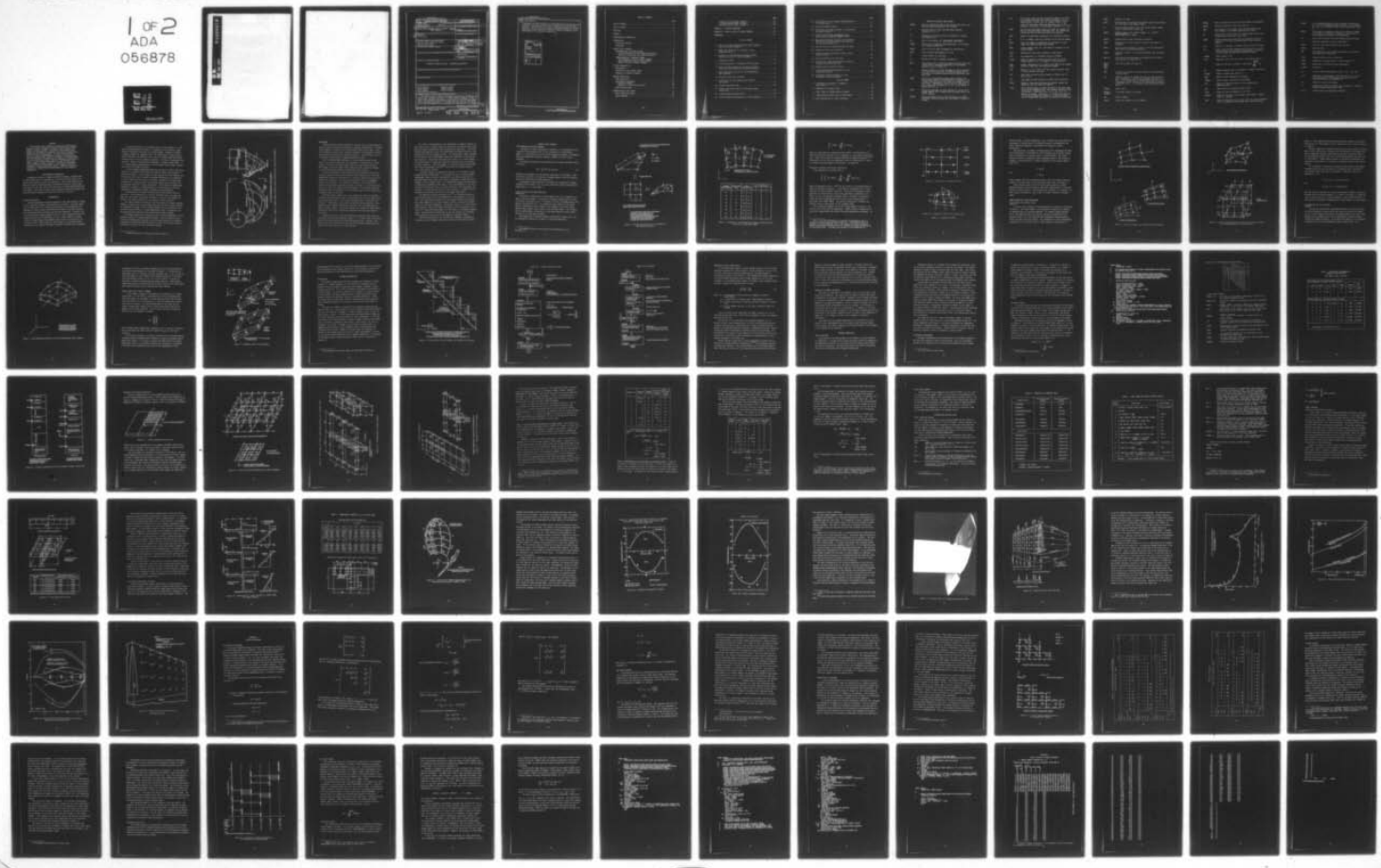
AD-A056 878 DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 9/2
CURVED FINITE ELEMENTS COMPUTER PROGRAM - PBLADE USER'S MANUAL.(U)
APR 78 J H MA

UNCLASSIFIED DTNSRDC-78/016 NL

1 OF 2
ADA
056878



AD No.
DDC FILE COPY

AD A056878

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: DTNSRDC-78/016

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle): CURVED FINITE ELEMENTS COMPUTER PROGRAM - PBLADE USER'S MANUAL

5. TYPE OF REPORT & PERIOD COVERED: Research and development report ~~Formal~~

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s): James H. Ma

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS: David W. Taylor Naval Ship Research and Development Center, Bethesda, Md. 20084

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Task Area SF 43 422 593, SF43422505, Task 17934, F43 422, 16, Work Unit 1-1730-312

11. CONTROLLING OFFICE NAME AND ADDRESS

12. REPORT DATE: April 78

13. NUMBER OF PAGES: ~~105~~ 102 P.

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report): UNCLASSIFIED

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Finite Elements
Plate Elements
Solid Elements
Supercavitating Propeller
Computer Program
Solution Method
Frontal Solution

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The present report provides the detailed instructions to perform a structural analysis using the curved finite element computer program - PBLADE. It defines input variables and format. It discusses the idealization of structures, their geometrical and material properties, the scope of computations and the core size requirements. It further discusses the effect of computational algorithm employed in the development and organization of the program.

(Continued on reverse side) → next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

387 682

78 06 29 025 hc

(Block 20 continued)

An example of the data formation in a well-known frontal solution procedure is described in detail to allow further exploitation of the efficient algorithm. Sample problems are given to illustrate applications and capabilities of the program to solve complex structural problems of a three-dimensional nature. Numerical results are presented to demonstrate the effectiveness of the programed computation.

ACCESSION for
NTIS White Section X
DDC Buff Section ☐
UNANNOUNCED ☐
JUSTIFICATION
BY
DISTRIBUTION/AVAILABILITY CODES
Dist. AVAIL. and/or SPECIAL
A

TABLE OF CONTENTS

	Page
LIST OF FIGURES	iv
LIST OF TABLES	v
NOTATION	vi
ABSTRACT	1
ADMINISTRATIVE INFORMATION	1
INTRODUCTION	1
SCOPE AND OBJECTIVE	1
BACKGROUND	4
CURVED FINITE ELEMENTS	6
TWO-DIMENSIONAL CONTINUUM ELEMENT	6
Local Coordinates and Node Numbering Convention	6
Numerical Integration and Stress Computation	9
THREE-DIMENSIONAL CONTINUUM ELEMENT	11
Solid Element of Arbitrary Shape (ASOLID)	11
Three-Dimensional Shell Element (TDSHEL)	14
Ahmad's Thin Shell Element (THNSHL)	16
PROGRAM ORGANIZATION	18
INTRODUCTION	18
SOLUTION OF LARGE SPARSE MATRIX	22
ADVANTAGE OF FRONTAL TECHNIQUE	23
PROBLEM FORMULATION	23
INPUT DEFINITION	23
CORE-MEMORY REQUIREMENT	24
Subroutine STORIJ	25
Examples of Core-Size Calculation	30
ARRAY SPECIFICATION	38
INSTRUCTIONS FOR DATA INPUT	38
INPUT ITEMS AND OPTIONS	38
SAMPLE PROBLEMS	42

Page

Stresses in Plate Beams, Problem 1. 42
A Supercavitating Blade, Problem 2. 44
The Solid Foil of TAP-1, Problem 3. 51

APPENDIX A - SOLUTION PROCEDURE 59

APPENDIX B - INPUT LISTINGS OF SAMPLE PROBLEMS. 81

REFERENCES. 91

LIST OF FIGURES

1 - Views of a Highly Skewed Propeller Blade Idealized with Curved Finite Elements. 3

2 - Shapes and Attributes of an Element in Two-Dimensional Space. 7

3 - Geometry of a Two-Dimensional Element Showing Position and Order of Input Node Number. 8

4 - Integration Scheme 10

5 - Printing Sequence for Integration Point Stresses 12

6 - Stress Printout Sequence at Interior and Surface Points for Three-Dimensional Continuum Elements. 13

7 - Node Numbering Convention for Three-Dimensional Shell Elements 15

8 - Extended Curved Finite Elements. 17

9 - Illustration of Front Movement and Variable Front Width. 19

10 - Program Organization Map 20

11 - Storage Allocations Used in the Working Column, Vector ELPA. 29

12 - A Simply-Supported Square Plate. 30

13 - Finite Element Representation of a Plate Quadrant. 31

Page

14 - Two Schemes for Finite Element Representation of a Stiffened Plate. 32

15 - Plate Beam Sample Problem. 43

16 - Distribution of Normal Stresses in a Plate Beam for Sample Problem 1 45

17 - Curved Finite Element Representation of a Supercavitating Propeller DTNSRDC P-3604 47

18 - Supercavitating Blade Stresses for a Uniformly Distributed 1-Pound Per Square Inch Pressure over the Entire Blade Face 49

