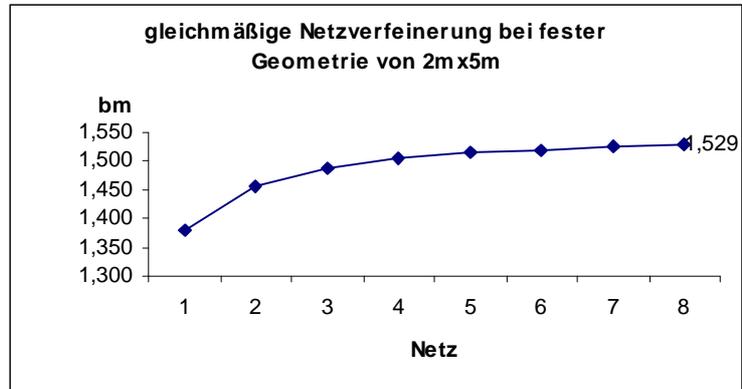


Diagramm 3: Konvergenz der mitwirkenden Plattenbreite bei gleichmäßiger Netzverfeinerung

Im Vergleich zur gleichmäßigen Netzverfeinerung wurde außerdem eine Verfeinerung ausschließlich in Systemlängsrichtung vorgenommen. Die Systembreite wurde konstant mit 24 Elementen diskretisiert.

Interessant ist dies, da somit die Anzahl und Länge der Balkenelemente variiert und im gleichen Zuge die Anzahl der Auswertpunkte im Schnitt erhöht wird, wohingegen die Elementierung der Platte in Querrichtung unverändert bleibt. Zu erkennen ist dabei eine nicht monotone Konvergenz die erheblich langsamer erfolgt.

Netz	bm
24x4	1,793
24x8	1,384
24x12	1,541
24x16	1,437
24x20	1,557
24x24	1,627
24x28	1,543
24x32	1,533
24x36	1,470
24x40	1,509
24x44	1,545
24x48	1,500
24x52	1,531
24x56	1,498
24x60	1,520
24x64	1,540
24x68	1,512
24x72	1,534

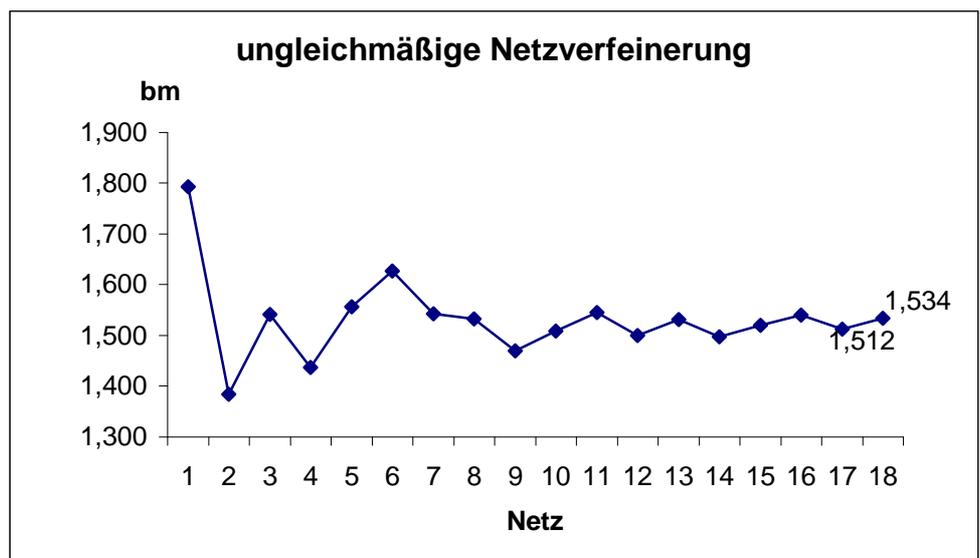


Tabelle 4, Diagramm 4: Erhöhung der Anzahl von Auswertpunkten durch einseitige Netzverfeinerung

Die qualitative Form des mitwirkenden Bereichs bei einer einwirkenden Flächenlast wird in Bild 54 gezeigt.

2.2. Punktlast

Unter einer Punktlast schnürt sich der mitwirkende Bereich in der Mitte infolge einer Spannungskonzentration unter der Last ein. Untersucht wurden die gleichen Deckengeometrien wie in Abschnitte 2.1. Es wurde nur ein Balken betrachtet. Zusätzlich zur größten auftretenden mitwirkenden Plattenbreite wurde die mitwirkende Plattenbreite in der Systemmitte ausgewertet. Die Größe der aufgetragenen Punktlast wurde aus dem Produkt aus Fläche und Flächenlast ermittelt, auch wenn dies keinen Einfluß auf den Betrag der mitwirkenden Breite hat.

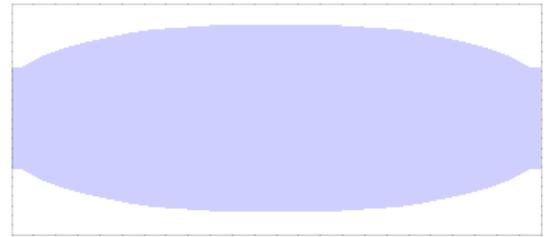


Bild 55: Mitwirkender Bereich bei einer Flächenlast

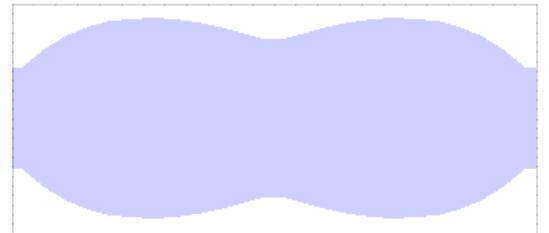


Bild 56: Mitwirkender Bereich bei einer Punktlast

Es ergibt sich ein zu Diagramm 1 gleichwertiger Verlauf, wobei die Werte für die größte mitwirkende Plattenbreite bei einer Punktlast geringfügig höher liegen als die bei einer Flächenlast. Die Werte für die mitwirkende Breite in der Mitte der Systemachse sind dagegen deutlich niedriger und unterschreiten die nach der 1/3-Regel ermittelten Werte.

Geometrie	1 Balken mit $b = 0,4$				Last
	1/3 L	max	mitte	nach Heft 220	
2m x 1m	0,333	0,407	0,317	0,864	-4
2m x 2m	0,666	0,777	0,560	1,280	-8
2m x 3m	1,000	1,135	0,802	1,530	-12
2m x 4m	1,333	1,467	1,006	1,760	-16
2m x 5m	1,666	1,738	1,372	1,862	-20
2m x 6m	2,000	1,742	1,416	1,936	-24
2m x 7m	2,000	1,839	1,491	1,968	-28
2m x 8m	2,000	1,786	1,481	2,000	-32

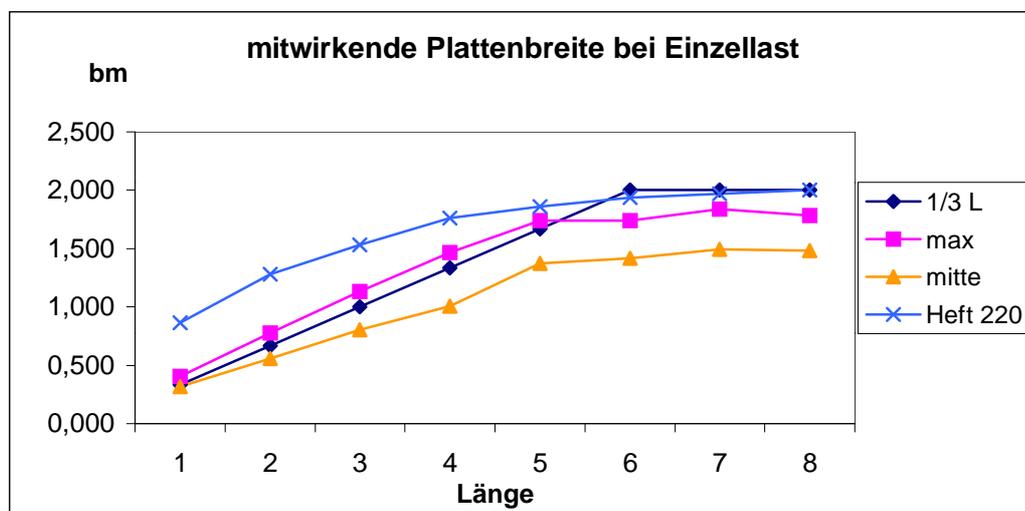


Diagramm 5: mitwirkende Plattenbreite bei einer Einzellast

3. Kragarm

Es wurden wiederum die maximalen mitwirkenden Plattenbreiten bei Flächen- sowie Einzellast untersucht. Um auf die gleichen Vergleichswerte aus Heft 220 DAfStb zurückgreifen zu können wurden die Systemlängen auf $2/3$ verkürzt. Die Punktlast wurde als Resultierende der Flächenlast in Kragarmmitte angeordnet. Auf das umgekehrte Vorzeichen der Last ist zu achten.

Es ergeben sich prinzipiell gleiche Ergebnisse wie in Abschnitt 2. Jedoch liefert in diesem Fall die Flächenlast größere Werte. Diese liegen näher an den Tabellenwerten von Heft 220 DAfStb als beim einfach gelagerten Balken.

Geometrie	1 Balken		
	1/3 L	mit $b = 0,4$ nach Heft 220	
2m x 0,666m	0,333	0,454	0,864
2m x 1,333m	0,666	0,935	1,280
2m x 2,000m	1,000	1,270	1,530
2m x 2,666m	1,333	1,647	1,760
2m x 3,333m	1,666	1,748	1,862
2m x 4,000m	2,000	1,829	1,936
2m x 4,666m	2,000	1,802	1,968
2m x 5,333m	2,000	1,842	2,000

Tabelle 5: Flächenlast

Geometrie	1 Balken		
	1/3 L	mit $b = 0,4$ nach Heft 220	
2m x 0,666m	0,333	0,451	0,864
2m x 1,333m	0,666	0,797	1,280
2m x 2,000m	1,000	1,090	1,530
2m x 2,666m	1,333	1,302	1,760
2m x 3,333m	1,666	1,513	1,862
2m x 4,000m	2,000	1,761	1,936
2m x 4,666m	2,000	1,714	1,968
2m x 5,333m	2,000	1,721	2,000

Tabelle 6: Punktlast in Flächenmitte

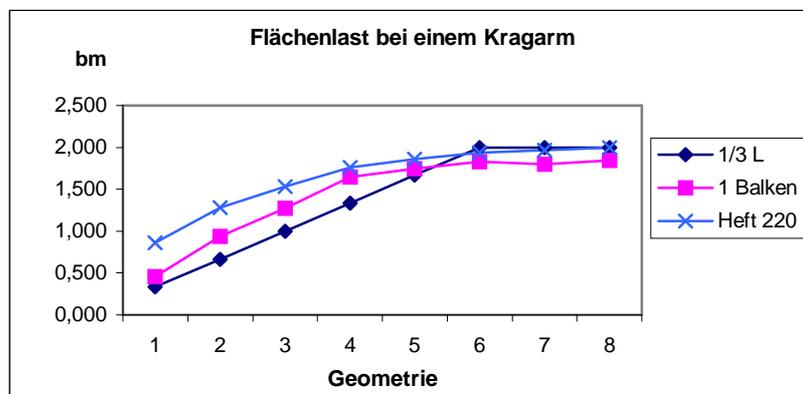


Diagramm 6: Maximale mitwirkende Breite

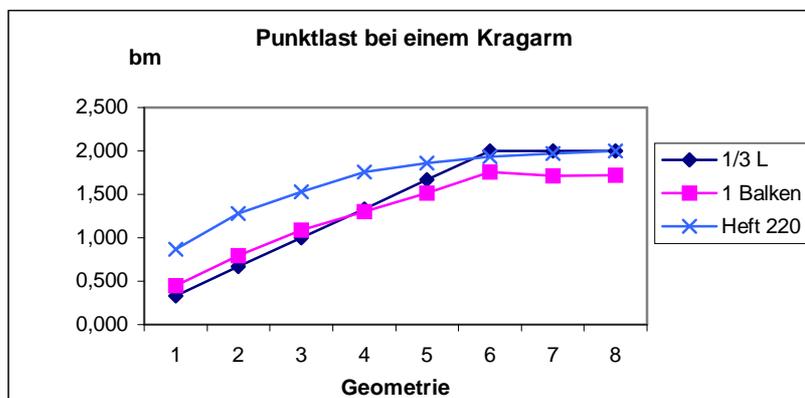


Diagramm 7: Maximale mitwirkende Breite

In der Grundrißansicht ist auffällig, daß an der freien Stirnseite des Balkens bei einer Systemlänge kleiner gleich 3m ein zweiter kleiner mitwirkender Bereich entsteht.

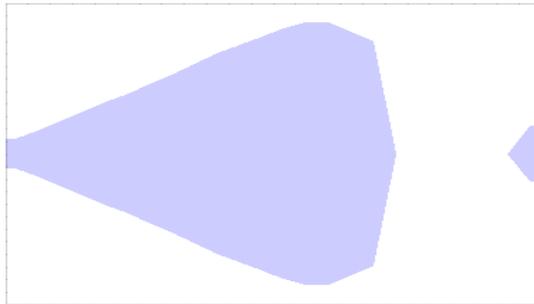


Bild 57: mitwirkender Bereich bei Flächenlast und einer Geometrie von 2mx3m, linker Rand eingespannt

Jedoch verschwindet dieser Bereich bei einer Querdehnungszahl von $\nu=0.0$. Ebenso verschwindet der Bereich bei einer Erhöhung der Plattendicke. Im Fall einer 2mx3m Platte passiert dies bei einer Anhebung der Plattendicke von 0,125m auf 0,243m. Eine Verminderung der Balkensteifigkeit wirkt auf die gleiche Weise.

Bei einer Punktlast tritt der zweite Bereich noch stärker auf. Er verschwindet hier erst bei einer Systemlänge von 9m. Auch das Nullsetzen der Querdehnungszahl bewirkt nur wenig. Bei einer Geometrie von 2mx7m vergrößert dies sogar den zweiten Bereich

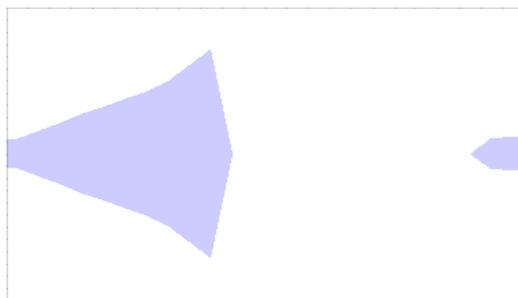


Bild 58: mitwirkender Bereich bei mittiger Punktlast und einer Geometrie von 2mx3m, linker Rand eingespannt

Bei großen Systemlängen löst sich der zweite Bereich vom Rand

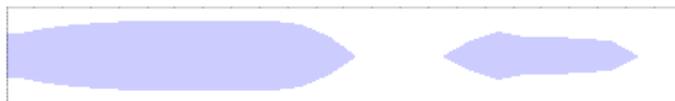


Bild 59: Plattenbalkensystem von 2mx14m, mittige Punktlast, linker Rand eingespannt

Bei einer am Kragarmende angebrachten Punktlast tritt dieses Phänomen nicht auf. Hier erstreckt sich der mitwirkende Bereich über die gesamte Systemlänge. Bild 59 zeigt einen solchen Bereich für eine Geometrie von 2mx4m.

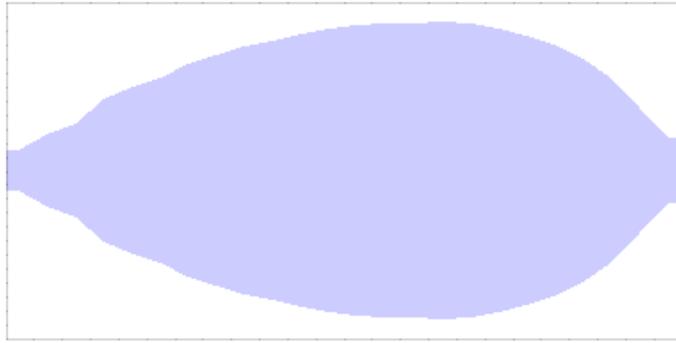


Bild 60: mitwirkender Bereich bei einer am Kragarmende angreifenden Einzellast, linker Rand eingespannt

4. Zweifeldriger Durchlaufträger

Es wird wiederum die maximale mitwirkende Breite infolge Flächen- bzw. Punktlast betrachtet. Analog zu Abschnitt 3 werden die Systemlängen so angepaßt, daß auf die unter Abschnitt 2 angegebenen Vergleichsgrößen zurückgegriffen werden kann. Da es sich hier um zwei Endfelder handelt ist der Korrekturfaktor $1/0,8$. Die gesamte Systemlänge ist durch die Betrachtung eines Durchlaufträgers mit zwei Feldern doppelt so groß wie in den Tabellen angegeben. Deshalb wird in Systemlängsrichtung mit der doppelten Anzahl von Elementen elementiert. Es ergibt sich somit ein Raster von 24×48 Elementen. Beim Lastfall Punktlast wird in beiden Feldmitten eine Last als Resultierende der Flächenlast im betreffenden Feld angeordnet.

