TECHNISCHE MECHANIK 11 (1990) Heft 4
Manuskripteingang: 25. 3. 1990

Ein Beitrag zur expliziten Ermittlung symbolischer Matrizen in der Technischen Mechanik

Gunnar Zessin

1. Symbolmanipulation

Die Verwendung von Computern zur Ausführung von aufwendigen symbolischen Rechenoperationen ist seit den 50er Jahren in verschiedensten Wissenschaftsgebieten Praxis geworden.
Für diese Art Rechnernutzung haben sich verschiedene Bezeichnungen eingebürgert, wie symbolische und algebraische Manipulation, Computer-Algebra oder auch Formelmanipulation, wobei diese Begriffe zuweilen unterschiedlichen Aufgabenbereichen zugeordnet werden.
Einige der bisher entwickelten Manipulationssprachen bzw. -systeme haben inzwischen einen relativ großen Verbreitungskreis gefunden.

Die Entwicklungsgeschichte solcher führenden Sprachen und Systeme wird z. B. in [1], [2] und [4] beschrieben. Eine Übersicht gebräuchlicher Systeme liefert [5]. Klassifizierungen der Systeme sind unter anderem möglich nach Verwendungszweck, Leistungsumfang und Nutzerfreundlichkeit. In [2] wird eine Einteilung, bezogen auf die Sprachebene, in 3 Gruppen vorgenommen:

- Nutzung eines Prozedurpaketes, geschrieben in einer – häufig numerisch orientierten – Programmiersprache.
- Nutzung einer Wirtssprache, wobei alle Fähigkeiten der Wirtssprache zur Programmierungstechnik vollständig zur Verfügung stehen.
- Problemorientierte (eigenständige) Programmierungssprachen.

Anwendungsmöglichkeiten findet man sowohl in der Chemie, der Biologie und der Physik als auch in der Mathematik und in den Computerwissenschaften ([6] und [7]).

2. Symbolmanipulation in der Festkörpermechanik

In [8] wurde noch 1979 konstatiert, daß Computer-Algebra-Systeme in der Festkörpermechanik im Vergleich zu anderen physikalischen Disziplinen nur zögernd Eingang finden. Neben üblichen Berührungsängsten ist dies sicher vor allem dem Umstand geschuldet, daß es für erfahrene Programmierer durchaus möglich ist, in verschiedenen Bereichen Formelmanipulationen durch effektive numerische Berechnungen zu umgehen.
Jedoch sind besonders seit Ende der 60er Jahre auch Bestrebungen dokumentiert, die Formulierung von charakteristischen Matrizen im Rahmen von Matrizenmethoden wie der Finite-Elemente-Methode (FEM) durch symbolische Computerrechnungen zu unterstützen. Neben der Orientierung auf Matrizenmanipulationen ist hier im allgemeinen auch die Generierung der Ergebnisse im Code einer bestimmten Programmiersprache (meist FORTRAN) typisch [9].

Einen Abriß von Anwendungsmöglichkeiten innerhalb der FEM zeigt [10].
Genutzt werden sehr verschiedene Systeme. Am häufigsten erwähnt wird die Anwendung von MACSYMA/VAXIMA, das das z. Z. am höchsten entwickelte System darstellt (z. B. in [8], [9], [11] bis [13]). Es entstand auf der Grundlage von LISP und ist nur auf wenigen Computern verfügbar.

Einige weitere Anwendungsbeispiele sollen im folgenden ohne Anspruch auf Vollständigkeit erwähnt werden:

- FORMULA ALGOL ([14]): eine Erweiterung von ALGOL
- FORMAC ([15] bis [17]): nutzt als Wirtssprache FORTRAN oder PL/1
- REDUCE ([9] und [18]): eine eigenständige Programmiersprache, deren Entwicklung an LISP angelehnt wurde. REDUCE ist das wohl am weitesten verbreitete Sprachsystem der Computer-Algebra und kann auf fast allen Hauptmodellen der Groß- oder Mikrorechner verwendt werden.
- CAL ([19]): ein Computer-Analysis-System unter Verwendung von FORTRAN
- ALTRAN ([20]): eine Sprache zur Manipulation von Polynomen mit FORTRAN als Wirtssprache
- ALGOL-Prozeduren ([21])
- FORTRAN-Prozeduren ([22] bis [24])