19 - Pictorial View of a Supercavitating Foil Model 52

20 - Planform of TAP-1 Foil and Strut 53

21 - Typical Pressure Distribution across a Chord Section of TAP-1 55

22 - Vertical Deflection of Foil Spans. 56

23 - Foil Stresses along Chord Section, 25 Percent Semispan, Top and Bottom Faces 57

24 - Bottom Surface Principal Stresses. 58

25 - A Finite Element Representation of a Two-Dimensional Problem. 67

26 - Occurrence of Nodal Variables in a Two-Dimensional Continuum Problem. 73

LIST OF TABLES

1 - Core-Memory Requirements, a Summary of Case Studies 28

2 - Dimensions of Standard Array 39

3 - Input Items for PBLADE Computer Program. 40

4 - Longitudinal Stresses (σ_y) in a Plate Beam 46

5 - Data Organization of a Front Procedure 68

NOTATION (INCLUDING INPUT TERMS)

APPEAR	Name of a subroutine used to keep track on the first, last, and intermediate appearances of a variable
C	Structure matrix of the load deflection equation, (Equation 3) C x = B
C^e	Stiffness coefficient matrix of an element, or element contributions to C
CM	Size of core memory, see "CORE-MEMORY REQUIREMENT"
CODEST	Subroutine to interpret coded destination, or the address of an element coefficient
COORD	Array of all the nodal coordinates in the structure
DISP	Array of nodal displacements (U, V, W)
E	Young's modulus of elasticity
e_s	Equation "s" used to eliminate variable x_s
ELPA	Long working vector comprising element contributions, loads, coefficients of assembled equations, etc; the storage diagram is given in Figure 11
G	Submatrix (Figure 9) covers the range of active variables; it is stored as a subvector of ELPA and contains the coefficients of assembled equations and the associated right-hand sides
INCEL	Matrix array representing element nodal incidence; the ordered nodal labels for each element are related to a set of shape functions used to prescribe the element displacement field. Consequently, they define the orientation of local curvilinear coordinates ξ, η, and ζ with respect to global axes X, Y, Z.
INCID	Options to rearrange the nodal sequence of a thick shell element (NNPE = 16) to conform with the format used in the current program
JCOUNT	Counts and labels the free nodal variables of a complete structural system and gives the total number of degrees of freedom

JDIS	Set of nodal labels for each individual element; the labels are regrouped from JDISP in an order conforming to the incidence array to ensure that the computed element coefficients C^e have proper labels and addresses; zero in JDIS represents constraints on displacement components U, V, and W
JDISP	Array of all nodal labels in the structure; it accounts for all free nodal variables (TOTAL DOF = ND * JN - JREST) and follows the order in which the joints were numbered
JREST	Number of displacement constraints in a structural problem
KL	KL = 1 to KUREL; covers the variables of the current element
KOUNT	Counts the number of appearances of a variable, in thousands, plus 1000 (see Subroutine APPEAR)
KUREL	Current element size, i.e., the number of variables in the current element
LABLE	Subroutine to list nodal variables of each element
LAST	Last appearance of a variable (Subroutine APPEAR)
LCUREQ	Number of variables already eliminated when the current element appears, i.e., the number of stored equations
LDES, LDEST	Element destinations are in LDEST array; LDES is the decoded version of LDEST (KL) (See Subroutine CODEST)
LNEL	When NF = 0, or 1, LNEL is the last numbered element where a surface pressure occurred.
LPB	Joint label at which printout begins for PLOAD array (if NF = 0, or 1)
LPT	Joint label at which printout for PLOAD array terminates
LPREQ	Number of stored equations when the previous element was assembled; used for output by elements
LVABL	Vector giving labels of element variables in the same order as set up in incidence array. They normally represent nodal displacement - components (e.g., u_i, v_i, w_i, etc.), or degrees of freedom. LVABL (m), m = 1, KUREL; note that the list covers the unknown variables of the current element (zeros, representing constraints, are deleted from the list)

LVEND	Dimension of LVABL
LVMAX	Maximum number of variables per element actually encountered, i.e., the size of the longest LVABL
LZ	Size of element segment, e.g., the current element segment extends from ELPA (1) to ELPA (LZ)
MAXELT	Maximum length of any element segment, i.e., MAXLETT = $\frac{NDOFPE\ (NDOFPE+1)}{2}$ + NDOFPE
MAXNIC	Maximum element label (nickname) encountered
MAXPA	Maximum size of front required in terms of the number of variables
MVABL	Array of active variables in front, or in the running variables used in the back-substitution
MVEND	Dimension of MVABL; it is problem dependent (see Subroutine STORIJ)
NBAXO +1	Position in vector ELPA marks the beginning of the buffer area reserved for equations, e_s
NBAXZ, NELZ, NPAR, etc.	Positions in ELPA; see Figure 11
NBLA	Control flag signals for stress printout; see instructions for Data Input
ND	Number of degrees of freedom per node; the nodal displacement components of a typical solid element are three, i.e., ND = 3. In dealing with a two-dimensional problem, each displacement vector consists of two independent, orthogonal components, ND = 2. In classical plates and shells theory, ND = 5
NDOFPN	Equal to ND
NDOFPE = ND*NNPE	The element degrees of freedom
NEL	Equal to NELEM
NELEM	Counts the elements from 1 to NELEMZ

NELEMZ	Number of elements used in the finite element representation
NELPAZ	Effective dimension of ELPA; see Figure 11
NELZ	End location of the element record in ELPA including the load column, or the element right-hand side (RHS)
NF	Control flag to input load or to compute nodal loads by the program; see instruction to data input
NFUNC	Function giving the position in the equivalent vector of term (I,J) in an upper triangular matrix (see Subroutine STORIJ)
NIC	Label for a variable, a nickname; always a positive integer
NIX	Vector using the same storage area as ELPA and is the main working area for the preliminary bookkeeping; it starts as a list of LVABL (m) arrays for successive elements
NIXEND	Usable length of NIX
NIZZ	Numerical count of the last label, for instance, let m_i = KUREL (I) the size of element i, then NIZZ = $\sum_{i=1}^{NELEMZ} m_i$
NJ	Highest numbered joint in a completely idealized structure
NLOADS	Number of loaded nodes when NF = 2
NNPE	Number of nodes assigned to an element
NPUNCH	Flag to punch a card deck for PLOAD ARRAY
NPT	Number of integration points in the Legendre-Gauss quadrature space
NPZ	Same as NPT; for integration along ζ-axis
NROW	Number of rows (see example of core size)
NSTRES	Number of stresses to be averaged, alias number of appearances of a variable
NSVJ	Number of midsurface nodes; those nodes for which coordinates are to be computed by the program (see Input Definition)

NTIREX	1 or 0 depending whether all the variables representing $\underline{U}$ (U, V, W) are presented together in output, or one element at a time (e.g., NTIREX=1, output entire displacement vector $\underline{U}$)
NVABZ	Total number of variables at the end of Subroutine APPEAR; it is found by counting the variables as they are eliminated, element by element, in the preprogram
NZ	Sum of free variables in all the elements being processed
P	PLOAD are regrouped for each element (NF=2)
PLOAD	Array of nodal loads imposed on the structure
PRATIO	Poisson's ratio
RHS	Right-hand side of an equilibrium equation, for instance, the components of nodal load
STFNS	Element stiffness array, or matrix C^e
STIFF	Subroutine to compute element stiffness matrix, C^e
STORIJ	Subroutine to establish storage requirements
T,TEMP	Constraints
$\overline{U}_i$	Vector of nodal displacement at node i, $u_i\hat{i} + v_i\hat{j} + w_i\hat{k}$
u,v,w	Components of displacement in the direction of x, y, and z axes, respectively; displacements are positive in the positive direction of coordinate axes
Vol	Volume of a given solid domain
X	Regrouping of COORD by element, for the purpose of computing element stiffness coefficients STFNS
x,y,z	Global system of rectangular coordinates

ABSTRACT

The present report provides the detailed instructions to perform a structural analysis using the curved finite element computer program - PBLADE. It defines input variables and format. It discusses the idealization of structures, their geometrical and material properties, the scope of computations and the core size requirements. It further discusses the effect of computational algorithm employed in the development and organization of the program. An example of the data formation in a well-known frontal solution procedure is described in detail to allow further exploitation of the efficient algorithm. Sample problems are given to illustrate applications and capabilities of the program to solve complex structural problems of a three-dimensional nature. Numerical results are presented to demonstrate the effectiveness of the programed computation.

ADMINISTRATIVE INFORMATION

A numerical solution to complex structural problems, including that of marine propellers, was developed in the Analytic Method Group of the Surface Ship Division, Structures Department. The work was funded by the Naval Sea Systems Command Materials and Mechanics Division (03511) under Task Area SF 43 422 505, Task 17934 (Structures for Hydrodynamic/Aerostatic Lift Ships). This task has since been incorporated under the Advanced Ship Structures Block Program, SF 43 422 593, Work Unit 1-1730-312.

INTRODUCTION

SCOPE AND OBJECTIVE

Recent developments in machine computation have made possible a modern tool for analyzing complex three-dimensional structures. One major objective of this report is to increase the awareness of user-engineers that emergence of finite element technology has provided a practical approach for calculating loads and predicting structural behaviors. The basic concept of the finite element method is that of a real continuum which can be treated analytically by subdividing it into a finite number of regions. In each of these regions, displacement and/or stress is described as an individual field. These fields are often chosen in a form that ensures continuity of the selected variables throughout the whole body (continuum).