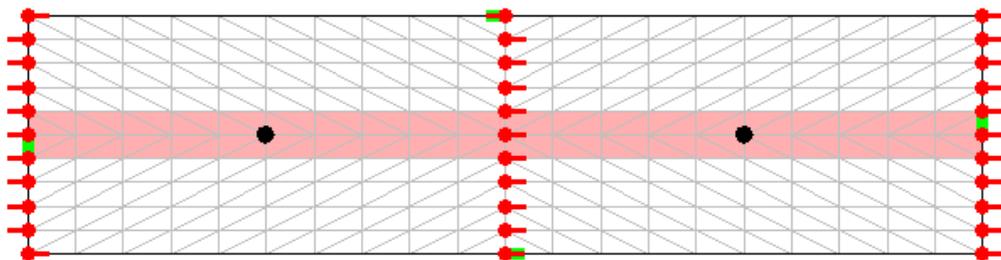


Bild 61: System eines Zweifeldträgers unter Punktlast

In der Grundrißansicht ergibt sich das erwartete Abreißen der mitwirkenden Plattenbreite über der Mittelstützung.

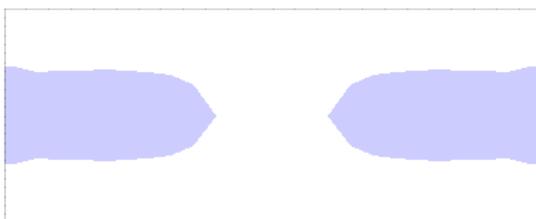


Bild 62: Grundrißansicht bei Flächenlast

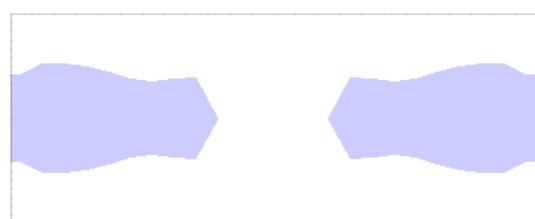


Bild 63: Grundrißansicht bei Punktlast

Gegenüber den bisher betrachteten Deckensystemen ergeben sich ansonsten kaum neue Erkenntnisse. Lediglich bei den Geometrien 6 und 7 ist ein stärkerer Einbruch als zuvor zu erkennen. Dieser wird dadurch verstärkt, daß in diesen Fällen die untersuchte Platte in Querrichtung nur äußerst geringfügig zu schmal ist um zwei weitere Auswertungspunkte für die Ermittlung der mitwirkenden Plattenbreite zu ermöglichen.

Geometrie	1 Balken			
	1/3 L	Flächenlast	Punktlast	nach Heft 220
2m x 1,250m	0,333	0,378	0,455	0,864
2m x 2,500m	0,666	0,691	0,852	1,280
2m x 3,750m	1,000	1,046	1,279	1,530
2m x 5,000m	1,333	1,422	1,670	1,760
2m x 6,250m	1,666	1,585	1,796	1,862
2m x 7,500m	2,000	1,498	1,672	1,936
2m x 8,750m	2,000	1,493	1,607	1,968
2m x 10,000m	2,000	1,722	1,846	2,000

Tabelle 7: Auswertungen für einen zweifeldrigen Durchlaufträger

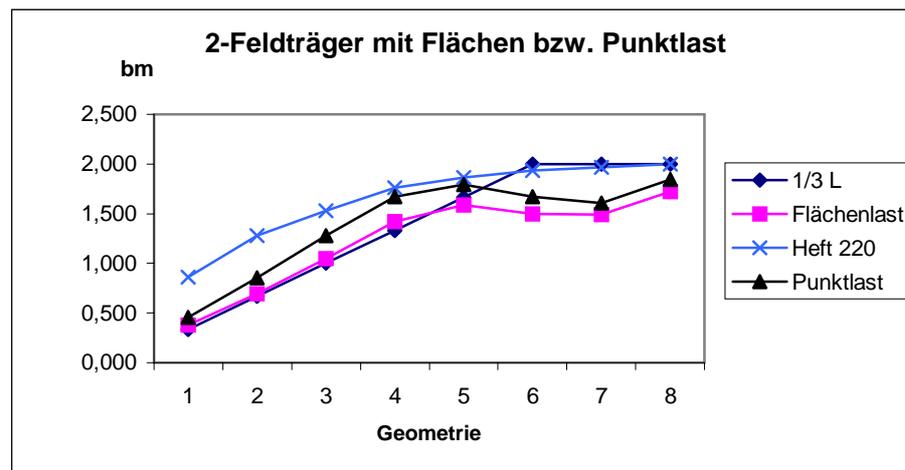


Diagramm 8: mitwirkende Plattenbreiten des zweifeldrigen Durchlaufträgers

5. Dreifeldriger Durchlaufträger

Es wird ein Durchlaufträger mit drei Feldern unter Flächen betrachtet. Die Feldlängen gehen aus den unter Abschnitt 2 verwendeten Geometrien durch die Korrektur mit $1/0,6$ hervor.

Es wird diesmal für die ersten 5 Geometrien mit einem Raster von 12×36 Elementen gearbeitet, bei den letzten 3 Geometrien wird ein Raster von 24×72 Elementen gewählt, da sonst zu wenig Auswertungsstellen für die Spannungsfläche zur Verfügung stehen. Dies bedeutet, das zwar die Verzerrung der Elemente gleichwertig zu den vorher gemachten Untersuchungen ist, jedoch wird in Richtung der Systembreite nur mit der Hälfte der Elemente gearbeitet.

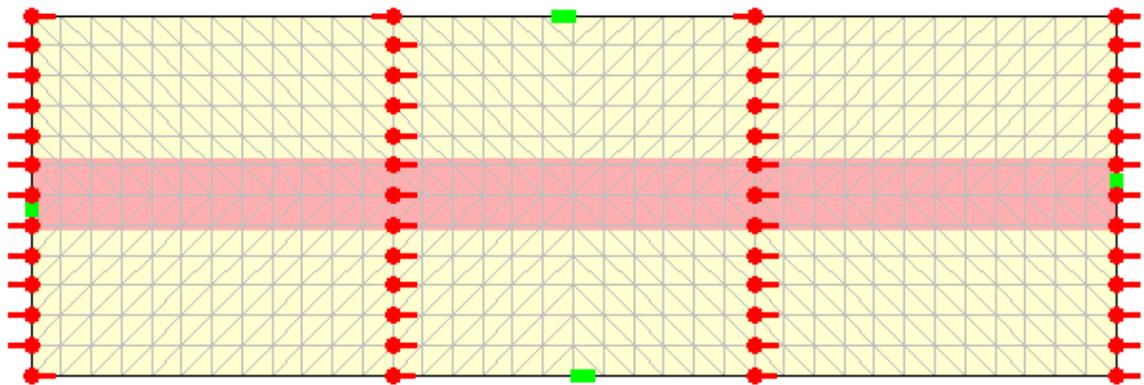


Bild 64: System des Dreifeldträgers unter Flächenlast

Die Ergebnisse sind den nachfolgenden Tabellen und Diagrammen zu entnehmen. Es sind wiederum die Geometrien eines Feldes angegeben.

Geometrie	1/3 L	1 Balken mit $b = 0,4$	nach Heft 220
2m x 1,666m	0,333	0,586	0,864
2m x 3,333m	0,666	0,976	1,280
2m x 5,000m	1,000	1,517	1,530
2m x 6,666m	1,333	1,744	1,760
2m x 8,333m	1,666	1,616	1,862
2m x 10,000m	2,000	1,739	1,936
2m x 11,666m	2,000	1,624	1,968
2m x 13,333m	2,000	1,885	2,000

Tabelle 8: Auswertungsergebnisse für den Dreifeldträger

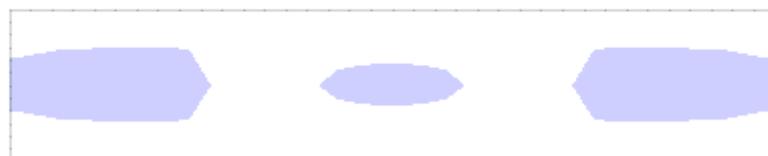


Bild 65: Grundrißansicht des mitwirkenden Bereichs bei den Systemabmaßen 2m x 10m

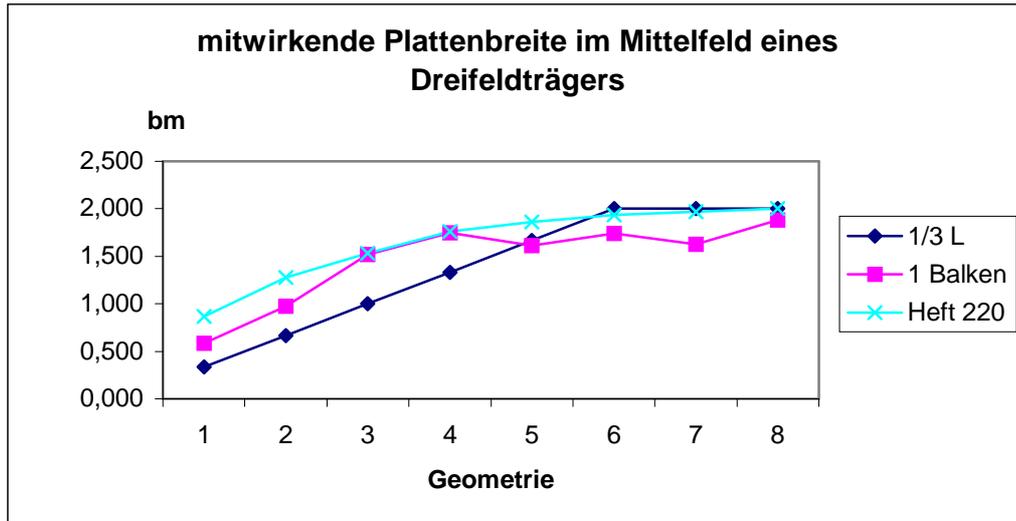


Diagramm 9: maximale mitwirkende Plattenbreite des Mittelfeldes des Dreifeldträgers unter Flächenlast

Zwar ist eine direkte Vergleichbarkeit mit den anderen Berechnungen nicht voll gerechtfertigt, da bei den ersten fünf Geometrien mit anderen Netzen gearbeitet wurde, jedoch ist zu erkennen, daß die Charakteristik des Kurvenverlaufs die gleiche geblieben. Bis zur fünften Geometrie können beide Werte nach Heft 220 als obere und untere Schranke angesehen werden. Nach der fünften Geometrie werden geringere Breiten erreicht.

6. Konvergenz der Durchbiegung

Neben der Bemessung des Stahlbetonquerschnittes ist im konstruktiven Ingenieurbau meistens auch die Kenntnis der Verformungen notwendig. Es wird daher nachfolgend eine Konvergenzuntersuchung bezüglich der Durchbiegung an einem ringsum einfach aufgelagertem Balken mit minimaler Scheibenlagerung gemacht.

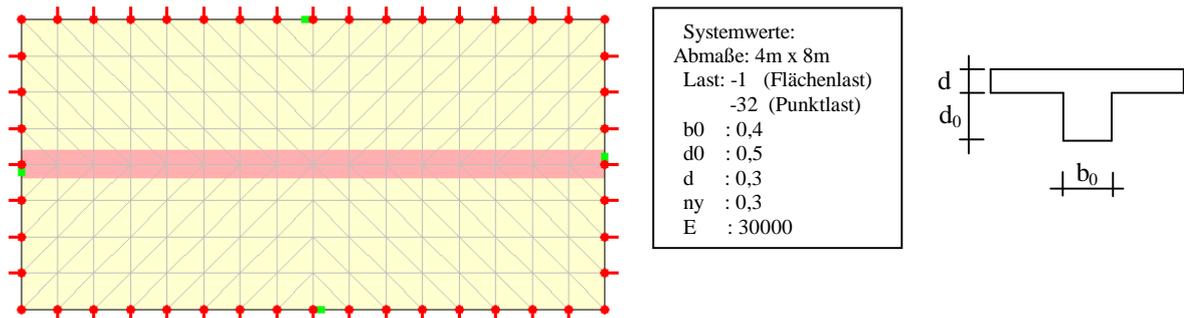


Bild 66: System für die Konvergenzuntersuchung

Elementanzahl	Netz	u3 (Punktlast)	absoluter Fehler	relativer Fehler	u3 (Flächenlast)	absoluter Fehler	relativer Fehler
8	2x4	-0,0887481641	-0,0086488359	10,7976385000	-0,0311914613	-0,0022406890	7,7396519056
32	4x8	-0,0823156021	-0,0022162739	2,7669069764	-0,0294164476	-0,0004656753	1,6085073489
72	6x12	-0,0810046821	-0,0009053539	1,1302890053	-0,0291507394	-0,0001999671	0,6907142163
128	8x16	-0,0805620510	-0,0004627228	0,5776862433	-0,0290619366	-0,0001111643	0,3839769760
200	10x20	-0,0803685684	-0,0002692402	0,3361329065	-0,0290210982	-0,0000703259	0,2429154541
288	12x24	-0,0802698812	-0,0001705530	0,2129268795	-0,0289987653	-0,0000479930	0,1657745068
392	14x28	-0,0802138575	-0,0001145293	0,1429840956	-0,0289851334	-0,0000343611	0,1186880255
512	16x31	-0,0801794266	-0,0000800984	0,0999988412	-0,0289761467	-0,0000253744	0,0876467119
648	18x36	-0,0801569234	-0,0000575952	0,0719047229	-0,0289698738	-0,0000191015	0,0659792416
800	20x40	-0,0801414732	-0,0000421450	0,0526159219	-0,0289652980	-0,0000145257	0,0501737910
1352	26x52	-0,0801160247	-0,0000166965	0,0208447441	-0,0289570188	-0,0000062465	0,0215762811
1800	30x60	-0,0801072843	-0,0000079561	0,0099327924	-0,0289538545	-0,0000030822	0,0106463481
2592	36x72	-0,0800993282	0,0000000000	0,0000000000	-0,0289507723	0,0000000000	0,0000000000

Elementanzahl	Netz	u3 (Punktlast)	absoluter Fehler	relativer Fehler	u3 (Flächenlast)	absoluter Fehler	relativer Fehler
8	2x4	-0,1141139274	0,0002153371	-0,1883481897	-0,0373025692	-0,0022749403	6,4947025290
32	4x8	-0,1134091528	0,0009201117	-0,8047910603	-0,0355144550	-0,0004868261	1,3898345828
72	6x12	-0,1139067762	0,0004224883	-0,3695364453	-0,0353004462	-0,0002728173	0,7788631676
128	8x16	-0,1141477308	0,0001815337	-0,1587814815	-0,0352189746	-0,0001913457	0,5462707754
200	10x20	-0,1142653262	0,0000639383	-0,0559247016	-0,0351735964	-0,0001459675	0,4167210416
288	12x24	-0,1143260884	0,0000031761	-0,0027780289	-0,0351432511	-0,0001156222	0,3300885719
392	14x28	-0,1143575985	-0,0000283340	0,0247828062	-0,0351209643	-0,0000933354	0,2664622269
512	16x31	-0,1143729350	-0,0000436705	0,0381971319	-0,0351036664	-0,0000760375	0,2170786387
648	18x36	-0,1143781951	-0,0000489306	0,0427979662	-0,0350897347	-0,0000621058	0,1773051787
800	20x40	-0,1143793260	-0,0000500615	0,0437871268	-0,0350782215	-0,0000505926	0,1444362681
1352	26x52	-0,1143651998	-0,0000359353	0,0314314101	-0,0350529993	-0,0000253704	0,0724296814
1800	30x60	-0,1143511199	-0,0000218554	0,0191161905	-0,0350410997	-0,0000134708	0,0384576416
2592	36x72	-0,1143292645	0,0000000000	0,0000000000	-0,0350276289	0,0000000000	0,0000000000

Tabelle 9: Berechnungsergebnisse der Konvergenzuntersuchung der Durchbiegung; oben: mit Unterzug, unten: gleiches Deckensystem ohne Unterzug

Auf der nachfolgenden Seite sind die Verläufe der relativen Fehler dargestellt. Als Bezugswert wurde jeweils der Wert bei einer Elementanzahl von 2592 Elementen gewählt.