Dabei reichen die Beispiele von der Ausführung spezieller mathematischer Operationen (wie Integration) bis zu kompletten Generatoren von Steifigkeits- oder Massenmatrizen für bestimmte FEM-Varianten.
Bei letzterem kontrastiert der Grad der Automatisation der Prozesse zwangsläufig mit der Möglichkeit, Matrizen für verschiedenartige Elementtypen ermitteln zu können.

3. Ein modulares Konzept

Da sich wahrscheinlich kein Nutzer permanent mit der Herleitung von FEM-Matrizen beschäftigt, ist es sicher zu akzeptieren, wenn auf die durchgängige Automatisierung verzichtet wird. Dadurch ist es dann auch eher möglich, diese Systeme modular aufzubauen, so daß der Zugang zu unterschiedlichen FEM-erfahren wie auch zu anderen Matrizenmethoden erfolgen kann.

Hauptzweck eines solchen Systems ist die Erledigung monotoner, zeitaufwendiger und fehleranfälliger Routinearbeiten. Unter Umständen wird die Ausführung dieser Arbeiten durch die Symbolmanipulation erst möglich gemacht.

Die Anzahl der notwendigen Module ist begrenzt. Sie könnten folgende Funktionen beinhalten:

- Multiplikation
- Addition/Subtraktion
- Inversion
- Integration
- Differentiation
- Vereinfachungen
- Ausgabe
- Servicefunktionen (Codierung, Vergleiche u. a.)

Auf obige Gliederung greift ein Prozedurpaket zurück, das vom Autor entwickelt wurde. Dabei handelt es sich um ein Computergestütztes Programmsystem zur expliziten Berechnung alpha-numerischer Matrizen in der Mechanik (COPAN).
COPAN läuft zur Zeit auf 16-Bit-PCs. Es wurde in FORTRAN geschrieben und ist auf Dialogbetrieb orientiert.
Verarbeitet werden Matrizen und Vektoren, die aus polynomartigen Ausdrücken bestehen. Da eine Codierung der Matrizen in INTEGER-Felder erfolgt, ist in der Regel eine Beschränkung auf ganzzahlige Exponenten erforderlich.
Günstig auf die Anwenderfreundlichkeit wirkt sich aus, daß ein Nutzer im wesentlichen keine Kenntnisse über die interne Arbeitsweise von COPAN benötigt. Die notwendigen Informationen werden von den Programmen im Dialog über das Bildschirm-Terminal abgefragt.
Das System ist beliebig erweiterbar, solange auf kompatible Codierungen der Matrizen zurückgegriffen wird.
Ein- und Ausgabe erfolgen derzeit im FORTRAN-Code oder als Tabelle. Um Übertragungsfehler zu vermeiden, erstellt das System auch die Quelltextdateien der Unterprogramme.

4. Anwendungsbeispiel

Im Bereich der FEM bietet sich ein derartiges System z. B. für die Ermittlung von hybriden Spannungselementen an, deren Steifigkeits- und Spannungsmatrizen im allgemeinen numerisch ermittelt werden.
Im folgenden wird kurz die Herleitung des linear-elastischen Rechteck-Scheibenelementes mit jeweils 2 Freiheitsgraden in den Knoten gezeigt (siehe Anhang). Das Ergebnis dürfte besonders für die Anwender interessant sein, die bereits mit dem entsprechenden Verschiebungselement arbeiten, da die Matrizen dann unter Umständen in den Programmen nur ausgetauscht werden brauchen.
Der theoretische Hintergrund sowie Konvergenzuntersuchungen sind in der Literatur ([25] bis [28]) ausführlich beschrieben und werden hier nicht berücksichtigt.