The marine propeller in its general form is a good example of a complex, three-dimensional structure. The geometry of the propeller blade normally determines the choice of a finite element. Elements of planar, elementary shapes are sometimes used. The application of these elements can be difficult, for instance, in the case of a skewed blade where blade surfaces are in a form of complex curvatures. Its face is incongruous with its back. Figure 1 illustrates the topology of a highly skewed blade. It is in those practical applications that the present finite element program has proved to be highly efficient.

A finite element displacement model was utilized to predict the elastic behavior of a propeller blade having an arbitrary shape and subjected to prescribed loading. Solid elements in their general form were adapted, and the reference to a set of natural curvilinear coordinates was introduced. It will be shown that the use of curvilinear coordinates in element space provides both a practical means for defining complex design surfaces and an expedient method for stress calculations. The curved three-dimensional elements fit readily to a skewed geometry or curved boundary, and their application to propeller problems is simple and straightforward.

The high degree of accuracy attained when curved elements were employed in a recent analysis of a full-size propeller blade, strongly suggest that the current development represents a realistic and reliable approach to the general solution of the propeller stress problem.[1*]

The report includes an outline of the organization of computer program PBLADE, an explanation of input definitions and data setup, sample problems to illustrate the usage of solid finite elements, and the analysis of complex structural problems of a three-dimensional nature to demonstrate the effectiveness and potential of the curved finite element program.

The present report also discusses the topics that are important for the numerical solution of large-structure matrices. Current theoretical and practical considerations essential to the efficient application of digital computation are delineated and illustrated to ensure that the user obtains an effective analysis of novel structural problems.

---

*A complete listing of references is given on page 91.

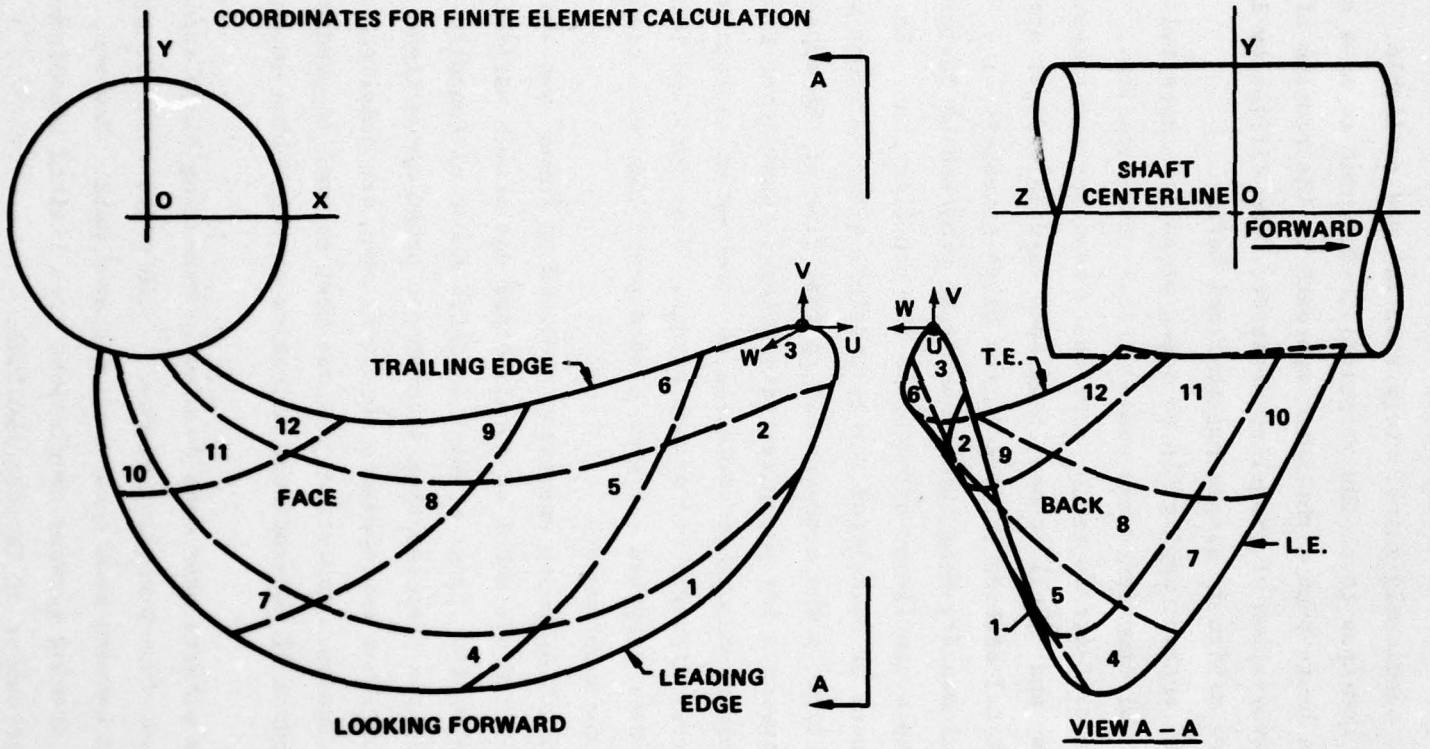


Figure 1 - Views of a Highly Skewed Propeller Blade Idealized with Curved Finite Elements

BACKGROUND

Marine propeller blades constitute a special class of structural problems for which no completely satisfactory solution was available. Although screw propellers continue to be the principal device used to move a ship, only recently has there been a realistic approach to the problem of obtaining an accurate evaluation of propeller stresses. The difficulty lies in describing a blade design in simple mathematical terms.

Most existing methods applicable to screw propellers have relied heavily on practical experience and semiempirical considerations. These provide a criterion of stress rather than the actual surface stresses. Both "beam" theory and "shell" theory have been used in earlier attempts to develop analytical methods for predicting blade stresses.

The use of elementary beam theory was first proposed by Taylor[2] who treated a blade as a cantilever attached to the propeller hub. He recommended that stresses be calculated for cylindrical blade sections with the neutral axis parallel to the nose-to-tail (pitch) line of the expanded section, see Reference 3 for propeller terminology. Cantilever beam theories have yielded reasonable estimates of stresses at certain selected points of relatively straight and narrow blades. Some modified forms of beam theory have been proposed for wide-bladed propellers with blade width-to-length ratios of about one.

The shell theory approach was first proposed by Cohen[4] who treated a simplified propeller blade model as a helicoidal shell with variable thickness and infinite width. Later studies included those of Connolly[5] and Atkinson.[6] Shell-type theories that incorporate broad assumptions do not appear to offer tangible improvement; moreover, they are complicated for routine design purposes. Analytical methods that attempt to predict blade stresses on the basis of conventional mechanics have not been eminently successful.

Considerable efforts have been devoted to measuring blade strains on both model and prototype propeller blades.[7,8] In certain cases, good agreement was obtained between beam theory and measured data. However, care must be taken in drawing general conclusions from limited measurements because of the large number of factors involved.

The trend in shipbuilding to full afterbodies for mammoth tankers and bulk carriers and to higher speeds for modern naval vessels has been accompanied by large irregularities and fluctuations in ship wakes. The thrust derived from blade-lift force is unsteady when the blades rotate in a non-uniform velocity field behind the ship. The interaction of these unsteady forces with the hull and appendages causes the excitation of the ship by the propellers. Blade skew, high blade area ratios (i.e., wider blades), and a large number of blades per shaft have all been tried in an attempt to reduce vibration. These innovations of propeller geometry drastically alter blade displacement patterns[9,10] and render the standard methods (i.e., beam theory) invalid.

A rational approach to the general solution of the propeller stress problem has been established.[11] It involves the use of a finite element displacement model to predict the behavior of an elastic body with an arbitrary shape under static loads (and the adoption of compatible solid elements in their general form). The formulation bypasses the constraints of simplifying assumptions and allows a closer approximation to the true structural configuration than is possible with most other approaches, for instance, by using classical plate or shell theories. Solutions for displacements and internal stresses can be obtained subsequently.

The numerical procedure developed by Ma[11] is completely general in nature and provides a full three-dimensional stress analysis of a structure. The solid elements employed in the computer program can represent the correct behaviors of a beam, a plate, a shell, or any of the varied aspects of structural components. Furthermore, there is no restriction on their geometry; the finite element program is just as applicable to the stress of a supercavitating blade as to the stress of a standard propeller blade. Detailed stress calculations at the root section of a propeller blade joining the hub can now be performed in a realistic manner.

Detailed descriptions of the curved finite elements, the numerical technique, and the computer program are given in the following sections.

CURVED FINITE ELEMENTS

TWO-DIMENSIONAL CONTINUUM ELEMENT

The nature of curved isoparametric elements in a two-dimensional continuum will be discussed first for the convenience of illustration, Figure 2. Generalization into full three-dimensional elements is immediate. These two-dimensional continuum elements, (CURVPL) can also be made available.