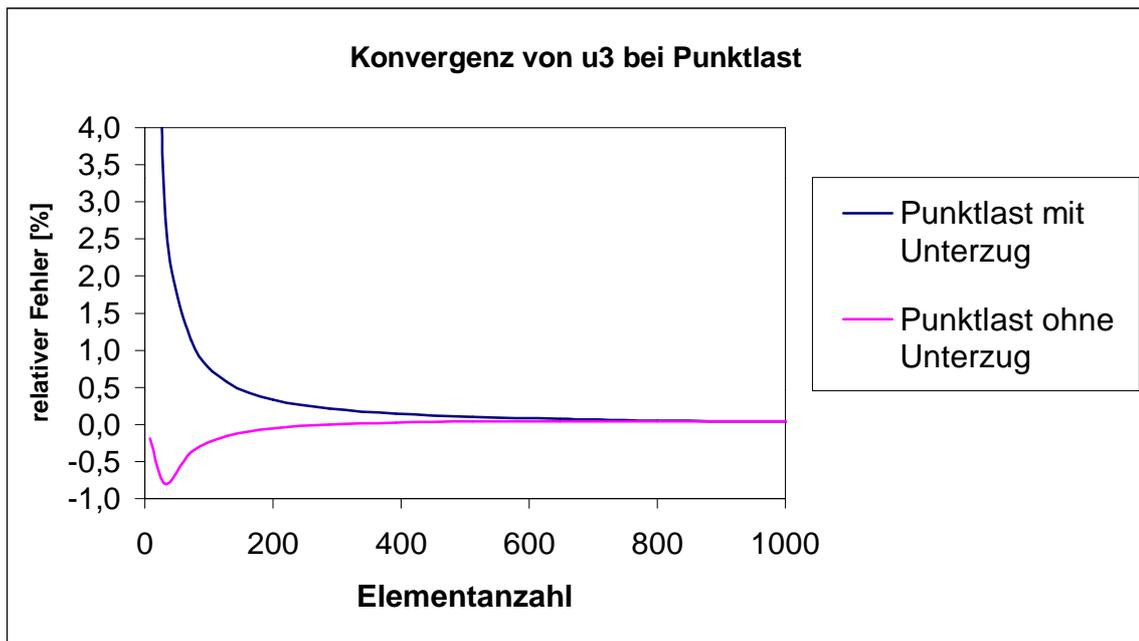


Diagramm 10: Konvergenzkurven des Systems mit und ohne Unterzug bei Punktlast

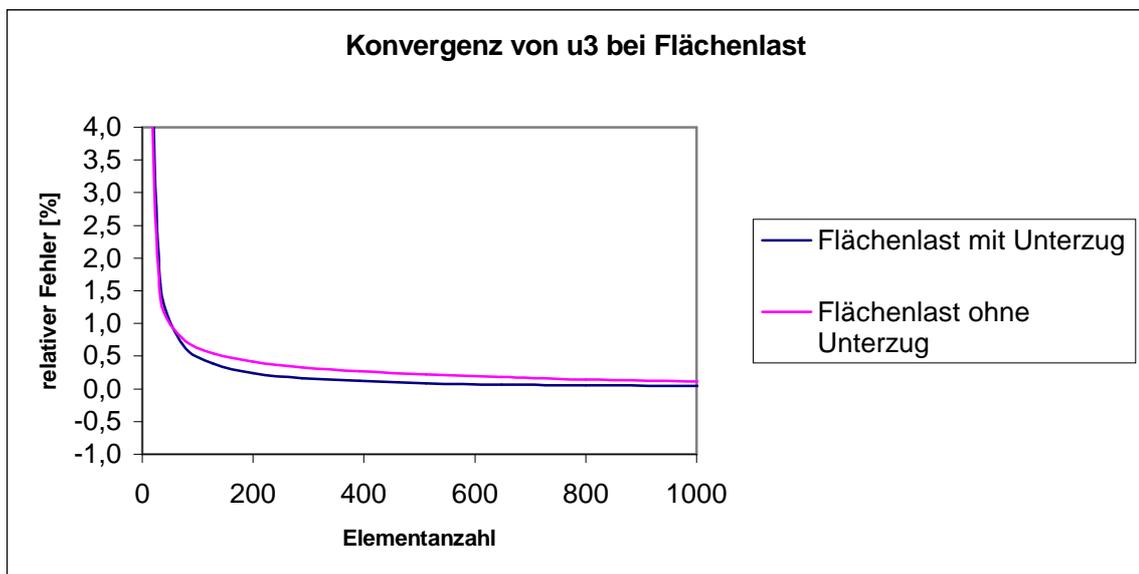


Diagramm 11: Konvergenzkurven des Systems mit und ohne Unterzug bei Flächenlast

Zu sehen ist, daß sich die Kurven der Relativen Fehler bei Vorhandensein einer Flächenlast kaum unterscheiden, was auf ein gleichwertiges Konvergenzverhalten hinweist. In Falle einer Punktlast ist auffällig, daß die Annäherung an den Wert Null von zwei verschiedenen Seiten erfolgt. Zudem scheint die Konvergenz bei dem System mit Unterzug geringfügig langsamer zu verlaufen. Bemerkenswert ist jedoch das bei allen untersuchten Systemen schon bei einem Netz von 4mx8m der relative Fehler unter 3% liegt.

7. Beurteilung

Insgesamt ergeben sich bei der Berechnung der mitwirkenden Plattenbreite mit dem erstellten FEM-Programmpaket trotz zahlreicher Näherungen durchaus sinnvolle Ergebnisse. Die Grundrißansichten sind bis auf das Phänomen des zweiten Druckbereichs beim Kragarm unmittelbar plausibel. Die erzielten Ergebnisse der mitwirkenden Plattenbreite liegen bei den jeweils ersten 5 untersuchten Geometrien jedes Systemtyps mit ihrer Größe zwischen den beiden in Heft 220 angegebenen Werten. Bei größeren Systemlängen ergibt sich eine geringere Ausnutzung des Querschnittes. Dies ist darin begründet, daß zum Erreichen der vollen Systembreite als mitwirkende Breite eine konstante Druckspannungsverteilung im Plattenquerschnitt vorhanden sein muß. Dies ist aber schon aus physikalischen Überlegungen heraus nicht möglich. Es wäre zu prüfen welche Überlegungen den in Heft 220 angeführten Werten und Regelungen in diesem Punkte zu Grunde liegen. Leider konnte keine Literatur dafür gefunden werden. Bemerkenswert ist das nach [14] bezüglich der Festlegung der Werte der mitwirkenden Plattenbreite unter den einzelnen Ländern große Diskrepanzen herrschen. So ist dem dort abgebildetem Verlauf der mitwirkenden Plattenbreite in Abhängigkeit von Systemlänge zu –breite nach den länderspezifischen Vorschriften zu entnehmen, das in Österreich bei einem Verhältnis von $L/b=20$ (b =Breite eines Plattenflügels) erst eine mitwirkende Plattenbreite von ~90% der Systembreite erreicht wird. Dies stimmt in etwa mit den oben erzielten Ergebnissen überein und ist bei einer Betrachtung größerer Systemlänge und weitaus verfeinerten Netzen im begrenztem Maße steigerbar. Das maximale betrachtete L/b Verhältnis war hier $8/((2-0,4)/2)=10$. Hiefür ergibt die in [14] für Österreich angegebene Kurve nur 70%!

Wie gezeigt liegen die mit den Tabellenwerten aus Heft 220 ermittelten mitwirkenden Plattenbreiten in allen untersuchten Fällen über den mit dem FEM-Programm errechneten, so daß ohne nachträgliche Korrektur zwecks Angleichung der Maximalwerte mit dem Heft 220 in der Praxis auch im Hinblick auf die Einfachheit eine Berechnung nach Heft 220 zu empfehlen ist. Der Vorteil des Programmpaketes liegt jedoch darin ein einheitliches Konzept für die Deckenberechnungen zur Verfügung zu stellen, sowie in der Möglichkeit beliebige Geometrien berechnen zu können. Auch ist die Grundrißansicht ein gutes Werkzeug um einen Überblick über das Tragverhalten zu bekommen. So ist z.B. bei der Bemessung eines einfach gelagerten Balkens mit T-Querschnitt zu beachten, daß im Bereich des maximalen Momentes eine Einschnürung der mitwirkenden Plattenbreite vorliegt, wie sie auch in [14] beschrieben wird. [15] gibt für diesen Fall eine Vereinfachte Erfassung durch Abminderung des maximalen Wertes um 40% an. Die im Rahmen dieser Arbeit ermittelten Werte lagen zwischen 20 und 30% unter den Maximalwerten.

Die qualitativen Ergebnisse früherer Arbeiten konnten mit dem entwickelten rechnergestützten Werkzeug nachvollzogen werden. Mit den angegebenen Erweiterungsmöglichkeiten wäre es somit möglich dies zu einem dem herkömmlichen Vorgehen äquivalentem Werkzeug auszubauen (mit den allgemeinen Vor- und Nachteilen eines FEM-Programms)

Kapitel G	Spezifikation der Oberfläche
-----------	------------------------------

Um eine komfortable Handhabung des FEM-Programms sowie eine Visualisierung der Berechnungsergebnisse zu gewährleisten, beinhaltet das Programmpaket eine graphische Benutzeroberfläche. Bereits die Vorgabe enthielt eine solche, an die speziellen Voraussetzungen angepaßte, Oberfläche. Auf diese aufbauend wurde eine Umgebung geschaffen, die es nun auch ermöglicht beliebig geformte Grundrißgeometrien einzugeben und zu berechnen. Es wurden Zugriffe auf die Ergebnisse der zusätzlichen Weggrößen- und Schnittkraftverläufe ermöglicht und spezielle Visualisierungsmöglichkeiten für die mitwirkende Plattenbreite geschaffen.

1. Konzept der Oberfläche

Die Oberfläche besteht aus zwei Teilen. Aus einem Fenster, das die Menüleiste sowie die Standardtoolbox zur Verfügung stellt und einem Inhaltsbereich, indem sich die mit den speziellen Funktionalitäten ausgestatteten Editoren (insgesamt drei) befinden.

Das Hauptfenster wurde einer bereits vorhandenen Implementierung entnommen und ist deshalb vom Typ „ScheibenAss“.

Der Inhaltsbereich wird durch die abstrakte Klasse EditPane repräsentiert, von der sich die einzelnen Editoren ableiten. Sie enthalten Toolboxes für ihre Spezifischen Funktionen, ein Textfeld zur Ausgabe von Hinweisen, sowie einen Darstellungsbereich.

Wie schon erwähnt handelt es sich um drei (grundsätzlich von einander getrennte) Editoren, deren Funktionalität an jeweils einer der Hauptphasen des Prozesses der FEM-Analyse orientiert ist.

- I. Festlegung der Geometrie => Makromodelleditor
- II. Festlegung der Lagerungsbedingung und aufbringen der Belastung sowie Anordnung der Unterzüge => Mikromodelleditor
- III. Berechnung und Visualisierung der Berechnungsergebnisse
=> Tester (FEM-Programm)

Um diesen Prozeß innerhalb einer geschlossenen Einheit durchlaufen zu können werden alle drei Editoren in den Inhaltsbereich des Hauptfensters geöffnet. Die Verfügbarkeit der Editoren hängt dabei von der jeweiligen Bearbeitungsphase ab.

Die Kommunikation aller Bestandteile der Oberfläche läuft über einen außenstehendem Master, der das wechseln der Editoren sowie alle Dateidialoge und die Verwaltung der Menüleiste vornimmt.

Bei Erzeugung eines Objektes des Hauptfensters wird ein Master initialisiert, der wiederum die Editoren erzeugt. Der Master enthält hiernach die Referenzen zu allen Bestandteilen der Oberfläche und kann somit seine Arbeit aufnehmen. Im wesentlichen wartet der Master auf Anfragen zum ausführen von objektübergreifenden Aktionen und greift dann auf die von den Editoren zur Verfügung gestellten Methoden zurück.

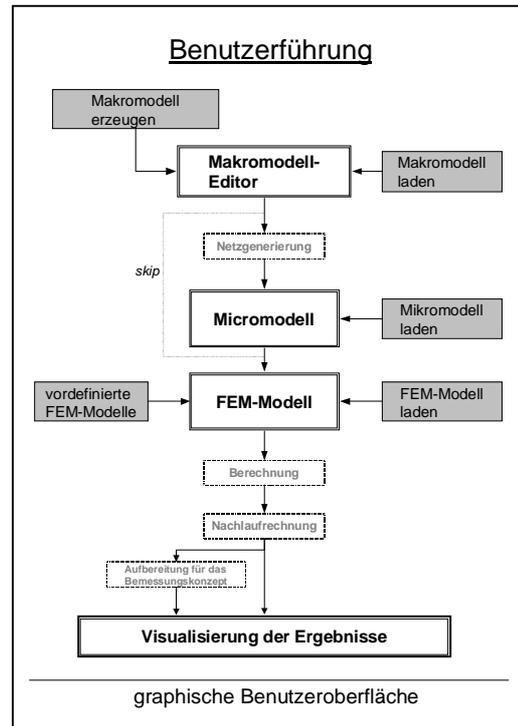
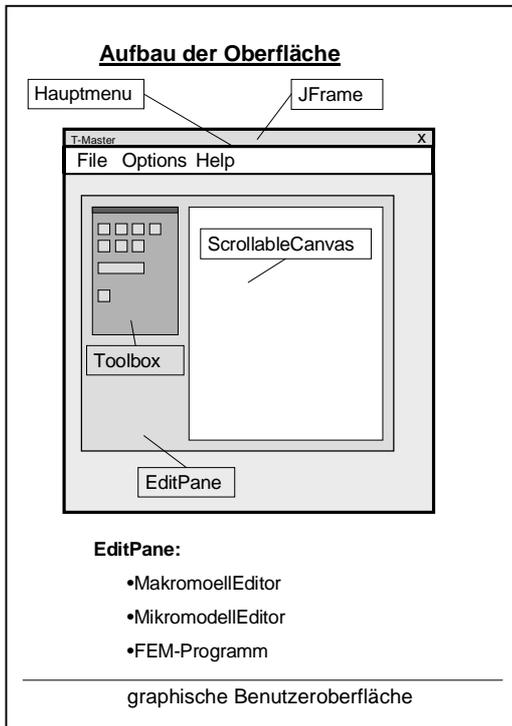


Bild 68: Modularisierung mit Editoren

Bild 67: Ablauf des Berechnungsprozesses

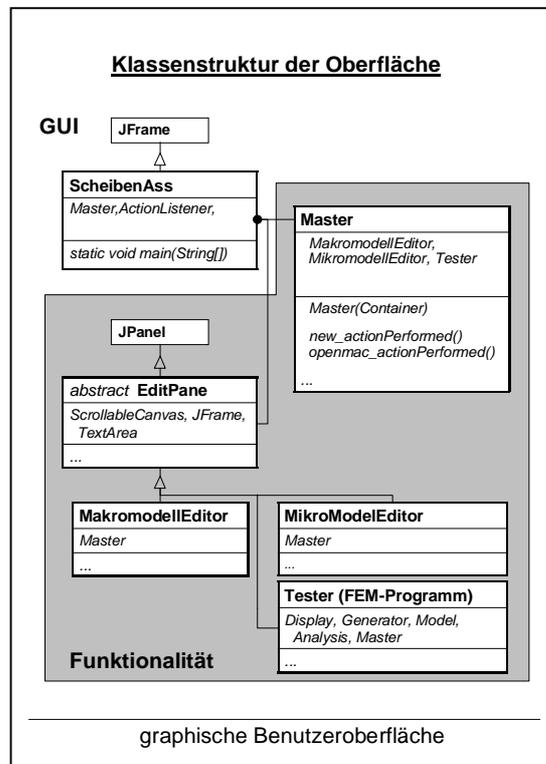


Bild 69: Klassenstruktur

2. Paketstruktur

Eine genaue Unterteilung war aufgrund der Zusammenführung verschiedener Softwareteile schwierig. Die verwendete Paketstruktur ist im rechts stehendem Bild veranschaulicht:

Das Paket Gui besteht nur aus dem Hauptfenster. Das Paket Funktionalität umfaßt den Inhaltsbereich mit allen Editoren sowie weitere Dialogfenster. In dem Paket plate sind alle Klassen die im Zusammenhang mit dem Aufstellen des Gleichungssystems stehen untergebracht. In standard sind die rein elementbezogenen Klassen des FEM-Systems zusammengefaßt. Die Pakete Graph und graphik umfassen die Darstellung wobei graphik zum eigentlichen FEM-Programm gehört (zum dritten Editor) und Graph zu den beiden anderen Editoren. In service sind verschiedene Klassen zum Organisieren einer Applikation, für die Ausgabe, für die Matrizenalgebra, die Formaterstellung usw. enthalten.

Models stellt kein Paket im eigentlichen Sinne dar. Hierbei handelt es sich lediglich um den bevorzugten Ordner zum Lesen und Speichern von Dateien. Er besitzt drei Unterordner für Makro-, Mikro- und FEM-Modelle.

Vom Programmpaket wird dieser Ordner unter dem Pfad :

/Eigene Dateien/Diplomarbeit/Applikationen/Plattenbalken/Models/

erwartet. Es ist empfehlenswert einen solchen Ordner anzulegen, da sonst bei einem Dateidialog immer das Homeverzeichnis angezeigt wird.

Weitere hier nicht erwähnte Pakete sind im Gesamtprogramm enthalten, deren Einbindung ausschließlich durch die Einbindung eines Netzgenerators nötig wurde.