Folgende Rechenoperationen wurden ausgeführt:

$$[E] = h \int_{(F)} [P]^T [C]^{-1} [P] \, dF \qquad (1),$$

worin $[C]$ = Elastizitätsmatrix,
$[P]$ = Matrix der Spannungsansätze,
F = Elementfläche,
h = Elementdicke.

$$[H] = [U]\,[A]^{-1} \qquad (2)$$

worin $[U]$ = Matrix der Verschiebungsansätze,
$[A]$ = Koordinatenmatrix.

$[P^s]$ bzw. $[H^s]$ entstehen aus $[P]$ bzw. $[H]$ durch Einsetzen der Koordinaten der betrachteten Elementränder.
$[T^s]$ beinhaltet die zum jeweiligen Rand gehörenden Richtungstransformationen.

$$[G] = \sum_{s=1}^{4} [P^s]^T [T^s] [H^s] \qquad (3)$$

Schließlich erhält man die Elementsteifigkeitsmatrix zu

$$[K] = [G]^T [E]^{-1} [G] \qquad (4)$$

und die Elementspannungsmatrix zu

$$[S] = 1/h \, [P] \, [E]^{-1} [G]. \qquad (5)$$

Die Spannungen im Element ergeben sich dann durch numerische Ausführung von

$$[\sigma] = [S]\,[u], \qquad (6)$$

wobei $[\sigma] = [\sigma_x \, \sigma_y \, \tau_{xy}]^T$
und $[u] = u_1 \, v_1 \, u_2 \, v_2 \, u_3 \, v_3 \, u_4 \, v_4]^T$ bedeuten.

Die Belegung der Ausgangsmatrizen sowie die Unterprogramme für die Steifigkeits- und die Spannungsmatrix sind im Anhang enthalten. Die Spannungsmatrix wurde für den allgemeinen Fall berechnet, so daß durch die Eingabe der entsprechenden Elementkoordinaten die Spannungen in beliebigen Punkten des Elementes ermittelt werden können. Möglich wären auch Spannungsmatrizen für bestimmte Elementpunkte gewesen.
Die Vereinfachung der Formelausdrücke wurde bis zu einem Grad ausgeführt, der ein manuelles Kopieren der Unterprogramme zumutbar werden läßt.

Anhang 1
Geometrie, Vorzeichendefinitionen, Knotennumerierung

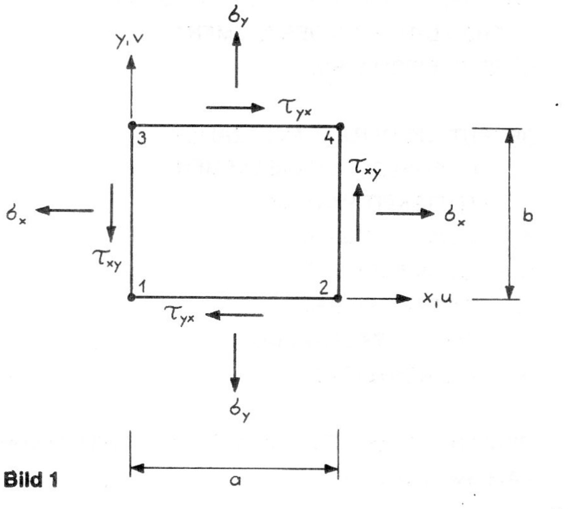

Bild 1

Anhang 2
Ausgangsmatrizen

$$[C]^{-1} = 1/E \begin{bmatrix} 1 & \mu & 0 \\ -\mu & 1 & 0 \\ 0 & 0 & 2+2\mu \end{bmatrix}$$

mit μ = Querdehnungszahl
E = Elastizitätsmodul

$$[P] = \begin{bmatrix} 1 & 0 & 0 & y & 0 & x & 0 \\ 0 & 1 & 0 & 0 & x & 0 & y \\ 0 & 0 & 1 & 0 & 0 & -y & -x \end{bmatrix}$$

$$[U] = \begin{bmatrix} 1 & x & y & xy & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & x & y & xy \end{bmatrix}$$

$$[A] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & a & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & a & 0 & 0 \\ 1 & 0 & b & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & b & 0 \\ 1 & a & b & ab & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & a & b & ab \end{bmatrix}$$