The element properties are represented by the element stiffness matrix $[C^e]$* where

$$[C^e] = \int [B]^T [D] [B] \, dvol \qquad (1)$$

depends on the geometry and the material properties of an element. Here the strain matrix [B] is a function of element geometry and the elasticity matrix [D] a function of element material.[11]

The geometric shape and size of an element are prescribed by element boundaries which connect a set of nodal coordinates normally referenced to global axes x, y, z; see Figure 3.

Local Coordinates and Node Numbering Convention

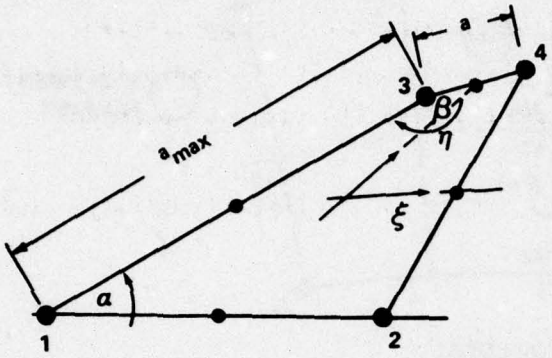
The distinct advantage of a curved element lies in its capability to take on arbitrary shapes. An auxiliary coordinate system, known as local curvilinear coordinates (ξ,η), is utilized to define the element space (for details, see Reference 11). Its orientation with respect to the global axes is established in each element by the sequence of the first three labeled nodal numbers used as input to the incidence table. The incidence table of nodal labels is read early by the main program to ensure proper connectivity of individual elements.

The integration of element stiffness coefficients, Equation (1), can be conveniently carried out by means of a Gaussian quadrature:

---

*The element stiffness coefficients were designated as K_e in Reference 11.

IT IS DESIRABLE TO HAVE AN ELEMENT SHAPE BOUNDED BY THE CRITERIA:

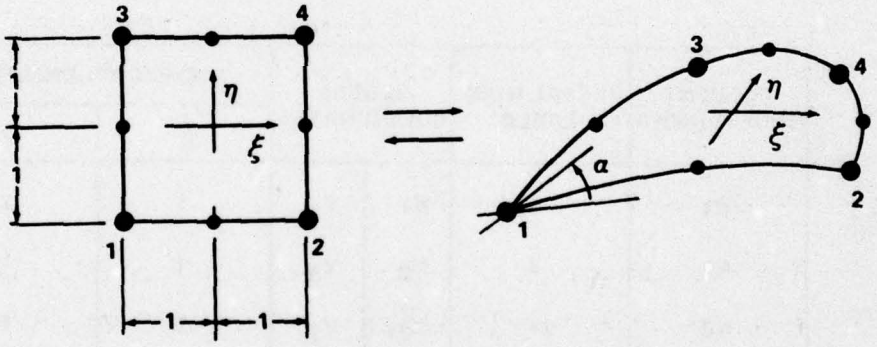


$\frac{a_{max}}{a} < 5$

$\alpha > 15$ DEG

$\beta < 180$ DEG

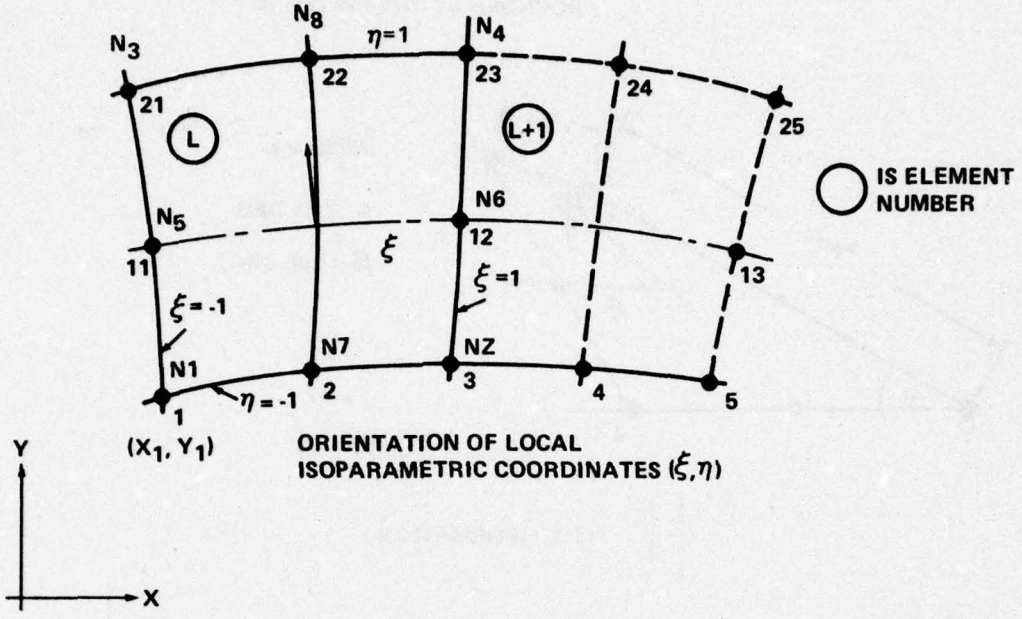
TRANSFORMATION



THE ELEMENT IMAGE IS A SQUARE, IDEAL FOR GAUSS QUADRATURE

THE GEOMETRY OF THE LOCAL COORDINATE LINES (ξ, η) IS DETERMINED BY THE SHAPE OF ELEMENT BOUNDARIES. TRANSFORMATION AND INTEGRATION (OR QUADRATURE) ARE DUE PROCESS IN ELEMENT STIFFNESS FORMATION

Figure 2 - Shapes and Attributes of an Element in Two-Dimensional Space



ELEMENT NODE NUMBER	GLOBAL NODE LABLE	GLOBAL COORDINATES		LOCAL COORDINATES	
				ξ	η
N1	1	X_1	Y_1	-1	-1
N2	3	X_3	Y_3	1	-1
N3	21	X_{21}	Y_{21}	-1	1
N4	23	X_{23}	Y_{23}	1	1
N5	11	X_{11}	Y_{11}	-1	0
N6	12	X_{12}	Y_{12}	1	0
N7	2	X_2	Y_2	0	-1
N8	22	X_{22}	Y_{22}	0	1

Figure 3 - Geometry of a Two-Dimensional Element Showing Position and Order of Input Node Number

$$\int_{-1}^{1} f(\xi)d\xi = \sum_{i=1}^{n} H_i * f(a_i) \qquad (2)$$

where n is the number of integration points (NPT). The application of a Gaussian quadrature requires a set of abscissae (a_i) and weight coefficients (H_i). These can be found in standard textbooks of numerical methods.[12,13] The position of the first few quadrature points is illustrated in a sketch to provide a physical image of integration. See Figure 4a.

Numerical Integration and Stress Computation

The integration of a surface area is

$$\int_{-1}^{1}\int_{-1}^{1} f(\xi,\eta)d\eta d\xi = \sum_{i=1}^{3} H_i \sum_{j=1}^{NPT} H_j f(a_i,a_j)$$

where the expression $f(\xi,\eta) = [B]^T [D] [B] |J^*(\xi,\eta)|$ is evaluated point by point in an element subregion. A three and two integration rule is indicated for the given element, Figure 4b. There are six integration points, and they are related solely to the local curvilinear coordinates (ξ,η) which, in turn, are defined by boundary nodes. An option is available to rotate the local coordinate axis (ξ) to axis (η) by setting INCID = 2 (see input term, INCID, NROT). The option is executed by realining the sequence order of nodal labels of the incidence table.

For a quadratic element, such as the 8-node element in Figure 3, the 3-point rule is the norm for integration. Higher orders of integration require more computational time without offering appreciable improvement. On

---

*Det $|J|$ is the determinant of a Jacobian Transformation and is a function of element shape and size (Figure 2). Its numerical value is computed at each integration point in an element space and consequently exerts a quantitative influence over the accuracy of element stiffness and of resulting stress. For details see Section 3.22 of Reference 11.