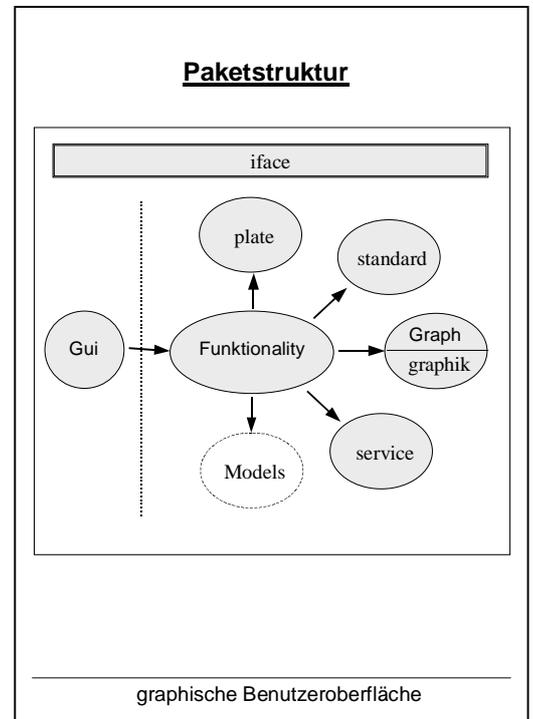


Bild 70: Paketstruktur

3. Leistungsumfang

Der Leistungsumfang der graphischen Benutzeroberfläche „T-Master“ wird in nachfolgenden Bildern zusammengefaßt:

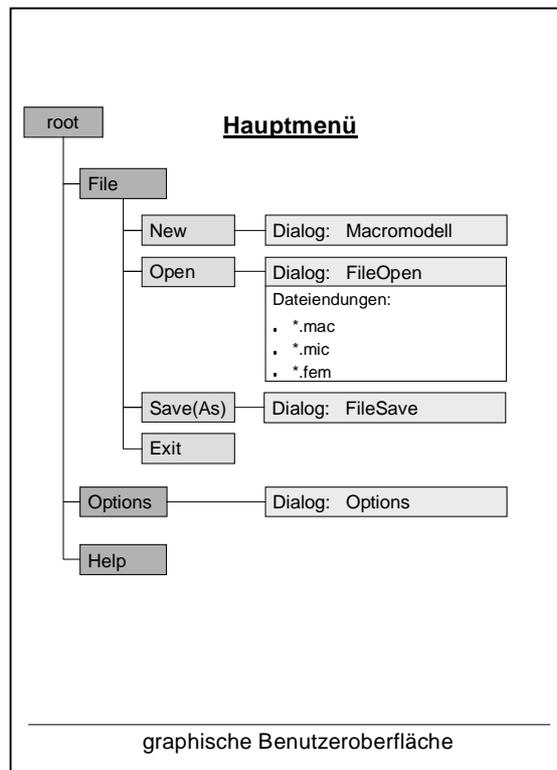
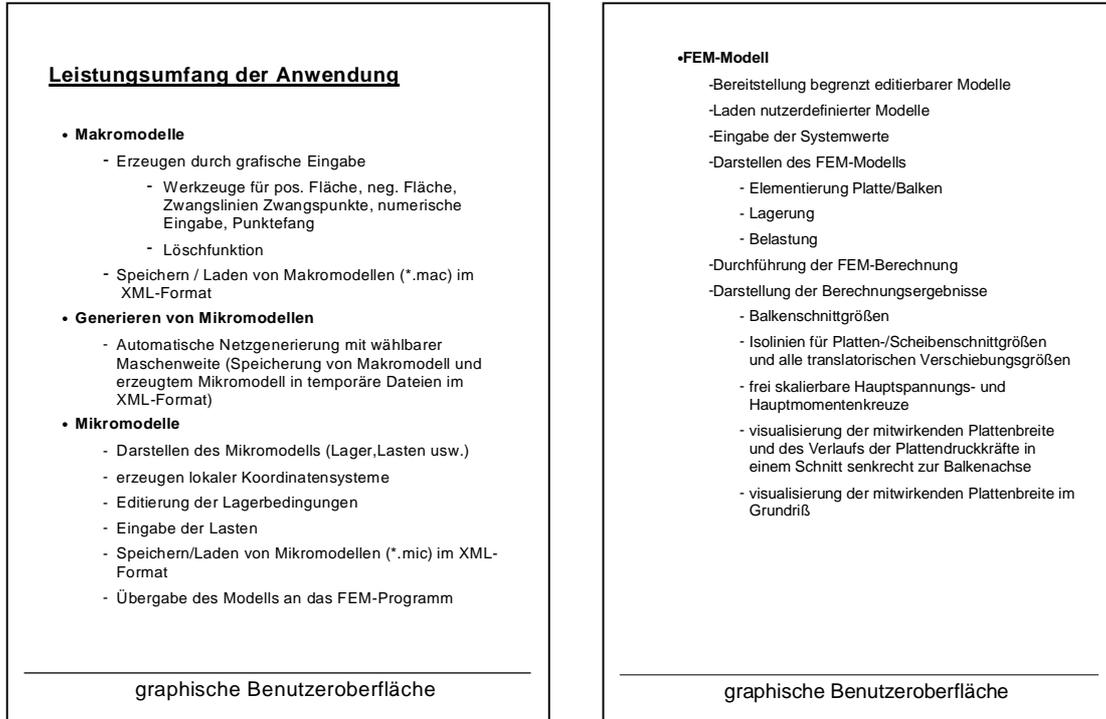


Bild 71: Leistungsumfang der Benutzeroberfläche und Darstellung des Hauptmenüs

3.1. Makromodelleditor

Im Makromodelleditor wird der Umriß des zu berechnenden Systems festgelegt. Es besteht die Möglichkeit eine zusammenhängende Fläche zu definieren, in die nach Bedarf noch Löcher, Zwangslinien und Zwangspunkte für die nachfolgende Netzgenerierung gelegt werden können. Die Eingabe ist dabei einerseits direkt per Mausklick und andererseits über eine numerische Koordinateneingabe möglich.

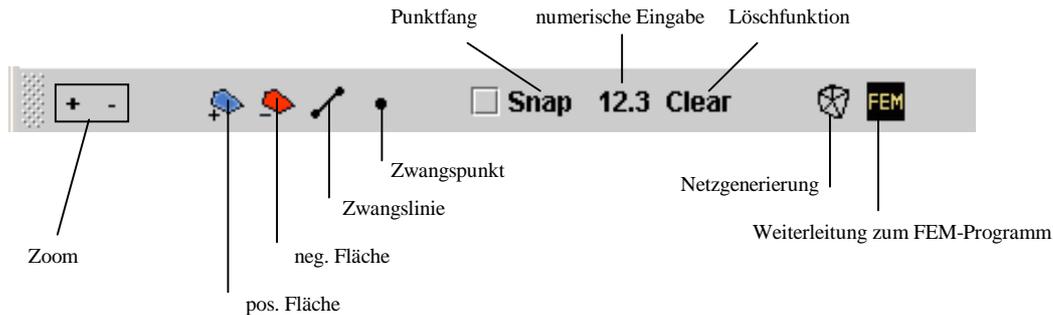


Bild 72: Toolbox des Makromodelleditors

Nachdem auf diese Weise der Umriß des zu berechnenden Systems festgelegt wurde erfolgt durch Betätigung des entsprechenden Knopfes der Toolbox eine automatische Netzgenerierung und Weiterleitung zum Mikromodelleditor.



Bild 73: Dialog zur Maschenbreite

Unmittelbar vor Start des Netzgenerators wird die gewünschte Maschenbreite des FE-Netzes abgefragt.

Weiterhin steht ein FEM-Knopf zur Verfügung, bei dessen Betätigung eine sofortige Weiterleitung zum FEM-Programm erfolgt. Dies ist insbesondere dann sinnvoll, wenn keine eigene Systemeingabe erfolgen soll, da auf ein bereits vordefiniertes Modell zurückgegriffen wird.

Zur Steuerung der Ansicht steht eine Zoomfunktion zur Verfügung, die den aktuellen Skalierfaktor verdoppelt oder halbiert. Für den Zeichenbereich stehen Scrollbars zur Verfügung.

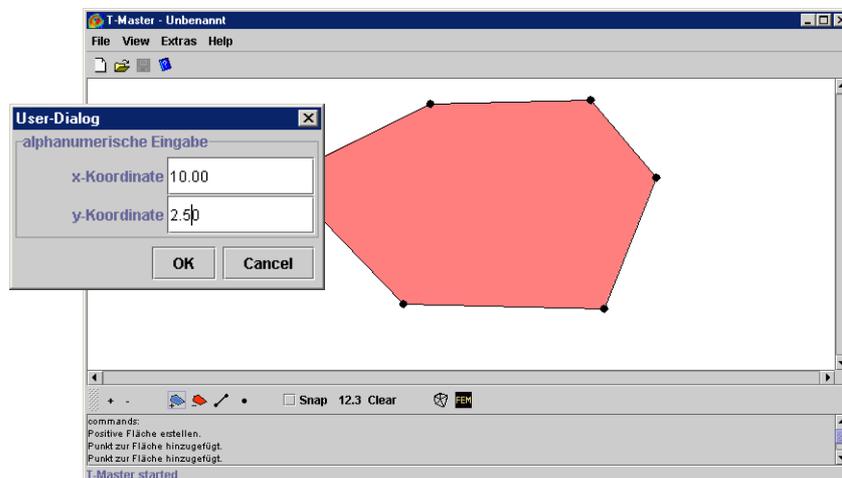


Bild 74: Makromodelleditor mit Dialog für die numerische Eingabe

3.2 Mikromodelleditor

Nach dem Generieren des FE-Netzes für ein eingegebenes Makromodell erfolgt die Eingabe der Lagerungsbedingungen, Lasten und Unterzüge im Mikromodelleditor. Das vom Generator erhaltene Netz wird automatisch eingelesen. Die Übergabe Makromodelleditor -> Netzgenerator -> Mikromodelleditor erfolgt dabei über temporäre XML-Dateien. Die Toolbox stellt zwei Werkzeuge zum Editieren bereit. Der Zeiger ist von vornherein aktiviert und ermöglicht das Fangen von Punkten, Linien und Dreiecken. Auf Betätigung der rechten Maustaste hin öffnet sich ein Kontextmenü, indem die jeweiligen Einstellungen vorgenommen werden können. Bei einem Dreieck ist hierbei nur die Eingabe einer Flächenlast möglich. Bei einem Punkt können die fünf Freiheitsgrade gesperrt, sowie Punktlasten definiert werden. Um die Definition der Lagerungen auch relativ zu beliebig liegenden Auflagerlinien eingeben zu können ist es möglich ein lokales Koordinatensystem zu definieren. Dies kann durch Eingabe eines Winkel relativ zu den Bildschirmachsen, durch interaktive Auswahl, sowie durch Übernahme von anderen Punkten erfolgen.

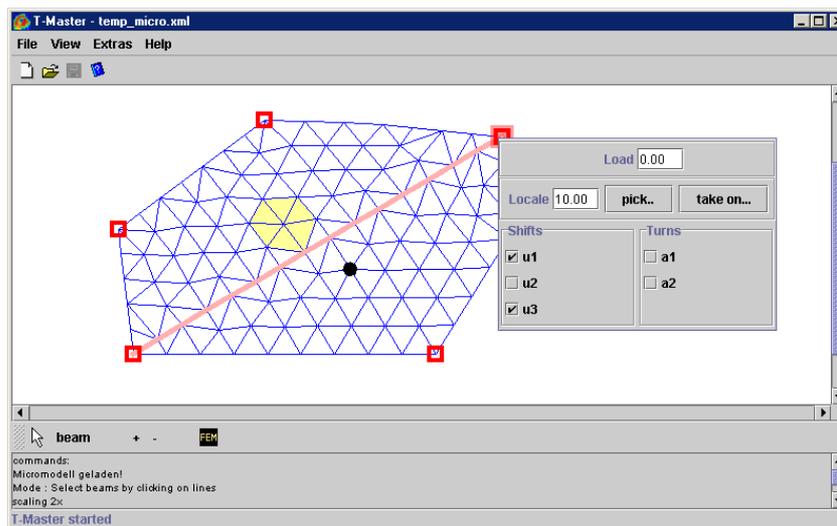


Bild 75: Mikromodelleditor mit Punktkontextmenü

Das Werkzeug „beam“ ermöglicht die Definition von Unterzügen. Dafür wurden im allgemeinen im Makromodell Zwangslinien vorgesehen, die vom Netzgenerator beachtet wurden. Diese müssen nun angeklickt werden um sie als Unterzug zu identifizieren. Dabei ist eine zusammenhängende Reihenfolge zu beachten. Es kann nur ein Unterzug eingegeben werden, jedes Eingegebene Balkenelement wird auf Zusammenhang mit dem zuvor eingegebenen überprüft. Durch Auswahl eines bereits festgelegten Balkenelementes ist das Löschen des ganzen Strangs möglich. Die Systemwerte von Unterzug und Platte werden, um eine leichte Änderungsmöglichkeit zu haben, erst im FEM-Programm festgelegt. Mit dem FEM-Knopf erfolgt die Übergabe des fertigen Modells an das FEM-Programm. Zuvor wird der Nutzer zum speichern des Modells aufgefordert. Ansonsten ähnelt der Aufbau des Mikromodelleditor dem des Makromodelleditor.

Hinweis: Es wurde ein am Fachbereich bereits vorhandener Netzgenerator verwendet. Dieser beruht auf die Verzerrung eines zunächst gleichmäßigen Netzes aus gleichseitigen Dreiecken. Im Rahmen der durchgeführten Programmtests mußte jedoch festgestellt werden, das dessen

Ergebnisse oft fehlerhaft sind. Insbesondere scheinen horizontale Zwangslinien nicht möglich zu sein. Auch fehlende und sich überschneidende Elemente können auftreten. Eine Behebung dieser Fehler ist oftmals durch eine Verfeinerung der Maschenweite möglich. Auch Programmabbrüche treten auf.

3.3. FEM-Programm

Beim FEM-Programm handelt es sich um das erweiterte Programm der Vorgabe. Es wurde so aufgearbeitet, daß es als eigenständiger Editor in die beschriebene Struktur der Oberfläche eingearbeitet werden konnte. Es stehen nun vier Steuerungspanels (Show, Options, Systemvalues, Viewcontrol) zur Verfügung. Im Optionspanel kann mit Hilfe einer Combobox zwischen verschiedenen vordefinierten Modellen gewählt werden. Es stehen die in Kapitel D und F verwendeten Testbeispiele zur Verfügung. Weiterhin ist der Generator des vorgegebenen Programms in Originalform und mit doppelt symmetrischem Netz integriert. Die Lagerungsarten der in Kapitel F verwendeten Beispiele können mit einer weiteren Combobox eingestellt werden. Mit dem Eintrag user spezifizied model ist es möglich eigene Modelle in den Editor zu laden.

Die dem jeweiligen Modell zugeordneten Systemwerte können im Panel Systemvalues eingegeben werden, dessen Funktionalität gegenüber der Vorgabe im Punkte Benutzerfreundlichkeit erweitert wurde.

Das Panel Viewcontrol ermöglicht die Steuerung der Darstellung mit Hilfe von Layern. Gegenüber der Vorgabe wurden Layer für Lasten, Lager und Balken aufgenommen. Die Lagerbedingungen werden bezüglich des am Punkt vorhandenen Koordinatensystems dargestellt.

Es wurde folgende Symbolik für die Visualisierung der Lagerbedingungen gewählt:

roter Punkt	die Verschiebung u_3 ist gesperrt
roter Strich	die Verdrehung um die zum Symbol parallele Achse ist gesperrt
grüner Strich	die Verschiebung in Richtung der zum Symbol parallelen Achse ist gesperrt

Die Farben Rot und Grün ermöglichen gleichzeitig eine Unterscheidung in Platten- und Scheibenfreiheitsgrade.

Über das Panel Show ist die Wahl der anzuzeigenden Berechnungsgröße möglich. Die Vorgabe wurde hier um die Scheibengrößen erweitert sowie ein „Facelifting“ vorgenommen.

Die Freigabe der einzelnen Funktionen geschieht situations- und modellabhängig.

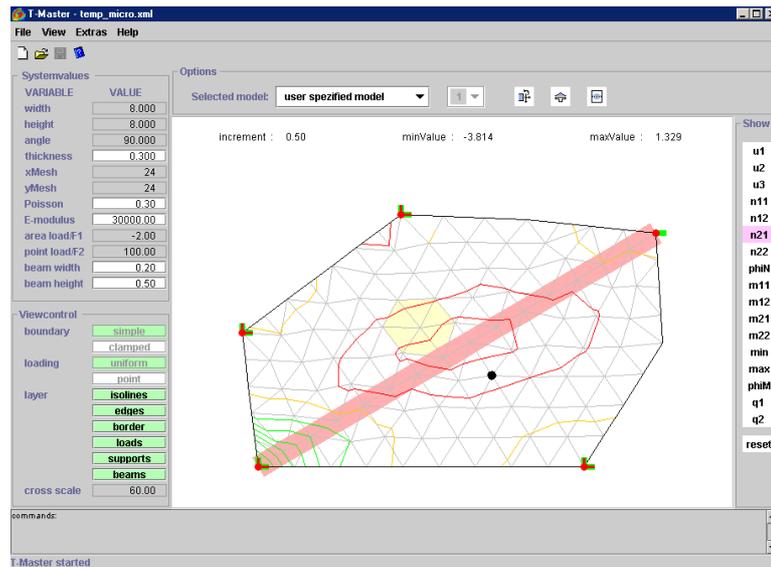


Bild 76: FEM-Programm als dritter Editor

Für die Darstellung der Hauptmomenten- und insbesondere der Hauptspannungskreuze ist über das Eingabefeld „cross scale“ eine freie Skalierung möglich. Im Gegensatz zur Vorgabe erfolgt die Darstellung der Kreuze proportional zum vorhandenen Wert. Dadurch wird ein besseres Verständnis der Tragwirkung ermöglicht.