Anhang 3
Elementsteifigkeitsmatrix

```
      SUBROUTINE STEIF (K, A, B, M, E, H)
C
C     UNTERPROGRAMM ZUR BERECHNUNG DES
C     OBEREN DREIECKS DER ELEMENT-
C     STEIFIGKEITSMATRIX
C
C     BEDEUTUNG DER PARAMETERLISTE:
C     K = (8,8)-FELD FUER DIE ELEMENT-
C         STEIFIGKEITSMATRIX
C     A = ELEMENTBREITE
C     B = ELEMENTHOEHE
C     M = QUERDEHNZAHL
C     E = ELASTIZITAETSMODUL
C     H = ELEMENTDICKE
C
      REAL*8K(8,8),AK,A,B,M1,M2,M,C,D,A1,B1,A3,B3,M3,M4
     *, AM,BM,A5,B5,E,H
C
      AK=E*H/144DO
      M1=1DO+M
      M2=1DO-M**2
      C=A**3*B*M2/12DO+A*B**3*M1/6DO
      D=A**3*B*M1/6DO+A*B**3*M2/12DO
      A1=A**2*M-2DO*B**2
      B1=2DO*A**2-B**2*M
      A3=A**3/D
      B3=B**3/C
      M3=36DO*M**2/M2
      M4=36DO/M2
      AM=18DO/A/M1
      BM=18DO/B/M1
      A5=A*M/C*A1
      B5=B*M/D*B1
C
      K(1,1)=AK*(B*(1DO/A*(M3+48DO)-B5)+A*(BM+A3))
      K(1,2)=AK*(M4*M+18DO/M1)
      K(1,3)=AK*(B*(1DO/A*(-M3-48DO)+B5)+A*(BM-A3))
      K(1,4)=AK*(M4*M-18DO/M1)
      K(1,5)=AK*(B*(1DO/A*(M3+24DO)+B5)+A*(-BM-A3))
      K(1,6)=-K(1,4)
      K(1,7)=AK*(B*(1DO/A*(-M3-24DO)-B5)+A*(-BM+A3))
      K(1,8)=-K(1,2)
      K(2,2)=AK*(A*(1DO/B*(M4+12DO)+A5)+B*(AM+B3))
      K(2,3)=-K(1,4)
      K(2,4)=AK*(A*(1DO/B*(M4-12DO)-A5)+B*(-AM-B3))
      K(2,5)=K(1,4)
      K(2,6)=AK*(A*(1DO/B*(-M4-12DO)-A5)+B*(AM-B3))
      K(2,7)=-K(1,2)
      K(2,8)=AK*(A*(1DO/B*(-M4+12DO)+A5)+B*(-AM+B3))
      K(3,3)=K(1,1)
      K(3,4)=-K(1,2)
      K(3,5)=K(1,7)
      K(3,6)=K(1,2)
      K(3,7)=K(1,5)
      K(3,8)=K(1,4)

      K(4,4)=K(2,2)
      K(4,5)=K(1,2)
      K(4,6)=K(2,8)
      K(4,7)=-K(1,4)
      K(4,8)=K(2,6)
      K(5,5)=K(1,1)
      K(5,6)=-K(1,2)
      K(5,7)=K(1,3)
      K(5,8)=-K(1,4)
      K(6,6)=K(2,2)
      K(6,7)=K(1,4)
      K(6,8)=K(2,4)
      K(7,7)=K(1,1)
      K(7,8)=K(1,2)
      K(8,8)=K(2,2)
C
      RETURN
      END
```