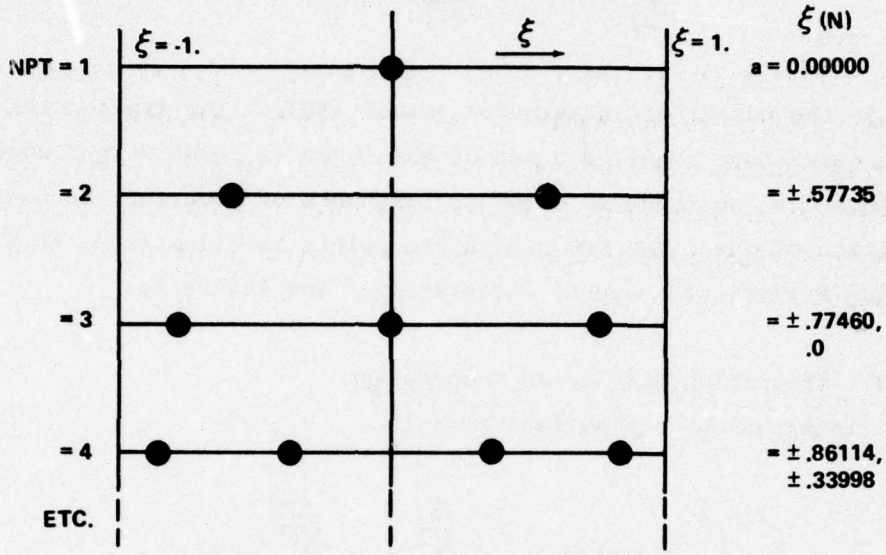


Figure 4a - Positions of Integration Points

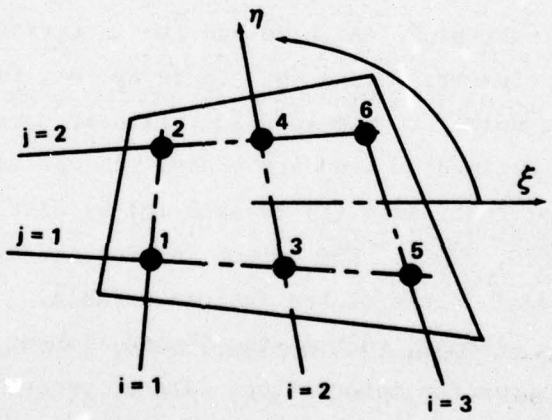


Figure 4b - Integration Points in an Element Space

Figure 4 - Integration Scheme

the other hand, a reduced integration, e.g., a 2-point rule, may hasten the convergence in certain kinds of structural shapes or configurations but users, especially beginners, are advised to be cautious in their applications.

The calculation of element stiffness matrix $[C^e]$, expressing the nodal force-displacement relationships of a unit element, also provides a framework for evaluating the various aspects of body stresses. The strain matrix [B] (which is computed at each integration point in the process of forming a stiffness matrix) is readily available for the stress computation,[1] i.e.,

$$\underline{\varepsilon} = [B]\ \underline{U}^a$$

and

$$\underline{\sigma} = [D]\ \underline{\varepsilon}$$

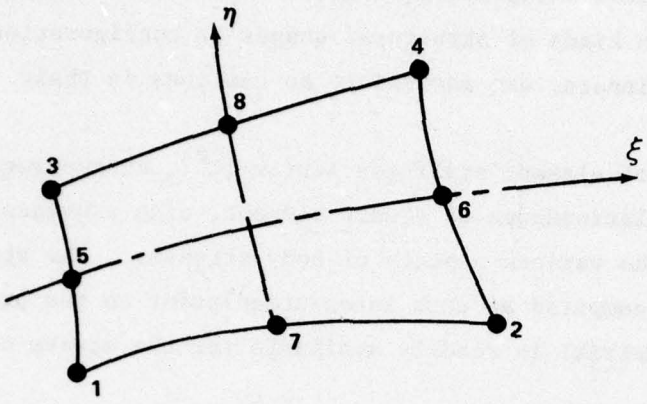
These stresses located at integration points within the element space are printed in the same order as the integration taken place; see Figure 5.

These stresses usually represent good quality numerical results. Boundary stresses can be obtained by additional computations of the strain matrix at specific points. Some optimization procedures, for instance, the least square interpolation, can be used to predict stresses at boundary nodes.

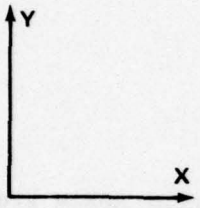
THREE-DIMENSIONAL CONTINUUM ELEMENT

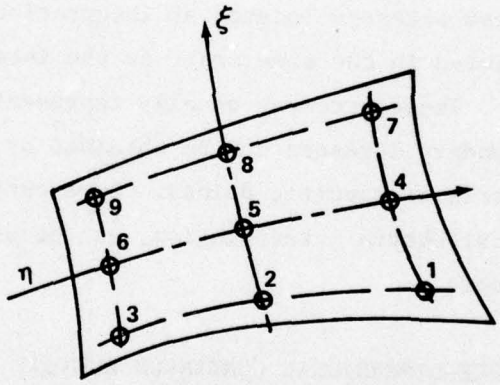
Solid Element of Arbitrary Shape (ASOLID)

A typical curved element of three-dimensional continuum is illustrated in Figure 6. The solid element is most general in nature and is capable of modeling bodies of arbitrary geometry. For complex design problems, (e.g., the blade interface with the hub, the nozzel intersection of a pressure vessel, etc.) the versatile three-dimensional element probably offers the most realistic solution. The orientation of local isoparametric coordinates (ξ,η,ζ) and the order of nodal numbers are also given in

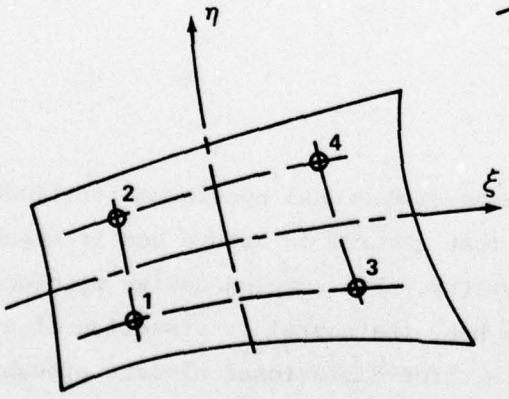


ELEMENT NODE NUMBERING CONVENTION





3–POINT INTEGRATION



2–POINT INTEGRATION

Figure 5 - Printing Sequence for Integration Point Stresses

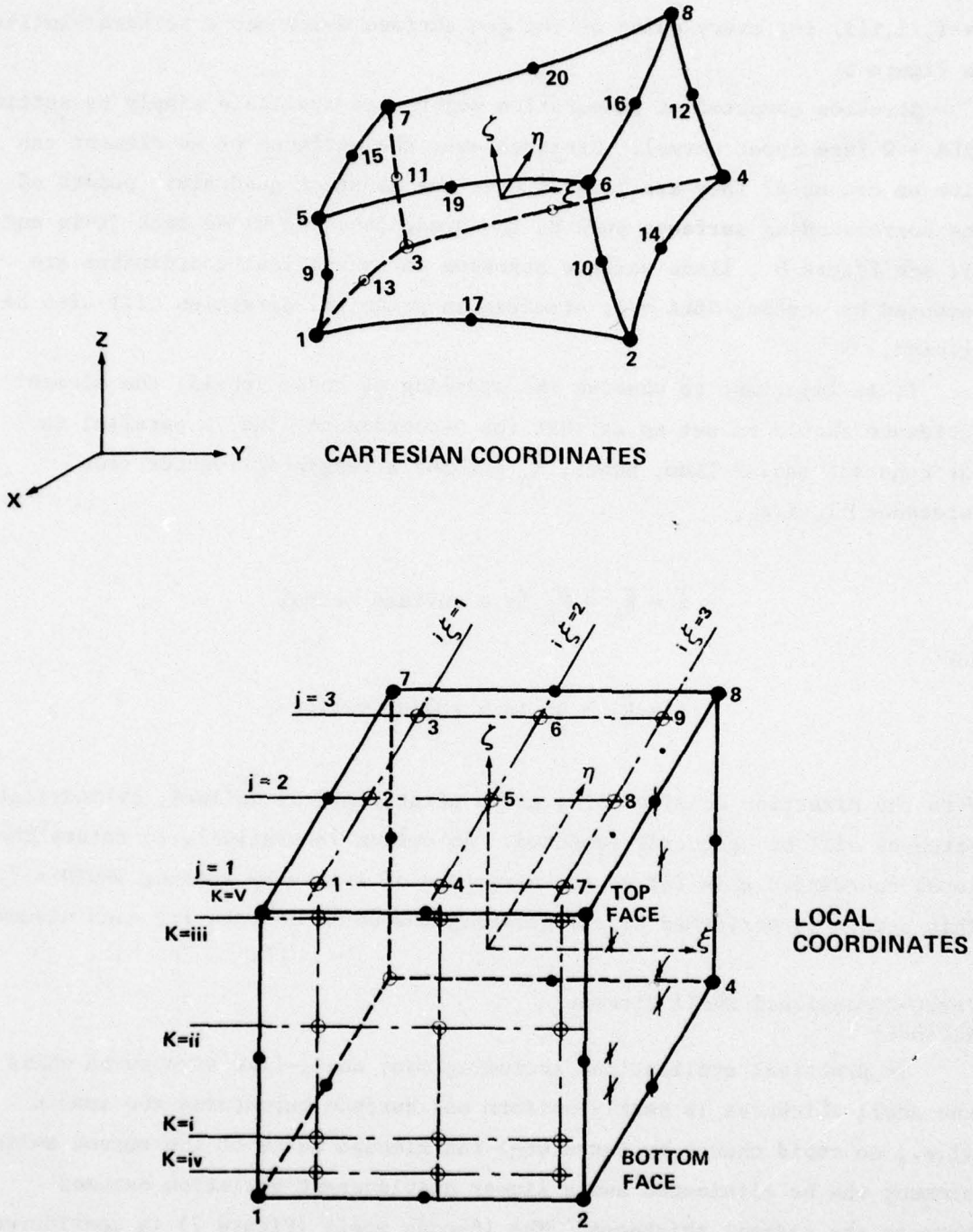


Figure 6 - Stress Printout Sequence at Interior and Surface Points for Three-Dimensional Continuum Elements

Figure 6. The integration points now include those along the ζ-direction (k=i,ii,iii) for every point on the ξ-η surface which has a pattern similar to Figure 5.