Zur Aktivierung der zusätzlichen Visualisierungsmöglichkeiten für die Balkenschnittgrößen, Plattendruckkräfte im Schnitt und die mitwirkende Plattenbreite stehen im Optionpanel Knöpfe zur Verfügung. Ihre Betätigung öffnet ein neues Fenster mit der gewünschten Darstellung. Bei den Balkenschnittgrößen und der Darstellung eines Schnittes ist dafür zunächst die Auswahl eines Balkenelementes notwendig.

Bei den Balkenschnittgrößen erfolgt die Darstellung der Verläufe von Moment, Querkraft und Normalkraft im klassischen System der Baustatik. Die Orientierung der x-Achse ist dabei von der Anordnung der Balkenelemente abhängig.

Ein senkrecht zum Balken verlaufender Schnitt wird mit dem Blick entlang der Richtung der Balkenachse dargestellt. Zu beachten ist, dass die dargestellte Platte in ihrer Breite nicht der wirklich vorhandenen Platte entspricht, sondern der Breite des Auswertungsbereiches. Die mitwirkende Plattenbreite wird im Schnitt mit dargestellt. Ein Überblick über den mitwirkenden Bereich der Platte liefert die Visualisierung in der Grundrißebene. Hierbei ist zu beachten, daß bei Randberührung die Darstellung des mitwirkenden Bereichs wiederum nur bis zum Rand des Auswertungsbereichs erfolgt.

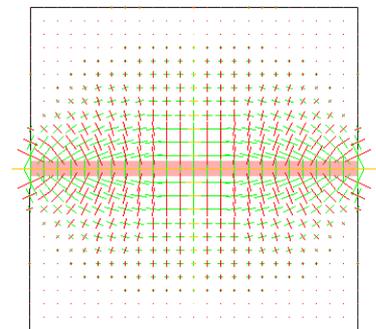


Bild 77: Darstellung der Hauptspannungen in der Platte

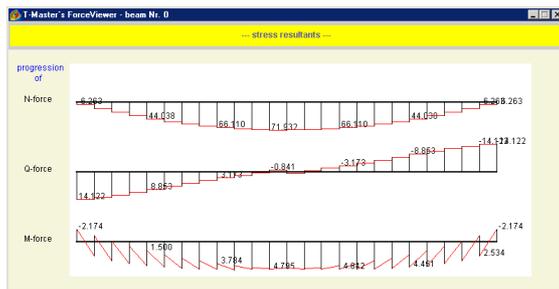


Bild 78: Visualisierung der Balkenschnittgrößen

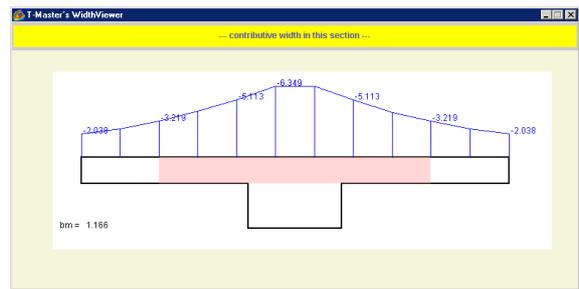


Bild 79: Schnitt senkrecht zur Balkenachse mit Verlauf der Plattendruckkraft und mitwirkender Plattenbreite

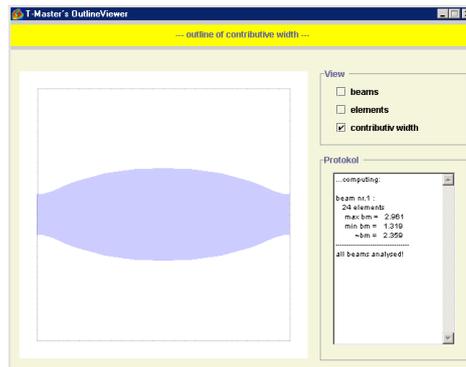


Bild 80: Mitwirkender Plattenbereich in der Grundrißansicht

4. Erweiterungsmöglichkeiten

Die Oberfläche enthält alle notwendigen Werkzeuge zur FEM-Analyse balkengestützter Deckensysteme. Jedoch sind bezüglich Benutzerfreundlichkeit noch etliche Erweiterungen denkbar. Besonders im Makromodelleditor wäre eine gezielte nachträgliche Bearbeitung einzelner Punkte und eine differenziertere Löschfunktion wünschenswert. Im Mikromodelleditor wäre eine Mehrfachauswahl sinnvoll.

Des Weiteren ist prinzipiell eine Verarbeitung mehrerer Unterzüge in ein und dem selben System möglich. Jedoch entstehen dabei Probleme bei der Berechnung der mitwirkenden Plattenbreite, da die Zuordnung der Plattendruckspannungen zu den jeweiligen Balken nicht von vornherein fest steht. Insbesondere bei nicht parallelen Balken ergibt sich dadurch die Problematik der Festlegung der Integrationsgrenzen bei der Bestimmung der Spannungsfläche. Ein Hinweis auf die Zuordnung könnte hier das Feld der Hauptspannungskreuze mit den ovalen Druckspannungsbereichen liefern.

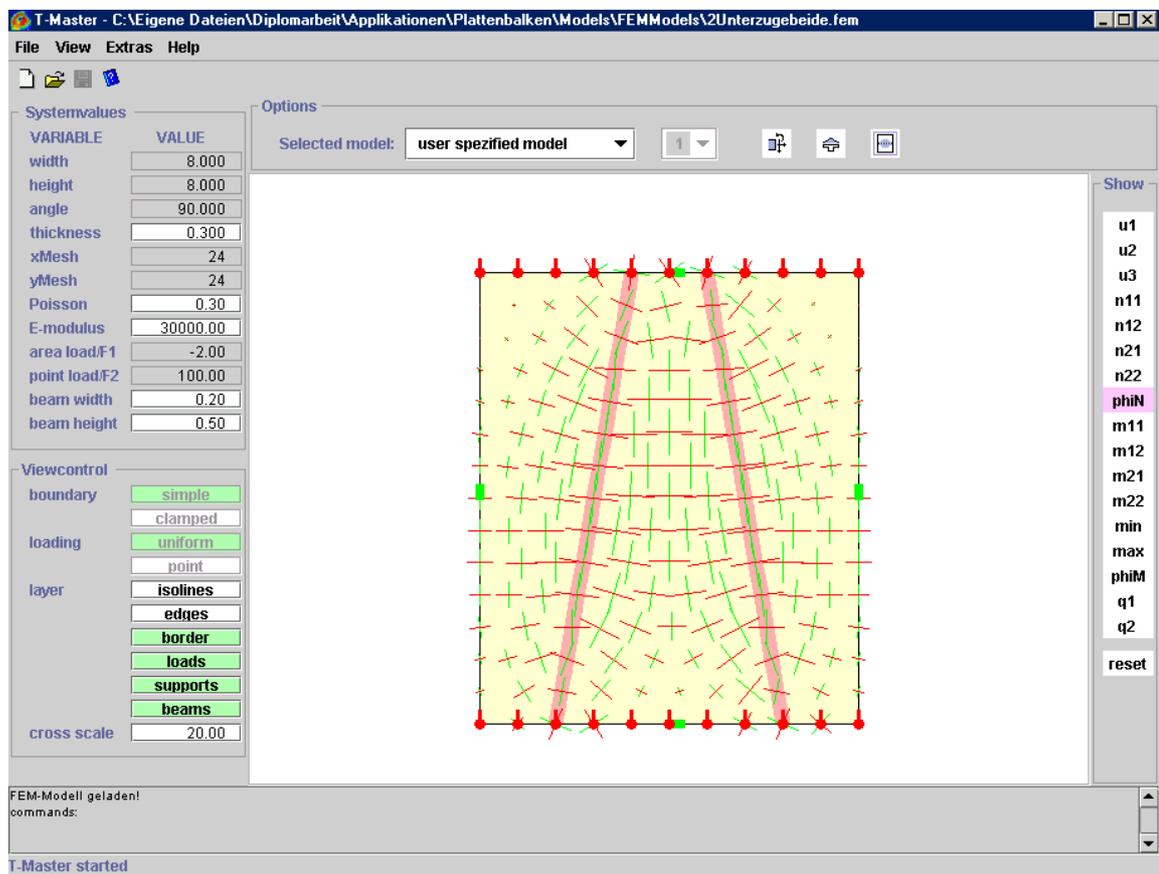


Bild 81: zwei nicht parallele Unterzüge mit gleichen Abmessungen bei einfach gelagertem Balken unter Flächenlast

Derzeit berechnet das Programm für jeden der beiden Unterzüge die mitwirkende Plattenbreite so als ob der jeweils andere Balken nicht vorhanden wäre. Dies ist in der Grundrißansicht an der Überschneidung der mitwirkenden Bereiche zu erkennen.

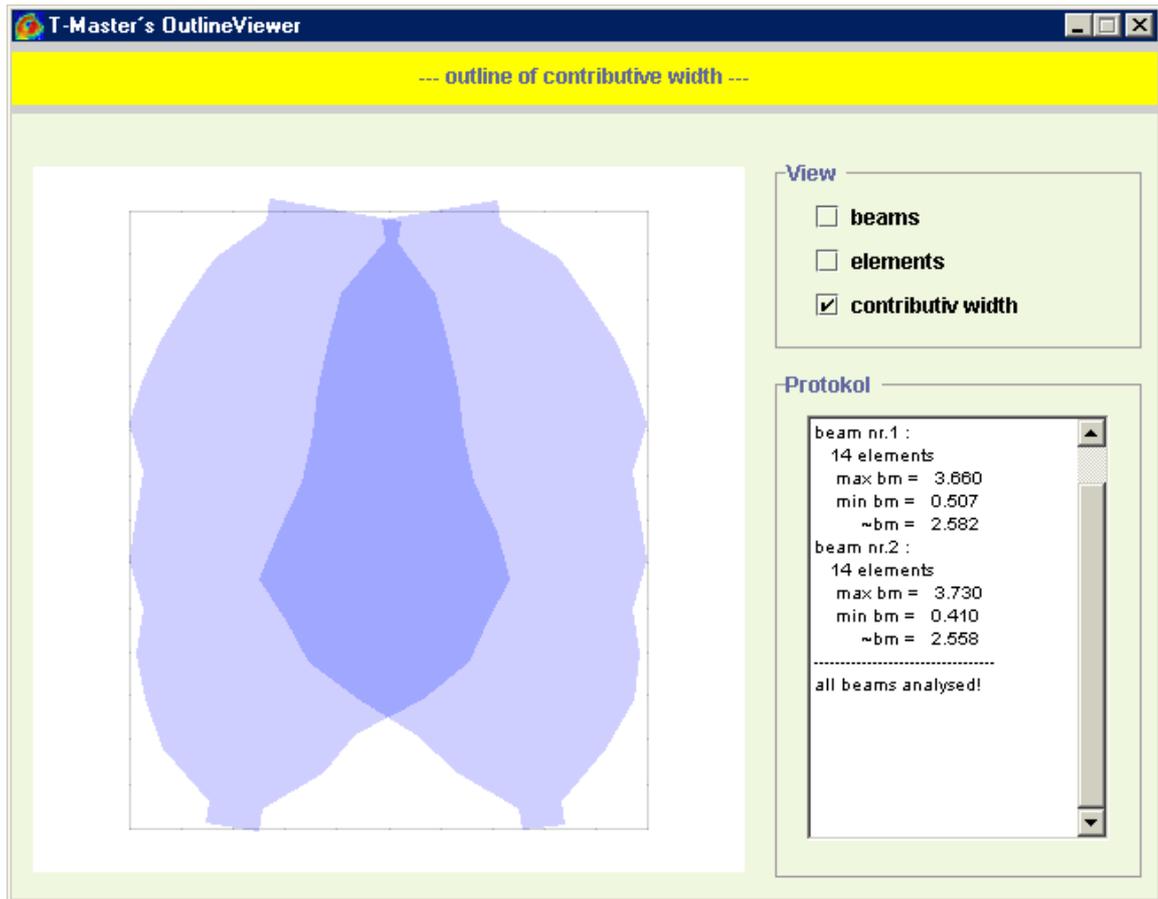


Bild 82: Überschneidung des mitwirkenden Bereichs beim Vorhandensein zweier nicht paralleler Unterzüge

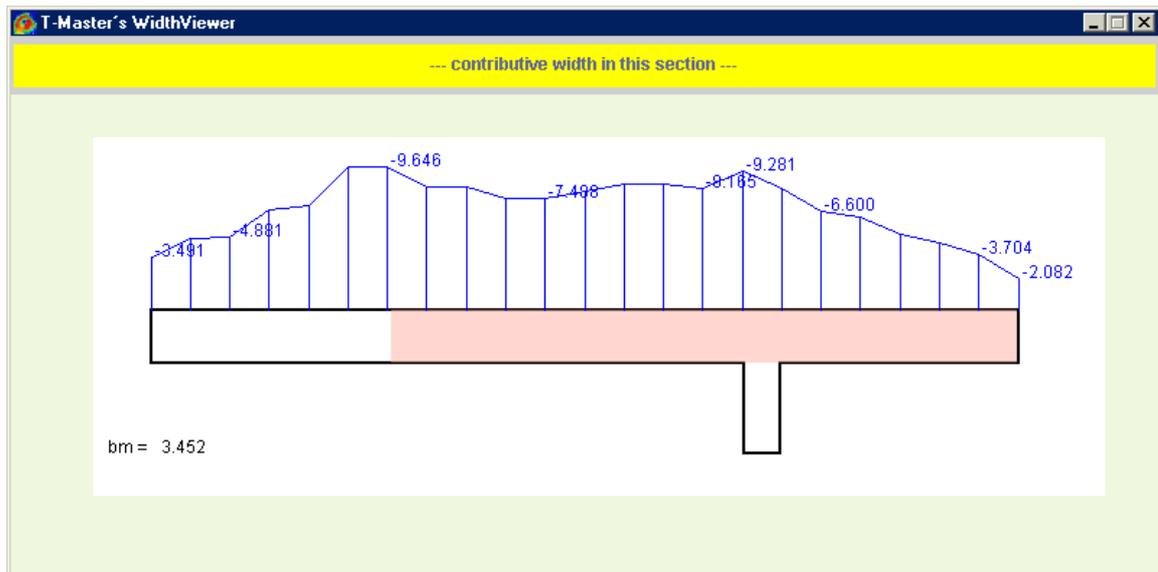


Bild 83: Schnitt für den rechten der beiden Unterzüge in Systemmitte

Da diese Problematik der Aufteilung der mitwirkenden Plattenbreite im derzeitigen Programmpaket keine Berücksichtigung findet wurde in der Oberfläche auf der Seite der Eingabe eine Beschränkung auf nur einen Balkenstrang vorgenommen. Über die .fem-Dateien im XML-Format können jedoch mehrere Balken erzeugt werden. Da die Abmaße der Unterzüge nicht mit in die XML-Datei aufgenommen wurden ist dann jedoch nur eine Berechnung mit gleichen Abmaßen für alle Unterzüge möglich.

Ebenso ist im FEM-Programm die Verarbeitung affiner lokaler Koordinatensysteme vorgesehen. Diese wird jedoch der Einfachheit halber in der Oberfläche zur Zeit nicht angeboten.

Zahlreiche Erweiterungsmöglichkeiten ergeben sich auch bei den Lasten und Lagerungsbedingungen. So wäre die Implementierung von Linienlasten, Federlagern und Auflagersenkungen denkbar.

Auch die in der Oberfläche vorgesehene Anbindung an ein Hilfesystem (z.B. Java-Help 1.1) wäre eine sinnvolle Erweiterung.

Bisher wurden nur der Bestimmung der mitwirkenden Plattenbreite Aufmerksamkeit geschenkt. Die für die Bemessung des Querschnittes ebenfalls wichtigen Gesamtschnittgrößen des T-Querschnittes wurden nicht ermittelt.

Im Zuge einer vollständigen rechnergestützten Berechnung von Balken-Deckensystemen wäre dies allerdings wünschenswert.