Spannungsmatrix

```
      SUBROUTINE SPANN(S, A, B, M, E, H, X, Y)
C
C     UNTERPROGRAMM FUER DIE BERECHNUNG DER
C     SPANNUNGSMATRIX
C
C     BEDEUTUNG DER PARAMETERLISTE:
C     S = (3.8)-FELD FUER DIE SPANNUNGSMATRIX
C     A = ELEMENTBREITE
C     B = ELEMENTHOEHE
C     M = QUERDEHNZAHL
C     E = ELASTIZITAETSMODUL
C     H = ELEMENTDICKE
C     X = ELEMENTKOORDINATE IN X-RICHTUNG
C     Y = ELEMENTKOORDINATE IN Y-RICHTUNG
C
      REAL*8S(3,8),AK,A,B,M1,M2,M,C,D,X,Y,A1,B1,M3,M4,
     *AM,BM,E,H
C
      AK=E/24DO
      M1=1DO+M
      M2=1DO-M**2
      C=A**3*B*M2/12DO+A*B**3*M1/6DO
      D=A**3*B*M1/6DO+A*B**3*M2/12DO
      A1=A-2DO*X
      B1=B-2DO*Y
      M3=A**2*M-B**2
      M4=A**2-B**2*M
      AM=18DO/A/M1
      BM=18DO/B/M1
      A5=A*M/C*A1
      B5=B*M/D*B1
      AM=1DO/C*A1*M3
      BM=1DO/D*B1*M4
C
      S,(1,1)=AK*(1DO/A*(-24DO-12DO/M2*M**2+24DO/
     *  B*Y)+M*BM)
      S(1,2)=AK*(-12DO/B/M2*M-AM)
      S(1,3)=-S(1,1)
      S(1,4)=AK*(-12DO/B/M2*M+AM)
      S(1,5)=AK*(1DO/A*(-12DO/M2*M**2-24DO/B*Y)
     *  -M*BM)
      S(1,6)=-S(1,2)
      S(1,7)=-S(1,5)
      S(1,8)=-S(1,4)
      S(2,1)=AK*(-12DO/A/M2*M+BM)
      S(2,2)=AK*(12DO/B*(-1DO/M2-1DO/A*A1)-M*AM)
      S(2,3)=-S(2,1)
      S(2,4)=AK*(12DO/B*(-1DO/M2+1DO/A*A1)+M*AM)
      S(2,5)=AK*(-12DO/A/M2*M-BM)
      S(2,6)=-S(2,2)
      S(2,7)=-S(2,5)
      S(2,8)=-S(2,4)
      S(3,1)=AK*(-6DO/B/M1-1DO/D*A1*M4)
      S(3,2)=AK*(-6DO/A/M1+1DO/C*B1*M3)
      S(3,3)=AK*(-6DO/B/M1+1DO/D*A1*M4)
      S(3,4)=-S(3,2)
      S(3,5)=-S(3,1)
      S(3,6)=AK*(-6DO/A/M1-1DO/C*B1*M3)
      S(3,7)=-S(3,3)
      S(3,8)=-S(3,6)
C
      RETURN
      END
```

LITERATUR

[1] Sammet, J. E.: Programming languages: history and fundamentals. Prentice-Hall Englewood Cliffs, N. J. 1969.

[2] Mätzel, K., Richter, R., Schulz, M.: Symbolische algebraische Manipulation auf Rechenanlagen. Wiss. Z. d. Techn. Hochschule Karl-Marx-Stadt 16 (1974) H. 3, S. 455 – 467.

[3] Buchberger, B., Collins, G. E., Loos, R. (Hrsg.): Computer algebra. Springer-Verlag Wien – New York 1982.

[4] Loos, R.: Introduction. In: [3], S. 1 – 10.

[5] Van Hulzen, J. A., Calmet, J.: Computer algebra systems. In: [3], S. 221 – 243.

[6] Calmet, J., Van Hulzen, J. A.: Computer algebra applications. In: [3], S. 245 – 258.

[7] Ng, E. W.: Symbolic-numeric interface: a review. In: Ng, E. W. (Hrsg.): Symbolic and algebraic computation, EUROSAM '79. Springer-Verlag Berlin (W.) – Heidelberg – New York 1979 (LNCS 72), S. 330 – 345.

[8] Noor, A. K., Andersen, C. M.: Computerized symbolic manipulation in structural mechanics – progress and potential. Computers & Structures 10 (1979), S. 95 – 118.