Stresses computed at integration points are available simply by setting NBLA = 0 (see input terms). Stresses over the surfaces of an element can also be computed; they are located over the Gaussian quadrature points of the corresponding surfaces such as the blade face and blade back (k=iv and v); see Figure 6. Blade surface stresses in cylindrical coordinates are computed by setting NBLA = 1; stresses in principal direction will also be printed.

It is important to observe the ordering of nodal labels; the element incidence should be set up so that the η-coordinate line is parallel to the constant radius line, hence, $\bar{R}_\eta$ becomes a tangential vector (see Reference 1), i.e.,

$$\bar{n} = \bar{R}_\xi \times \bar{R}_\eta \text{ is a surface normal}$$

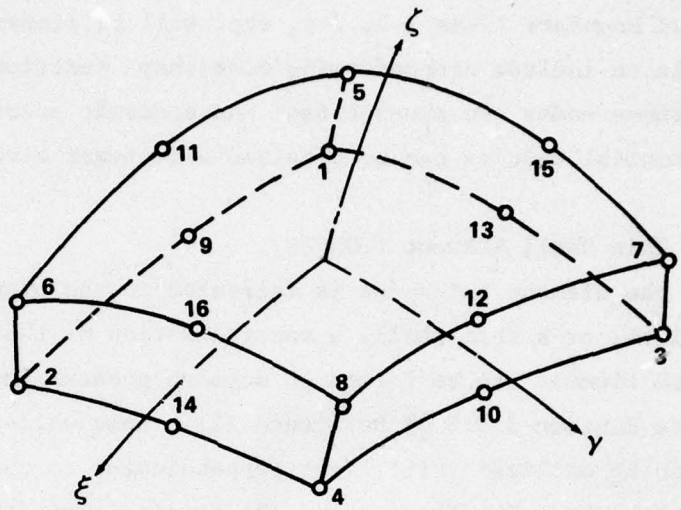
and

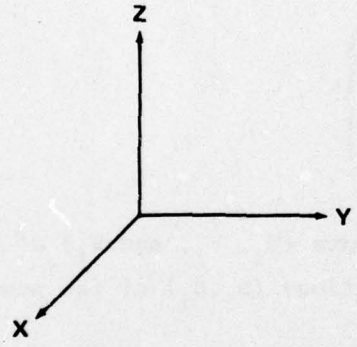
$$\bar{r} = \bar{R}_\eta \times \bar{n} \text{ is a radial vector}$$

With the direction consines at a given point properly defined, cylindrical stresses will be correctly computed. An option is available to rotate the local coordinate axis (ξ) to the direction of η-axis by setting INCID = 2. This option is performed by resequencing the nodal numbers for each element.

Three-Dimensional Shell Element
(TDSHEL)

In practical applications including many shell-like structures where the shell thickness is nearly uniform and surface curvatures are smooth (i.e., no rapid change in curvature) the midedge nodes of the curved solid element can be eliminated and a linear displacement variation assumed through the element thickness. The 16-node shell (Figure 7) is considered representative of the three-dimensional shell element group. The





THERE ARE ALSO 12– AND 8-NODE SHELL ELEMENTS WITH OPTIONS TO INCLUDE NONCONFORMING MODES

Figure 7 - Node Numbering Convention for Three-Dimensional Shell Elements

orientation of local curvilinear coordinates (ξ,η,ζ) and the positions of the ordered nodal numbers are also shown in Figure 7. A 12-node three-dimensional shell is obtained by further deleting nodes 13, 14, 15, and 16. The displacement for this element varies linearly along the η-coordinate line, and boundary lines 1-3, 2-4, etc. will be linear. An option is available to include nonconforming mode-shape functions. When properly chosen these modes can provide fast and economic solutions; in other words, good numerical results can be obtained with fewer elements.

Ahmad's Thin Shell Element (THNSHL)

As the element thickness is decreased to the proportion of a medium thick plate, or a thin shell, a specialization of the three-dimensional continuum element can be formed to achieve greater economy and effectiveness (see Section 3.3.3 of Reference 11). Some well-known shell assumptions can be utilized: (1) lines perpendicular to the middle surface remain straight under loading and (2) strains along these lines can be ignored in the energy summation. At each node of the shell (Figure 8), there are now five degrees of freedom:

$$\overline{U}_{sh} = \begin{Bmatrix} u \\ v \\ w \\ \alpha \\ \beta \end{Bmatrix}$$

They represent three translational components (U_i, V_i, and W_i) of displacement at the midsurface node i and two rotations (α_i,β_i) of the nodal normal $\hat{V}_{3i}$.

The transition element is a further extension of the three-dimensional continuum element development which combines a curved solid element and an Ahmad shell element. It is used to connect the Ahmad shell to the three-dimensional solid. Its stiffness is derived by relating the five degrees of freedom of each midsurface shell node to the displacements of the

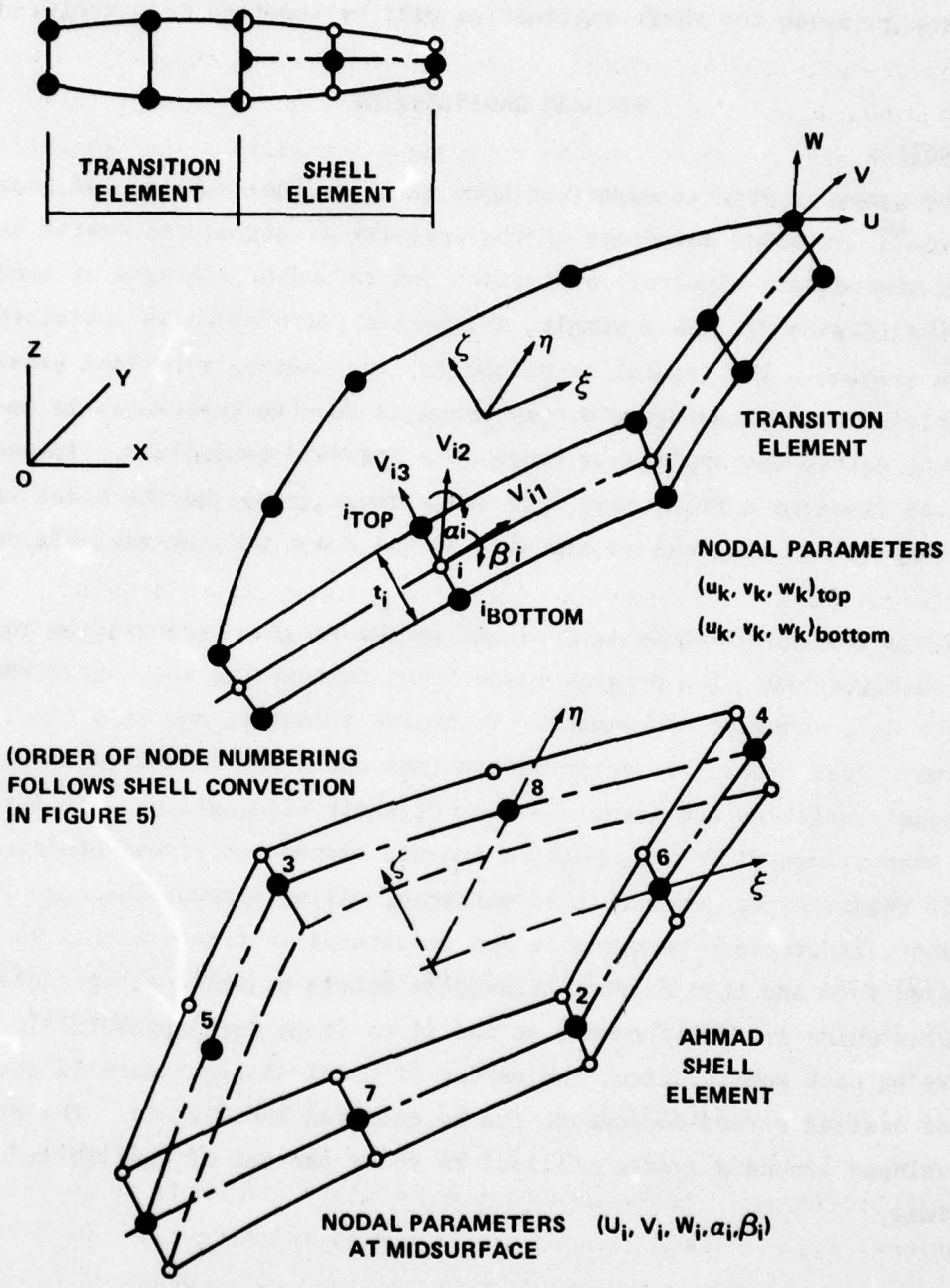


Figure 8 - Extended Curved Finite Elements

corresponding pair of nodes on the top and bottom surfaces of the connecting three-dimensional shell element (Figure 8). The performance of these elements and guidance for their application will be reported separately.