Die Kraftgrößen liegen nach der Berechnung getrennt nach der Zuordnung zu den beiden Tragelementen Platte/Scheibe und Balken vor und müssen nachträglich vereint werden. In [16] werden dabei die Gesamtschnittgrößen für das Modell ebenes Faltelement mit exzentrischem Balken im Sinne des Bemessungsmodells wie folgt angegeben:

$$M_{pb} = M_B + N_B \cdot e + \int_D m_{yy} dx$$

$$Q_{pb} = Q_B + \int_D q_y dx$$

„Die Integrale ... sind in einem Bereich beidseits der Systemlinie des Plattenbalkens auszuwerten, der näherungsweise durch den Vorzeichenwechsel der Integranden (m_{yy}) oder durch eine zweckmäßig festgelegte mitwirkende Breite bestimmt ist.“

Zu erkennen ist das diese Kraftgrößen auf die Plattenmittelebene bezogen sind! Eine Visualisierung der so gewonnenen Kraftgrößen entlang der Balkenachse wäre eine sinnvolle Erweiterung und würde Zusätzlich zur mitwirkenden Plattenbreite die zweite Eingangsgröße für die Bemessung liefern.

5. Anwendungsbeispiel

Nachfolgend wird an einem typischen System eine Beispielrechnung durchgeführt.

Nach dem Start des Programms durch Eingabe der Kommandozeile

```
java Gui.ScheibenAss
```

(der Nutzer muß sich im den Paketen übergeordneten Ordner befinden) öffnet sich der Makromodelleditor mit einem neuen noch unbenannten Modell. Die Eingabe einer positiven Fläche kann durch Betätigung des entsprechenden Knopfes der Toolbox sofort begonnen werden. Um eine numerische Eingabe zu ermöglichen wird zusätzlich der Knopf „12.3“ betätigt. Es öffnet sich daraufhin der entsprechende Dialog in dessen Textfelder die Koordinaten des ersten Punktes eingegeben werden können. In diesem Fall soll das der schon angegebene Punkt (0.00, 0.00) sein. Die Eingabe kann durch Drücken der ENTER-Taste oder durch Betätigung des OK-Knopfes geschehen. Daraufhin wird der erste Punkt in der Zeichenfläche sichtbar. Das Dialogfeld der numerischen Eingabe bleibt geöffnet. Auf diese Weise wird die im nachfolgenden Bild gezeigte Fläche eingegeben. Zu beachten ist, daß für die Verwendung des eingebundenen Netzgenerators eine Eingabe der (positiven und negativen) Flächenelemente gegen den Uhrzeigersinn erfolgen muß.

Die Eingabe erfolgt bis zum letzten Punkt, danach wird zuerst die numerische Eingabe durch drücken der ESC-Taste oder des CANCEL-Knopfes des Dialoges und danach die Eingabe der positiven Fläche beendet. Zweiteres geschieht wiederum durch Drücken der ESC-Taste oder durch Betätigung der rechten Maustaste. Nach Abbruch beider Funktionen wird das Polygon automatisch zur abgebildeten Fläche geschlossen.

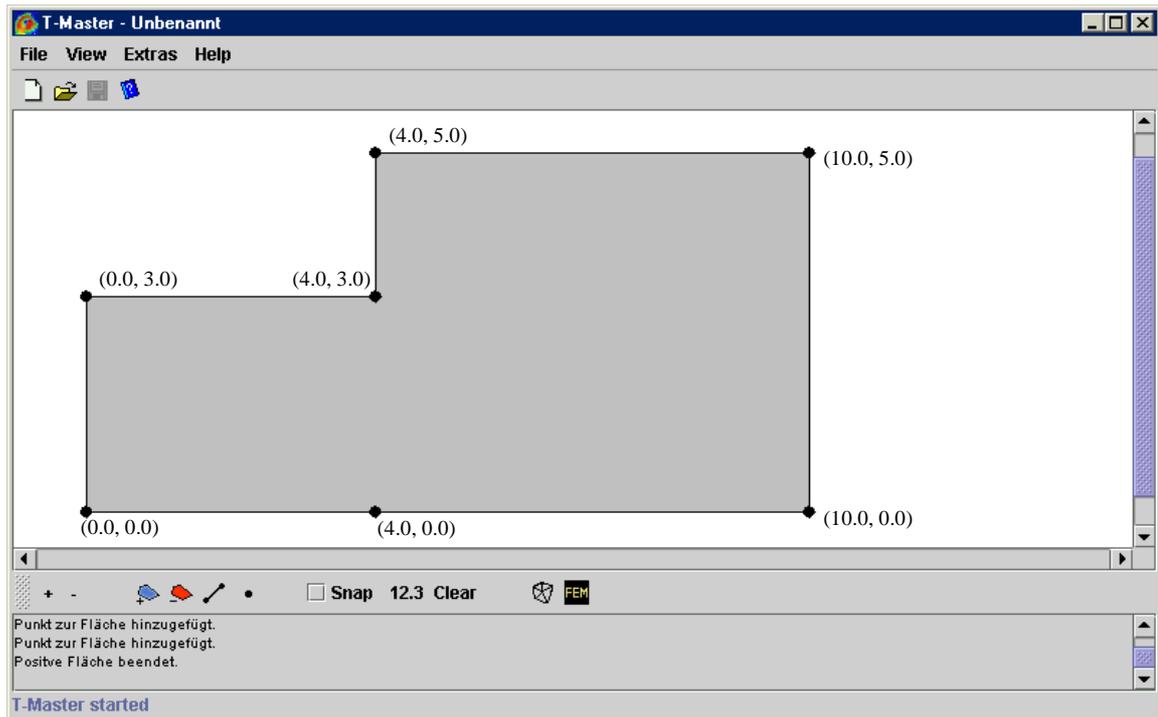


Bild 84: Positive Fläche des Anwendungsbeispiels

Vollkommen Analog wird nun mit Hilfe der Funktion „negative Fläche“ ein Loch in die Deckenplatte geschnitten.

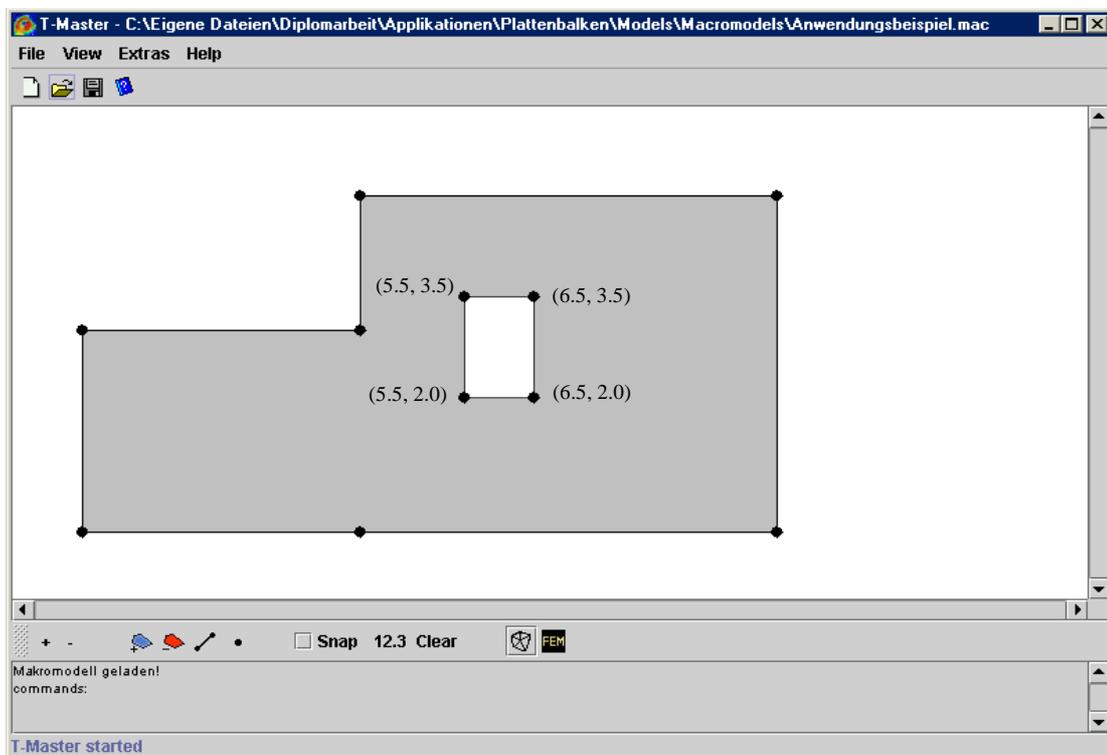


Bild 85: Deckenplatte mit eingeschnittener Öffnung

Für die spätere Definition des Unterzuges wird nun eine Zwangslinie eingezeichnet. Hierzu wird zusätzlich die Snap-Funktion aktiviert.

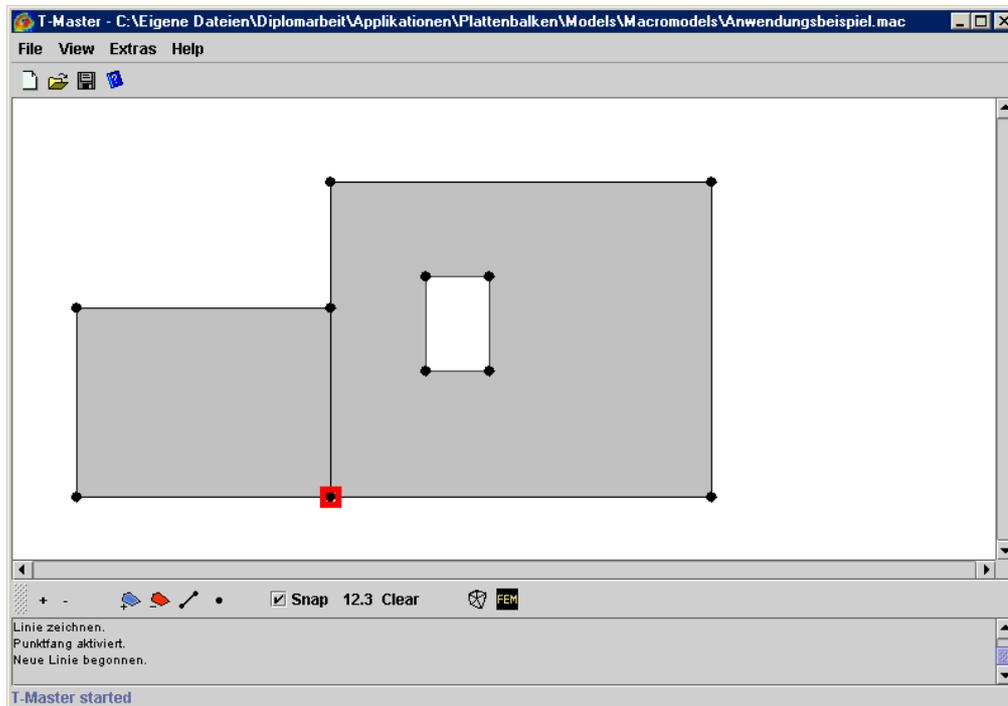


Bild 86: Einziehen der Zwangslinie mit Hilfe des Punktfanges

Auf die Definition eines Zwangspunktes für eine Einzellast wird verzichtet. Um das so definierte Makromodell wieder verwenden zu können wird es mit Hilfe des Save as Menüpunktes als .mac-Datei gespeichert. So können bei Bedarf aus der erzeugten Geometrie mehrere Netze generiert werden. Nun wird der Netzgenerator durch Betätigung des vorletzten Knopfes der Toolbox gestartet. Die folgende Sicherheitsabfrage vor dem Wechseln des Editors kann einfach bestätigt werden. Es folgt daraufhin die Abfrage der zu verwendenden Maschenweite. Hier wurde eine Maschenweite von 0.6 gewählt. Nach Bestätigung erfolgt die Netzgenerierung und der Editorwechsel. Das FEM-Netz wird sogleich angezeigt und das Auswahlwerkzeug aktiviert. Durch Anwählen von Punkten bzw. Elementen und gleichzeitigem Drücken der rechten Maustaste wird ein entsprechendes Kontextmenü geöffnet. Hier können die Lager, Koordinatensysteme und Lasten eingegeben werden. Es wird eine Flächenlast von -1 auf allen Elementen und eine allseitige einfache Auflagerung in jedem Knoten gewählt. Eine Ausnahme bildet die linke untere Wand, hier wird keine Lagerung vorgesehen (freier Rand). Die gelagerten Knoten werden rot markiert. Die Lagerungen können aufgrund der achsenparallelen Geometrie sowohl im globalen System oder in einem eigenen lokalen System gewählt werden. Es ist auf die richtige Wahl der Lagerung zu achten. Eine differenziertere Darstellung erfolgt erst im FEM-Programm. Hier wurde auf der linken oberen Wand ein lokales Koordinatensystem mit einem Winkel von 180° gewählt. Alle anderen Lager werden im globalen System gewählt. Am oberen Balkenende wird eine volle Scheibenlagerung gewählt, am unteren wird die Verschiebung in 1-Richtung gesperrt. Auf den Ecken werden keine Verdrehungen gesperrt.

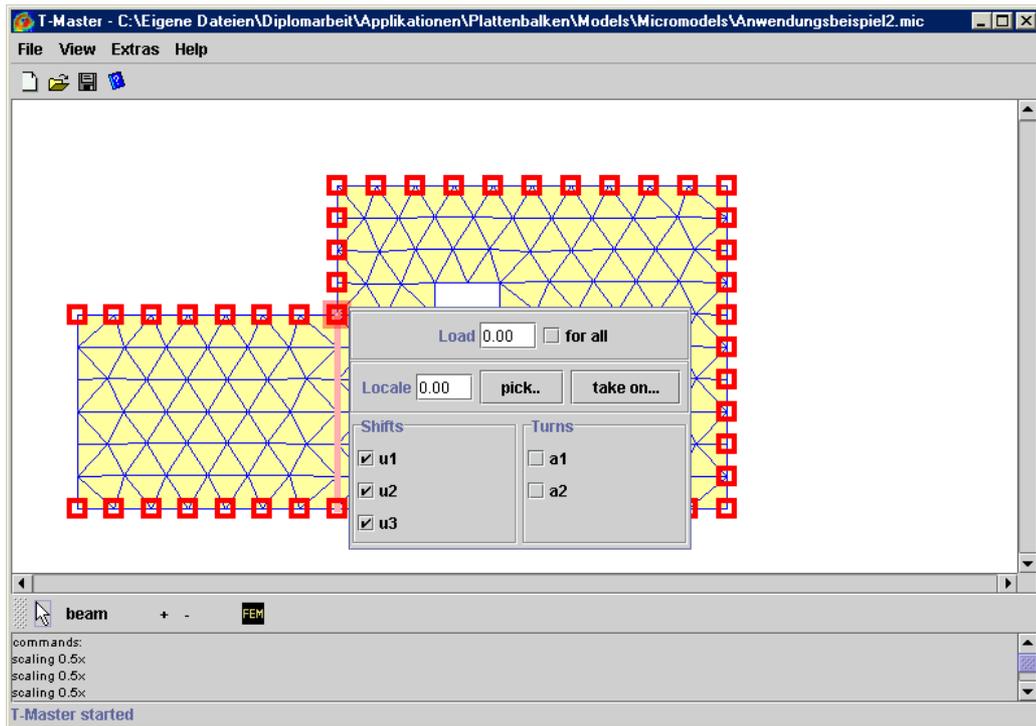


Bild 87: Eingabe der Lager, Lasten, Unterzue und Koordinatensysteme im Mikromodelleditor

Anschließend wird mit dem entsprechendem Werkzeug der Balken definiert. Vor dem Wechsel in das FEM-Programm wird wiederum eine Speicherung, diesmal als .mic-Datei vorgenommen um später eventuell leichte Änderungen an den Lagern oder Lasten vornehmen zu können. Nach dem Wechsel in das FEM-Programm wird das gerade definierte System sofort eingelesen, so daß sofort die verschiedenen Visualisierungsmöglichkeiten genutzt werden können.