[9] Wang, P. S. Chang, T. Y. P., Van Hulzen, J. !: Code generation and optimization for finite element analysis. In: Fitch, J. (Hrsg.): EUROSAM 84. Springer-Verlag Berlin(W.) u. a. 1984 (LNCS 174), S. 237 – 247.

[10] Beljawitschjus, R. W., Kulwetenje, R. W., Kulwetis ,G. P.: Die Anwendung analytischer Berechnungen in der Methode der finiten Elemente (russ.). In: International conference on computer algebra and its applications in theoretical physics. Dubna 1985, S. 301 – 306.

[11] Fenves, S. J., Korncoff, A. R.: Symbolic computing applications. In: Conference on computing in civil engineering. New York 1978. S. 157 – 174.

[12] Noor, A. K., Andersen, C. M.: Computerized symbolic manipulation in nonlinear finite element analysis. Computers & Structures 13(1981), 379 – 403.

[13] Wang, P. S.: Computer-aided finite element analysis: interfacing symbolic and numerical computational techniques. In: Tou, J. T. (Hrsg.): Computer-based automation. Plenum Press New York – London 1985, S. 203 – 212.

[14] Wong, A. K.-C.: Symbolic computing. In: Fenves, S. J. u. a. (Hrsg.): Numerical and computer methods in structural mechanics. Academic Press New York – London 1973, S. 459 – 477.

[15] Wilkins, Jr., D. J.: Applications of a symbolic algebra manipulation language for composite structures analysis. Computers & Structures 3 (1973), S. 801 – 807.

[16] Pedersen, P.: On computer-aided analytic element analysis and the similarities of tetrahedron elements. International Journal for Numerical Methods in Engineering 11 (1977), S. 611 – 622.

[17] Jensen, J.: On the formulation and solution of shell problems by means of computerized algebraic manipulation. Computers & Structures 11 (1980), S. 279 – 287.

[18] Rizzi, N., Tatone, A.: Symbolic manipulation in buckling and postbuckling analysis. Computers & Structures 21 (1985), S. 691 – 700.

[19] Leimbach, K. R.: CAL-EL – an extension of CAL for coding element formulations from any variational principle. Computers & Structures 21 (1985), S. 1361 – 1371.

[20] Pavlovic, M. N., Sapountzakis, E. J.: Computers and Structures: Non-numerical applications. Computers & Structures 24 (1986), S. 455 - 474.

[21] Vos, R. G.: Generalization of plate finite elements to shells. Journal of the Engineering Mechanics Division 98 (1972) H. 2, S. 385 – 400.

[22] Cecchi, M. M., Lami, C.: Automatic generation of stiffness matrices for finite element analysis. International Journal for Numerical Methods in Engineering 11 (1977), S. 396 – 400.

[23] Noor, A. K., Hartley, S. J.: Evaluation of element stiffness matrices on CDC STAR-100 computer. Computers & Structures 9 (1978), S. 151 – 161.

[24] Hoa, S. V., Sankar, S.: A computer program for automatic generation of stiffness and mass matrices in finite-element analysis. Computers & Structures 11 (1980), S. 147 – 161.

[25] Pian, T. H. H.: Derivation of element stiffness matrices by assumed stress distributions. AIAA Journal 2 (1964), S. 1333 – 1336.

[26] Todorow, M.: Anwendung finiter hybrider Spannungselemente in der Elastostatik. Diss., TH Magdeburg 1978.

[27] Altenbach, J., Sacharov, A. S. (Hrsg.): Die Methode der finiten Elemente in der Festkörpermechanik. VEB Fachbuchverlag Leipzig 1982.

[28] Bathe, K.-J.: Finite-Elemente-Methoden. Springer-Verlag Berlin (W.) u. a. 1986.

Anschrift des Verfassers:
Dr.-Ing. G. Zessin
Bauakademie der DDR
Institut für Projektierung und Standardisierung
Abt. Rechenstation
Plauener Str. 163 – 165
Berlin
1092