PROGRAM ORGANIZATION

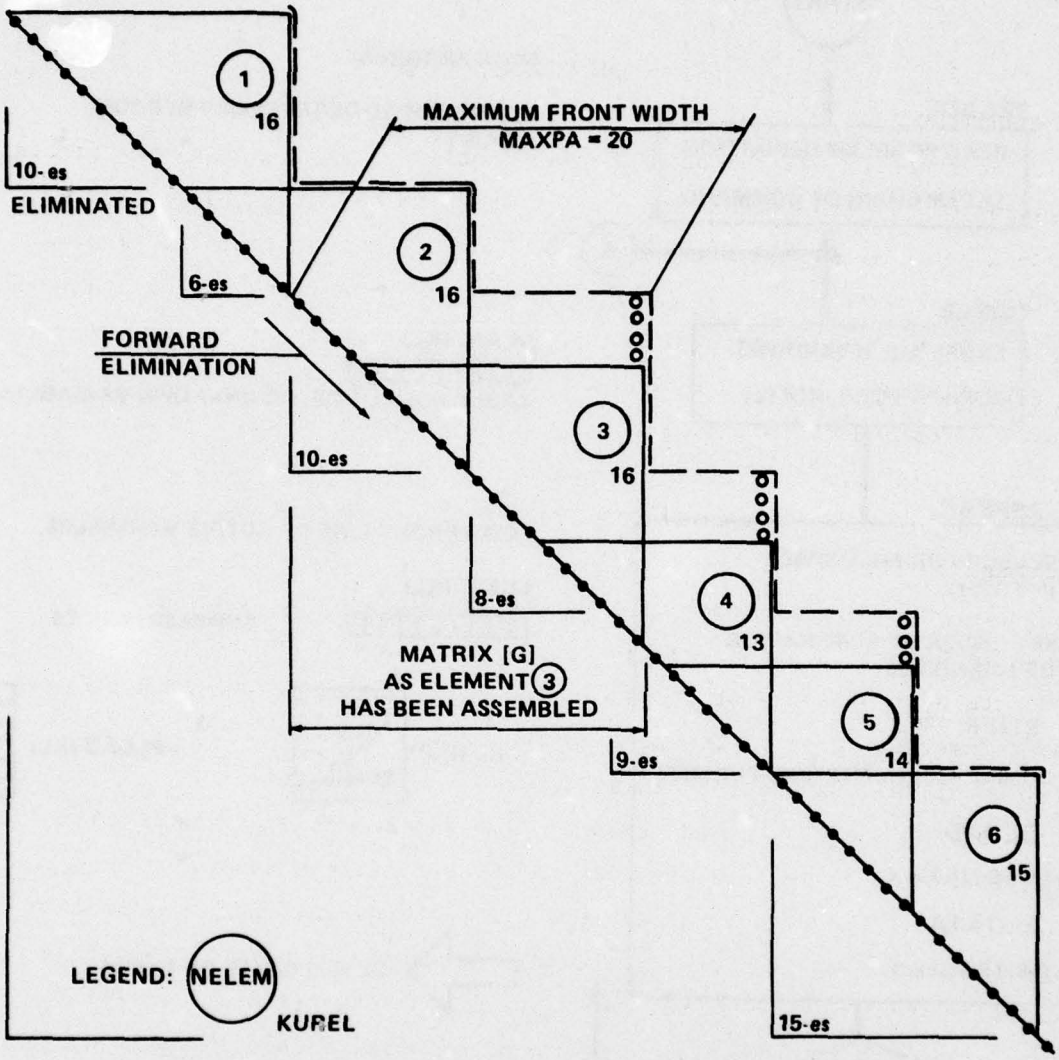
INTRODUCTION

The computer program described here utilizes the technique of frontal solution.[14] It takes advantage of the sparsity of structural matrix by evading unnecessary algebraic operations and redundant linkages of nodal variables (Figure 9). As a result, arithmetic performance is optimized for a given problem. The frontal technique is, in essence, a refined version of partitioning a structure commonly known as substructure which is employed to extend the applicable range of a standard bandsolver. Indeed, the front dissects a given body into two element groups having nodal variables completely processed on one side of the front and raw variable on another.

The accent on elements is apparent in the Program Organization Map shown in Figure 10. The program reads input through the subprogram PBLADE,* and sets up a sequence of command. It admits elements, one at a time, in their numerical order. It calls Subroutines LABLE and APPEAR to label the free nodal variables and forms an order of their eliminations. It further calls subroutines STIFF and GLOAD to compute element stiffness coefficients and, if required, to generate load matrices. After sizeup, the core requirement, it proceeds to assemble the structural stiffness matrix in a segmented form and then to form triangular matrix by eliminating those variables which are fully summed at the given stage (see Appendix A). Finally by back substitution, the vector of nodal displacements is obtained, and desired stress-components can be computed immediately. The program is developed around a system utilized to solve the set of equilibrium equations.

---

*All program and subroutine names are underlined in Figure 10.



ELIMINATION OF UNKNOWN VARIABLES IN A SYMMETRIC POSITIVE-DEFINITE STRUCTURE MATRIX [C] FOR A TWO DIMENSIONAL PROBLEM HAVING 58 D.O.F.
(FOR DETAILS OF A FINITE ELEMENT REPRESENTATION SEE P. 67)