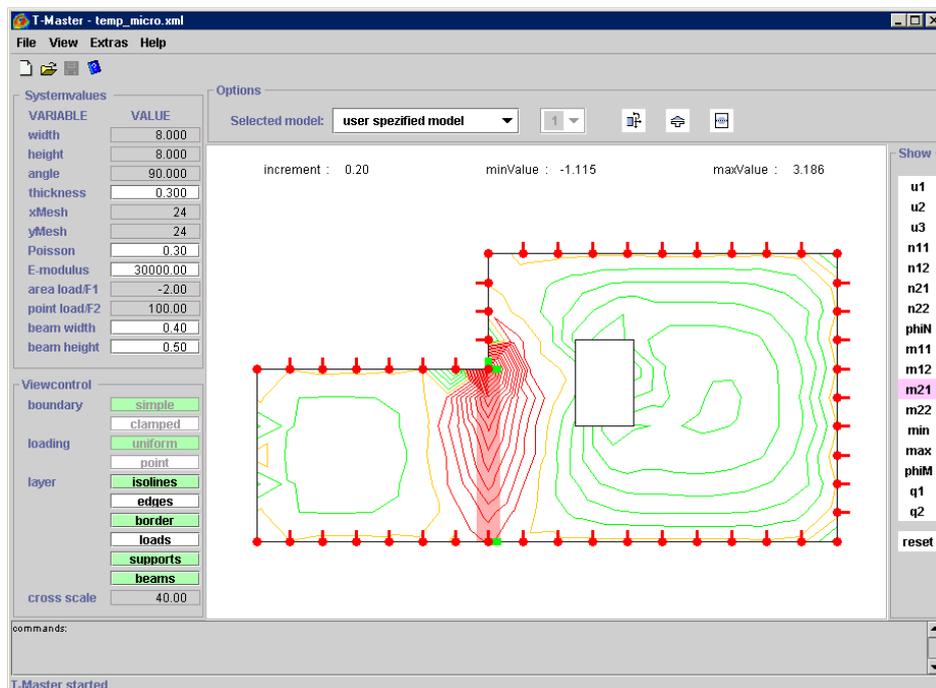


Bild 88: Isolinien für m21 bei einer Plattendicke von 0,3 und einem Unterzug von 0,4x0,5

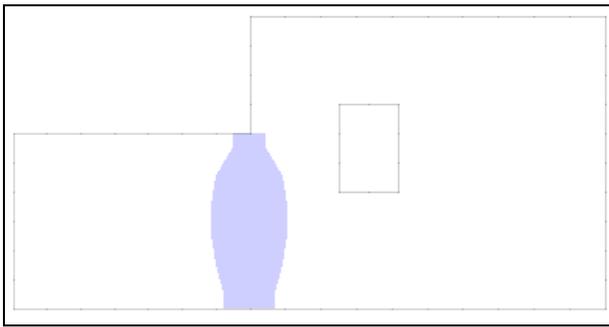


Bild 89: Mitwirkender Bereich des Systems

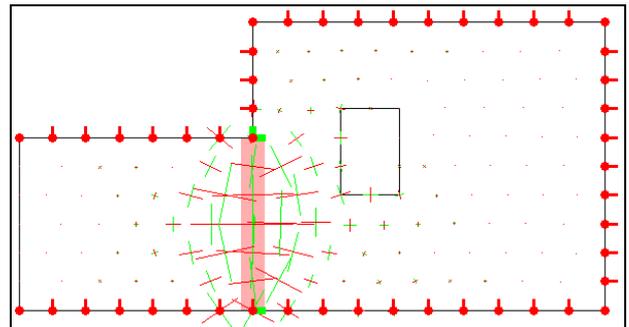


Bild 90: Hauptspannungskreuze

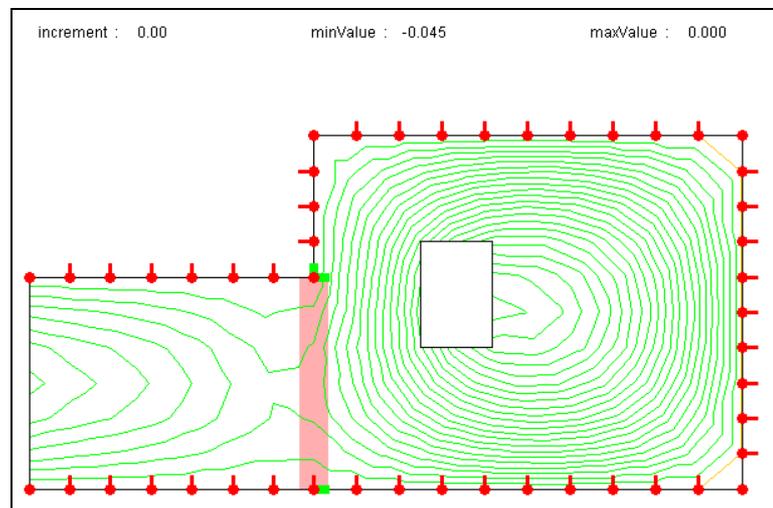


Bild 91: Durchbiegung des betrachteten Systems.

Es ergeben sich sehr anschauliche Ergebnisse, die alle in dieser Form erwartet werden konnten. Als maximale mitwirkende Breite ergab sich ein Wert von 1,270m bei einer vorhandenen Systemlänge von 3m.

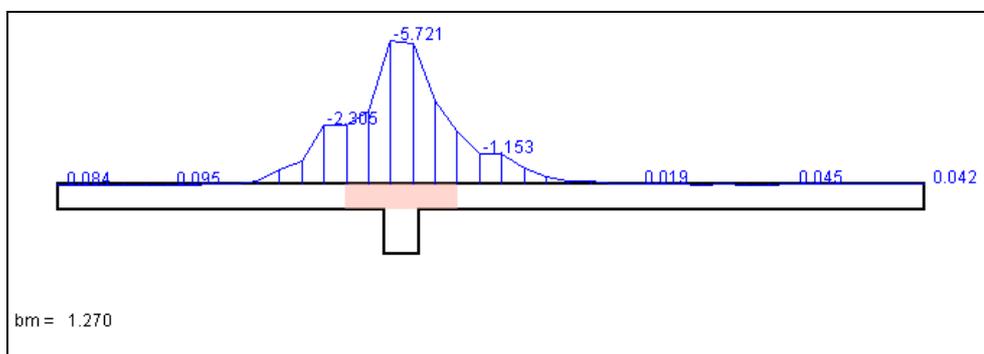


Bild 92: Spannungsverteilung im Schnitt

1. Allgemeines

Die Implementierung des gesamten rechnergestützten Werkzeuges geschah in Java. Dies ist bewußt gewählt, um die Vorteile der Plattformunabhängigkeit in Verbindung mit einer sehr leistungsfähigen, objektorientierten Programmiersprache voll nutzen zu können.

Nun bietet jedoch die Programmiersprache Java neben der reinen Plattformunabhängigkeit bezüglich der verwendeten Rechenmaschine und deren Betriebssystem noch einen weiteren sehr großen Vorteil: Ihre enge Verbindung mit dem Internet. So wird Java schon seit einiger Zeit dafür verwendet die Attraktivität von Internetseiten durch kleine Animationen und Programme zu erhöhen. Von vielen dynamischen oder auch interaktiven Homepages ist Java schon lange nicht mehr weg zu denken.

Jedoch trifft man den Einsatz von vollständigen Programmpaketen im Internet noch immer recht selten an. Dies hat die unterschiedlichsten Gründe und liegt nicht zuletzt auch immer noch an den Übertragungsraten.

Nichts desto trotz soll hier dieser Weg beschritten werden. Ziel ist es die Lauffähigkeit des entwickelten Programmpaketes möglichst unter Aufrechterhaltung der vollen Funktionalität, auch im Internet zu gewährleisten um die Resultate dieser Arbeit einem möglichst großen Interessentenkreis zur Verfügung stellen zu können.

2. Überführung in ein Applet

Der Unterschied zwischen der stand-alone-Anwendung und einem web-basierten Programm liegt darin, daß der Quellcode zunächst nicht lokal auf der Festplatte vorliegt, sondern aus dem Netz in den Speicher geladen werden muß. Der Anlaß für diesen Vorgang ist ein spezieller Applettag im HTML-Code der das Programm zur Verfügung stellenden Homepage (für Näheres siehe Abschnitt 3.2. dieses Kapitels). Um ein Programm auf diese Weise ausführen zu können ist es notwendig, daß das Ausgangsobjekt das durch den erwähnten Tag angesprochen und ausgeführt werden soll von der Klasse Applet abgeleitet wird. Dabei stellt das Applet eine Art in die Seite integriertes Fenster dar. Die unmittelbare Initialisierung erfolgt bei Aufruf der Seite durch den Browser mittels einer `init()`-Methode.

Der sonst übliche Start des Programms durch den Aufruf der Java VM mit Hilfe des `java`-Kommandozeilentools entfällt hierbei, das Applet läuft in der Browser eigenen Java VM ab, das, wie später noch gezeigt, weiterführende Konsequenzen hat.

Um den Start des FEM-Programms dem Besucher der Seite zu überlassen wurde ein sehr einfaches Applet gewählt, das lediglich einen Startknopf zur Verfügung stellt. Bei Betätigung dieses Knopfes wird lediglich ein Objekt der graphischen Oberfläche erzeugt. Jeder weitergehende Quellcode bleibt somit unverändert. Der wesentliche Teil ist nachfolgend kurz gezeigt:

```
public class TApplet extends Applet implements ActionListener
{
    Button sim = new Button("START");

    public void init()
    {
        this.add(sim);
        sim.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e)
    {
        Gui.ScheibenAss frame = new Gui.ScheibenAss();
        frame.setVisible(true);
    }
}
```

3. Einbettung in die HTML-Seite

Wie schon erwähnt wird das nun umgeschriebene Programm mit Hilfe eines Applet Tags in den HTML-Code eingebunden. Um eine möglichst kurze Übertragungszeit zu erreichen wird das gesamte Softwarepaket in ein jar-Archive gepackt. Dies hat zwei Vorteile: Zum einen werden die Dateien analog zum zip-Archive gepackt, zum anderen handelt es sich bei dem jar-Archive um nur eine Datei, was folglich nur einen Verbindungsaufbau zur Übertragung erforderlich macht. Dies ist bei einer großen Anzahl von Dateien ein wesentlicher Punkt da mitunter die Zeit für den Aufbau der einzelnen HTTP-Verbindungen die eigentliche Ladezeit überschreiten kann.

Wird die das Applet enthaltene HTML-Seite vom Browser geöffnet so erkennt dieser das eingebettete Applet und beginnt damit das komplette jar-Archive zu laden. Dies liegt dann lokal im Speicher vor. Somit wird ein direkter Zugriff auf alle verwendeten Dateien möglich.

3.1. Erzeugen eines jar-Archives

Im JDK ist ein einfaches Kommandozeilentool zur Erzeugung von jar-Archiven enthalten. Es wird beispielsweise über folgende Befehlszeile aufgerufen:

```
jar cfv TMaster.jar *.class graphik\*.class graphik\*.gif
```

Die Parameter c, f, v bedeuten in ihrer Reihenfolge, daß ein neues Archiv erstellt wird, daß der Name des Archives angegeben wird und das eine ausführliche Ausgabe der zugefügten Dateien am Bildschirm ausgegeben wird.

Es wird somit ein Archiv namens TMaster.jar erzeugt, dem alle .class Dateien im aktuellen Verzeichnis und im Unterverzeichnis graphik in das Archiv eingebettet. Die vorliegende Verzeichnisstruktur, die der Paketstruktur der Applikation entspricht bleibt dabei auch im Archiv erhalten, so daß keine Änderungen am Quellcode nötig werden.

Es ist darauf zu achten, daß alle Dateien, die benötigt werden in das jar-Archive eingefügt werden, dies gilt auch für Java-Erweiterungen, die vom vorausgesetzten

Browser nicht unterstützt werden. In diesem speziellen Fall betrifft das Klassen die Funktionalitäten zum Verarbeiten von XML-Dateien zur Verfügung stellen.

Auch das durchgehend im Programmpaket verwendete Java-Swing wird von den Browsern zur Zeit noch nicht unterstützt, da diese meist ältere Java-Versionen verwenden. Um jedoch die Dateigröße und damit die Ladezeit nicht unnötig zu erhöhen wird die Bereitstellung von Swing über ein plug-in, welches noch weitere wesentliche Vorteile bietet gewährleistet. (Näheres dazu siehe Abschnitt 4.2.)

Das so gepackte jar-Archive des entwickelten Programms hat eine Größe von 450kb

3.1.1. Probleme mit Bildern

Wie bereits erwähnt sind prinzipiell keine Quellcodeänderungen notwendig um das Applet direkt aus dem jar-Archive ausführen zu können. Es gibt jedoch eine Ausnahme, die das Laden von Bildern und ähnlichen Objekten betrifft.

Wird versucht ein Bild über den Standardweg zu laden, so wird das Bild nicht gefunden.

Der folgende Code führt zu einem solchen Fehler:

```
ImageIcon i = new ImageIcon("graphic/pic.jpg");  
image = i.getImage();
```

Um ein Bild dennoch laden zu können muß ein Umweg über ein URL-Objekt beschritten werden. Es wird zunächst die url des betreffenden Bildes relativ zu der das Bild ladenden Klasse ermittelt und anschließend das Bild mit Hilfe der getImage(..)-Methode des Toolkits geladen:

```
URL url = graphik.Isolines.class.getResource("pic.jpg");  
image = Toolkit.getDefaultToolkit().getImage(url);
```

3.2. Das <APPLET>-Tag

Um dem Browser mitzuteilen, daß ein Applet auszuführen ist muß dieses mit Hilfe des <Applet>-Tags in die HTML-Seite eingebunden werden. Standardmäßig ist der Name und Pfad des auszuführenden Applets sowie Breite, Höhe und Ausrichtung des Appletsbereichs innerhalb der Seite anzugeben.

Um die Ausführung direkt aus einem jar-Archive zu ermöglichen muß der relative Pfad des Archives mit Hilfe des archive-Parametes angegeben werden:

```
<APPLET
  CODE = "TApplet.class"
  archive = "../..../java/program/TMaster.jar"
  WIDTH = "220" HEIGHT = "115" ALIGN = "baseline">
  Applet konnte nicht geladen werden!
</APPLET>
```

Der Alternativtext wird angezeigt, wenn ein Starten des Applets nicht möglich ist.

4. Sicherheitsbeschränkungen

Das bisher erzeugte Applet ist bereits im Netz lauffähigen und kann mit einem Java-fähigem Browser gestartet werden. (Bei Einfügen aller benötigter Klassen in das Archive - inklusive des Swing-Paketes).

Jedoch führen einige Aktionen wie z.B. das Öffnen und Speichern von Dateien zu Fehlern.

Dies ist darin begründet, das ein nicht vertrauenswürdige Applet –dazu gehören voreinstellungsmäßig alle- bestimmten Sicherheitsbeschränkungen unterliegt.

4.1. Java Sandbox

Die Java Sandbox ist eine Sicherheitseinrichtung, die die Sicherheitsgrenzen für nicht vertrauenswürdigen Code festlegt.

Wird eine Applikation lokal auf einem Rechner durch einen direkten java-Dateiaufruf gestartet, so wird davon ausgegangen, das der Nutzer sich über die Vertrauenswürdigkeit der aufgerufenen Anwendung selbst vergewissert hat. Das Programm wird nicht in einer Sandbox gestartet, das heißt es erhält alle Rechte.

Wird dagegen ein Applet vom Browser gestartet, so ist nicht von vornherein klar ob der geladenen Code vertrauenswürdig ist. Er wird deshalb in der Sandbox ausgeführt, was dazu führt das nicht erlaubte Aktionen Fehlermeldungen liefern.

Grundsätzlich sind alle Zugriffe auf das lokale Dateisystem, die Veränderung von Systemwerten, das Beenden der Java VM, sowie der eigenständige Verbindungsaufbau zu anderen Servern als zu dem von dem das Applet geladen wurde untersagt.

Da aber das Laden und Speichern von Dateien eine zentrale Funktionalität des entwickelten Programmpaketes ist, ist es notwendig die Sandbox zu verlassen.

Ziel ist es dem Anwender im Internet eine Nutzung zu ermöglichen, die sich in keinem Punkt von der einer lokal installierten Applikation unterscheidet.

Prinzipiell ergeben sich dabei zwei Möglichkeiten, die geforderten Berechtigungen zu erhalten:

1. Der Anwender teilt seinem System mit Hilfe von Policies von vornherein mit, daß das betreffende Applet vertrauenswürdig ist.
2. Das Applet wird zertifiziert bzw. signiert.

4.2. Java plug-in

In der vorliegenden Implementierung wird ein Java plug-in (Version 1.3) verwendet. Dies hat mehrere Gründe. Einerseits ist es auf diesem Wege möglich das Vorhandensein einer bestimmten Java-Version voraussetzen zu können, was insbesondere das bereits angesprochene Swingpaket betrifft, andererseits verwenden die beiden wichtigsten Browser IE und Netscape verschiedene eigene Sicherheitsmechanismen. Um die Entwicklung mit beiden Browsern nutzen zu können ist es somit einfacher eine Java Version z.B. 1.3 von SUN, sowie ein Sicherheitskonzept vorzusetzen und dies dem Browser mitzuteilen.

Dies spart einerseits Entwicklungsarbeit, andererseits wird auch die Ladezeit des eigentlichen Applets reduziert, da weniger Dateien in das Archive eingefügt werden müssen.

Wird die das Applet enthaltene Seite geladen, so soll der Browser veranlaßt werden das entsprechende Java plug-in zu verwenden. Ist dieses nicht auf dem lokalen Rechner verfügbar, so wird der Nutzer durch ein Fenster dazu aufgefordert das entsprechende Paket unter der angegebenen Adresse herunterzuladen.

Um dem Browser nun explizit mitzuteilen welches plug-in er verwenden soll muß die in Abschnitt 3.2 beschriebene Einbindung des Applets in die HTML-Seite geändert werden.

SUN bietet dafür den HTML-Converter, ein Konvertierungs-Tool das eine normale HTML-Seite mit Applet-Einbindung in eine Seite mit Plug-In-Aufruf umwandelt, an.

Dieses Tool hat eine grafische Oberfläche, die intuitiv bedient werden kann.

Der mit diesem Tool konvertierte Teil der Seite sieht folgendermaßen aus:

```
<!--"CONVERTED_APPLET"-->
<!-- CONVERTER VERSION 1.3 -->
<OBJECT classid="clsid:8AD9C840-044E-11D1-B3E9-00805F499D93"
        WIDTH = "250" HEIGHT = "115" ALIGN = "baseline"
        codebase="http://java.sun.com/products/plugin/1.3/jinstall-13-
                win32.cab#Version=1,3,0,0">
    <PARAM NAME = CODE VALUE = "TApplet.class" >
    <PARAM NAME = ARCHIVE VALUE = "../..../java/program/TMaster.jar" >
    <PARAM NAME="type" VALUE="application/x-java-applet;version=1.3">
    <COMMENT>
    <EMBED type="application/x-java-applet;version=1.3"
        java_CODE = TApplet.class
        java_ARCHIVE = "../..../java/program/TMaster.jar"
        WIDTH = "250" HEIGHT = "115" ALIGN = "baseline"
        pluginspage="http://java.sun.com/products/plugin/1.3/plugin-
                install.html">
    <NOEMBED>
    </COMMENT>
        Laden von ActiveX-Steuerelementen muß ermöglicht sein!
    </NOEMBED>
    </EMBED>
</OBJECT>
```

Um die Funktionsweise sicher zu stellen muß im Browser die Übertragung von ActiveX-Steuerelementen aktiviert sein.

4.3. policy-files

Policies sind Textdateien, die für bestimmte Applets erweiterte Rechte vorsehen. standardmäßig erwartet Java alle Policies in der Datei `.java.policy`, die sich im Home-Verzeichnis des Users befindet. Auch das verwendete plug-in greift auf diese Datei zu. Browser eigene Laufzeitumgebungen jedoch nicht. Zur Erstellung solcher Policies steht das JDK-Tool `policytool` zur Verfügung. Die Eingabe `c:\>policytool` öffnet eine grafische Oberfläche zur Bearbeitung solcher Dateien.

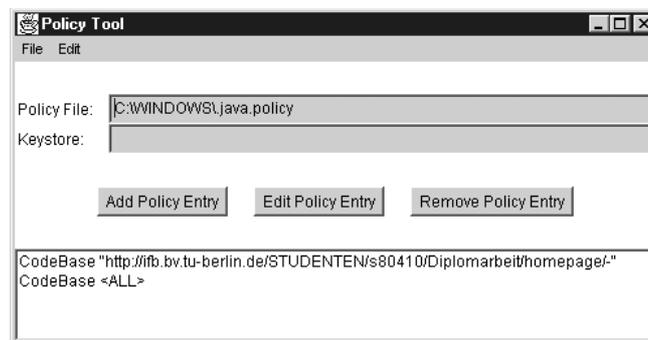


Bild 93: Startfenster des policy-Tools

Standardmäßig wird versucht die Datei `.java.policy` zu erweitern, ist diese noch nicht vorhanden, so wird eine Fehlermeldung angezeigt, die ignoriert werden kann. Um eine problemlose Funktion des Applets zu gewährleisten müssen die in der folgenden Abbildung zu sehenden Rechte erteilt werden:

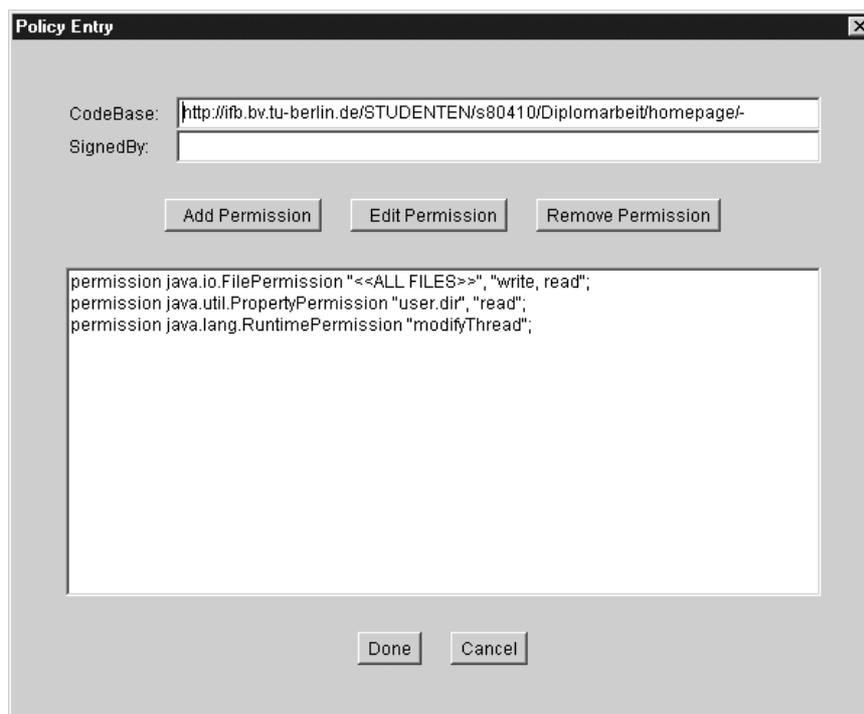


Bild 94: vom policy-Tool eingetragene permissions

Um nun nicht allen Applets die betreffenden Rechte zu geben wird unter CodeBase die Herkunftsadresse des betreffenden Applets angegeben. Der Bindestrich bedeutet, daß alle HTML-Seiten in dem betreffenden Verzeichnis diese Rechte bekommen.

Alle vergebenen Rechte werden vom entwickelten Programm ausschließlich beim lesen und schreiben von Dateien benötigt.

Es sei noch auf einen Sicherheitsaspekt hingewiesen:

Das beschriebene Verfahren sichert dem Applet aufgrund seiner Herkunft erweiterte Rechte zu. Es wird nicht explizit überprüft welcher Code geladen wird, so daß ein von dritten auf dem Server negativ verändertes oder gar vollständig ersetztes Programm auch diese Rechte und somit unerwünschten Zugang erhalten würde.

4.4. signierte jar-Archive

Um sicherstellen zu können, daß ein Applet auch von dem Programmierer kommt, von dem es gemacht ist, ist es möglich, Applets mit einer digitalen Unterschrift auszustatten.

Das Prinzip beruht auf dem Vergleich zweier Schlüssel: dem öffentlichen und dem privaten Schlüssel. Der private Schlüssel wird dazu benutzt Daten zu signieren, der öffentliche Schlüssel hingegen um überprüfen zu können ob das Archive vom erwarteten Autor verfaßt wurde.

Das Erzeugen der beiden Schlüssel erfolgt mit dem Kommandozeilentool keytool, wobei das Schlüsselpaar in einem keystore hinterlegt wird. Danach ist es einerseits möglich den öffentlichen Schlüssel in Form eines Zertifikates zu exportieren, andererseits läßt sich mit dem Tool jarsigner ein bestehendes jar-Archive mit dem privaten Schlüssel signieren.

Da dieses selbst Signieren von Archiven nur wenig Sinn macht gibt es kommerzielle Authentifizierungsstellen (z.B. VeriSign), bei denen sich ein Softwareentwickler sein Zertifikat bestätigen lassen kann. Zertifikate, die auf diese Weise als vertrauenswürdig eingestuft wurden werden vom Browser erkannt. Der Anwender wird daraufhin mit einem Dialog konfrontiert, um zu entscheiden ob das Applet ausgeführt werden soll oder nicht.

Ein Beispiel für ein solches Dialogfenster ist nachfolgend zu sehen:



Bild 95: Dialog bei Authentifizierung

Wird die Anfrage bestätigt, so wird dem damit zertifizierten Applet die geforderten Rechte erteilt. Äußerlich wird dies dadurch sichtbar, daß alle vom Applet geöffneten Fenster nicht mit dem sonst üblichen Warnhinweis „Java Applet Window“ gekennzeichnet sind.

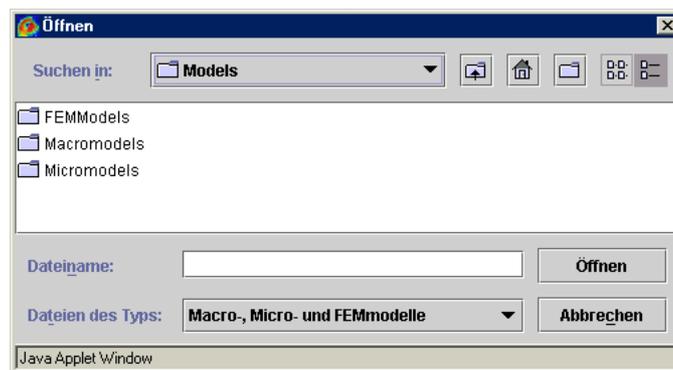


Bild 96: Öffnen-Dialog eines nicht signierten Applets

Bei selbstsignierten Applets funktioniert diese Vorgehensweise bei einigen Browsern ebenfalls, mit dem Hinweis, daß das die Authentizität des Zertifikats nicht bestätigt werden kann.



Bild 97: Dialog bei einem selbst zertifiziertem Applet

Da der User die Brisanz dieses kleinen Unterschiedes oftmals nicht erkennt, ist die selbst Zertifizierung in neueren Browsern nicht mehr auf diese Art möglich. So auch beim verwendeten Java plug-in.

Um zu Testzwecken dennoch eine Selbstzertifizierung zuzulassen muß das eigene Zertifikat in den Browser geladen und als vertrauenswürdiger Stamm gekennzeichnet werden. Bei der Verwendung des plug-in muß analog vorgegangen werden.

Ist dies nicht der Fall, so wird das Applet wiederum in der Sandbox ausgeführt. Ein Dialog erscheint nicht.

4.5. gewählte Variante

Da, wenn von einer kostenpflichtigen Authentifizierung abgesehen wird, sich beide aufgeführten Methoden zur Aufweichung der Sandbox für den Endnutzer nur unwesentlich unterscheiden ist die Wahl recht beliebig (beide Verfahren erfordern, daß vor dem starten des Applets eine Datei heruntergeladen und richtig hinterlegt wird).

Aufgrund der Einfachheit wurde hier trotz des gewissen -aus den unter 4.3. erläuterten Gründen bestehenden- Sicherheitsrisikos der Weg über das Policy-file gewählt.

Da im allg. keine Policy-files auf dem Rechner des Anwenders vorhanden sind wird die komplette Datei .java.policy zum Herunterladen angeboten. Sie muß vor dem Starten des Applets in das Homeverzeichnis kopiert werden.

Ist bereits eine solche Datei vorhanden, so muß der Inhalt der zum Download bereitgestellten Datei manuell der auf dem lokalen System vorhandenen Datei angefügt werden. Der Inhalt hat folgende Gestalt:

```
grant codeBase "http://ifb.bv.tu-berlin.de/STUDENTEN/s80410/Diplomarbeit/homepage/-" {
    permission java.io.FilePermission "<<ALL FILES>>", "write, read";
    permission java.util.PropertyPermission "user.dir", "read";
    permission java.lang.RuntimePermission "modifyThread";
};
```

Mit dieser Maßnahme ist die Lücke zwischen der stand-alone-Anwendung und dem webbasierten Applet entgültig geschlossen. Es treten auch bei größeren Systemen (zumindest in den untersuchten Fällen) keinerlei Beeinträchtigungen auf.

4.6. web-basierte Dokumentation

Im Hinblick auf das erklärte Ziel diese Arbeit einem möglichst breitem Interessentenkreis zur Verfügung stellen zu können wurde eine Homepage erstellt, die die gesamte Dokumentation sowie auch das erstellte Applet enthält. Ferner steht das komplette Programmpaket auch als jar-Archive zum Download bereit. Die Seite ist über einen Link auf der Seite www.jantulke.de zu erreichen.

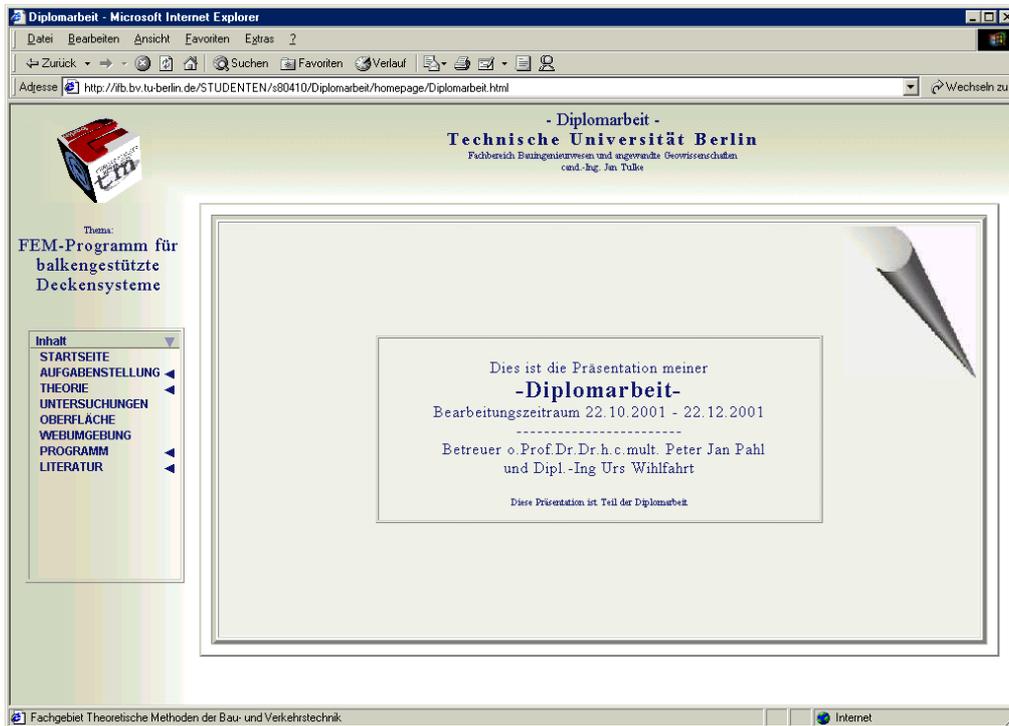


Bild 98: Internetpräsenz dieser Arbeit

Sie ist aus drei Frames zusammengesetzt. Die Navigation wird durch ein Java-Menü im linken Frame realisiert. Ein Würfel in der oberen linken Ecke stellt die Verlinkung zu den wichtigsten, zur Entstehung dieser Arbeit im Bezug stehenden Seiten her: TU Berlin, Fachbereich Theoretische Methoden der Bau- und Verkehrstechnik, eigene Homepage sowie eine E-Mailadresse.

Anlaß und Inhalt der Internetpräsenz gehen unmittelbar aus der Startseite hervor. Der größte Teil des Inhalts besteht aus den einzelnen nach Kapiteln aufgeteilten Abschnitten dieser Dokumentation, die im pdf-Format bereitgestellt werden.

Kapitel I	Literatur
-----------	-----------

- [1] O.C. Zienkiewicz; Methode der finiten Elemente; Carl Hanser
- [2] K.J. Bathe; Finite-Element-Methode; Springer
- [3] K. Knothe, H.Wessels; Finite Elemente; Springer
- [4] H.R.Schwarz; Methode finiten Elemente; Teubner
- [5] F. Hartmann, C. Katz; Statik mit finiten Elementen; Springer
- [6] P.J.Pahl; Plates; Skript zur Lehrveranstaltung Theoretische Methoden der Bau- und Verkehrstechnik Vertiefung II an der TU Berlin
- [7] S.Savidis; Grundbau und Bodenmechanik 1; Skript zur gleichnamigen Lehrveranstaltung an der TU Berlin
- [8] P.J.Pahl; Finite Element Methode für Physikalische Aufgaben; Skript zur Lehrveranstaltung Theoretische Methoden der Bau- und Verkehrstechnik IV an der TU Berlin
- [9] M.Specht, P.Lorenz; Konstruktiver Ingenieurbau 1; Skript zur gleichnamigen Lehrveranstaltung an der TU Berlin
- [10] R.Harbord; Statik der Baukonstruktionen III; Skript zur gleichnamigen Lehrveranstaltung an der TU Berlin
- [11] R.Harbord; Statik der Baukonstruktionen Vertiefung I; Skript zur gleichnamigen Lehrveranstaltung an der TU Berlin
- [12] Heft 220; DAfStb
- [13] DIN 1075
- [14] Brendel; Die „mitwirkende Plattenbreite“ nach Theorie und Versuch; Beton- und Stahlbetonbau, Heft 8, 1960
- [15] E.Grasser, W.Moosecker; Hilfsmittel zu näherungsweise Bestimmung der mitwirkenden Breite von Plattenbalken im Stahlbetonbau; Beton- und Stahlbetonbau, Heft 6, 1982
- [16] W.Wunderlich, G.Kiener, W.Ostermann; Modellierung und Berechnung von Deckenplatten mit Unterzügen; Bauingenieur 69, Springer 1994
- [17] E.Niedermaier, M.Niedermaier; Internet-Programmierung mit JAVA; DATA BECKER

[18] D.M.Geary; graphic JAVA 2.0; SUN, Markt+Technik