Figure 9 - Illustration of Front Movement and Variable Front Width

Figure 10 - Program Organization Map

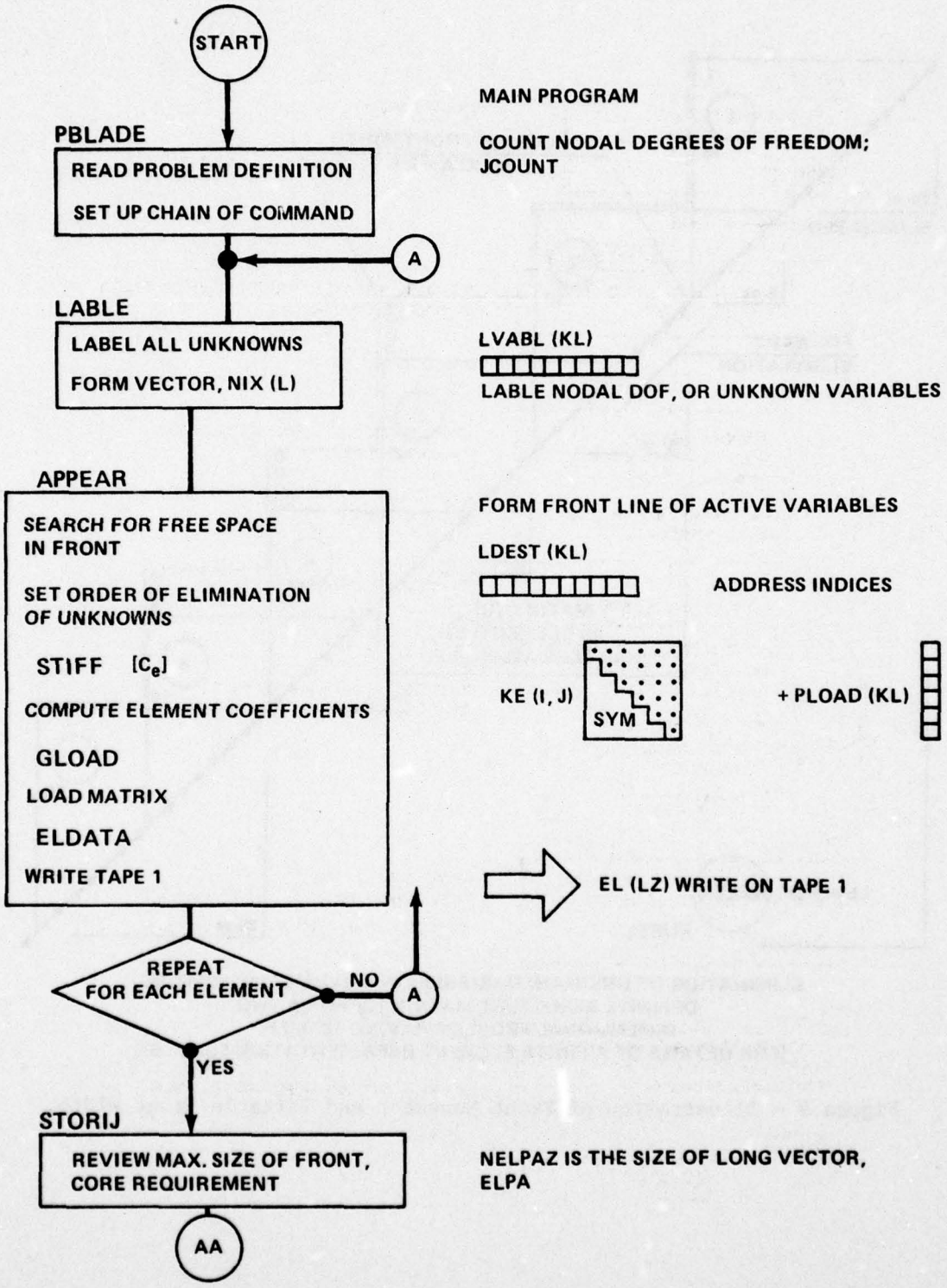
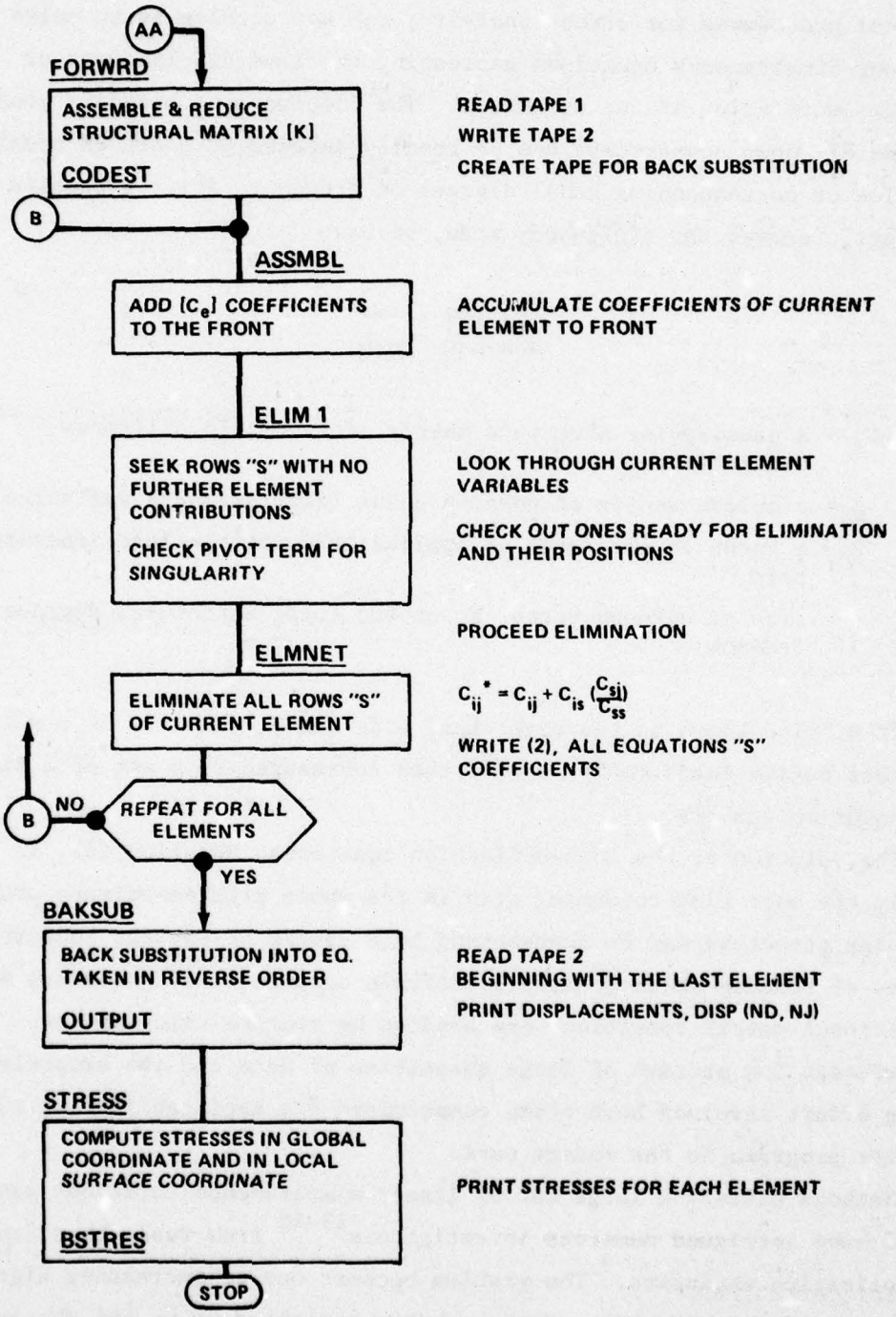


Figure 10 (Continued)



SOLUTION OF LARGE SPARSE MATRIX

In the displacement method of finite element analysis as well as other numerical procedures for stress analysis, the key problem is to solve a set of linear simultaneous equations expressing the load-displacement or equilibrium relation of the structure. The displacement boundary conditions in terms of known constraints can be readily imposed on a set of nodal variables or corresponding nodal degrees of freedom. After reduction which, in effect, removes the rigid-body mode, we have

$$\underset{n\times n}{[C]}\ \underset{n\times m}{[X]} = \underset{n\times m}{[B]} \tag{3}$$

where [C] = a nonsingular structure matrix of symmetric stiffness coefficients

$\underline{X}$ = a column vector of unknown nodal (displacement) variables

$\underline{B}$ = a known vector (m=1) of applied load with at least one nonzero term

n = size of unknown vector X, or the total structural degrees of freedom

If $\underline{B}$, also known as the right-hand side (RHS), consists of a set of alternate design loads (m>1), X will then correspond to a set of m displacement solution vectors.

The solution of the load-deflection equations, Equation (3), is usually the most time-consuming step in the whole problem-solving process. A complex structure may be represented by a system of several thousand degrees of freedom which greatly overstrain computational processes wherein conventional matrix operations are handled by routine calculations. The requirements for storage of large quantities of data and the excessive computing effort involved have often compromised the applicability of such computer programs in the recent past.

Methods whereby a large set of linear simultaneous equations can be solved have intrigued numerous investigators[15-20] from research scientists to application engineers. The problem becomes one of increasing significance and scope as the size of the unknown variables grows and the system

design of a digital computer becomes involved. The mode in which the massive amount of data is generated, stored, and, if necessary, rearranged, has a strong influence on the speed of the analysis, the number of arithmetic operations, and the computer core-size requirements, etc. Further, it dictates the type (or shape characteristics) of a structure and the format of data input that can be admitted for an efficient numerical solution. In sum, the (eventual) effectiveness of a program development hinges on the selection and implementation of an efficient computational algorithm to solve Equation 3.

ADVANTAGE OF FRONTAL TECHNIQUE

One practical advantage of the program is the ease with which a node system can be numbered. The nodal label is simply a nickname and places no constraint on the solution process. Normally, when a branched structure is treated by a bandsolver, a rather cumbersome numbering scheme has to be employed to optimize the bandwidth. But to a frontal technique, any set of nodal numbers which conveniently covers the topology of a structure is satisfactory and is as good as any alternative. In summary, it is worth reiterating that the order of elements is an important consideration for the application of the present program but that the order of node numbering is immaterial. (For example, see Case (2) of Examples of Core-Size Calculation.)

The proper rule for element numbering and an assessment of its influence on core requirements will be outlined in the subroutine STORIJ. First, however, the topic of defining a structural problem to be processed by the program will be discussed.

PROBLEM FORMULATION

INPUT DEFINITION

The function of the MAIN program is to establish a logical sequence of computation. It reads a set of input data which define the geometrical and mechanical properties of a structural problem together with loading and support information. The data also indicate the options selected by the user for the execution of programed calculations.

Input data consist of a problem title,* properties of material, and a set of control parameters essential to describe the problem. The control parameters include 24 input terms read from four data cards. They define the problem size (NJ, NELEMZ), nodal connectivity pattern (INCID), loading (NF, NLOADS) and support conditions (JREST). Further, they define the size of principal arrays (NELPAZ, etc.), integration scheme (NPT), printing options, and generation of intermediate nodal data (GEOORD) where required.

Specifically, the parameter NF prescribes the load input option; known nodal loads (NLOADS) can be read or, alternatively, equivalent nodal forces are calculated, element by element, for a design pressure over a given surface of the body. INCID provides a pattern of connectivity and the orientation of individual elements by way of element incidence. Nodal coordinates define the position and boundary surface of the structure in terms of a set of global coordinates (COORD). The array size MVEND poses a measure of front width which depends on the number of sequential elements that intervene between a pair of adjacent elements anywhere in the discretized structure. The boundary constraints (JREST) and the number of free nodal variables (JCOUNT) are counted. Problem specifications including the geometrical and material description will be reiterated as output. They include the coordinate and incidence table, the loading input, and the joint constraints.

The numerical solution of nodal displacements (DISP) is obtained through a group of subroutines up to and including BAKSUB (Figure 10). The stress components with respect to the global coordinates are immediately available at integration points. Stresses referenced to an arbitrary surface can be included in the output. Other stresses of interest, such as principal stresses, can also be computed at surface points.

CORE-MEMORY REQUIREMENT

One major consideration in the application of the computer program is the core size required to solve a given problem. The core requirement depends on the complexity of the structural configuration and the number

---

*See Instruction for data input.

of degrees of freedom needed to represent it. The matter of concern is whether the problem can be solved in sufficient detail, i.e., whether a certain degree of accuracy, which is frequently the objective of an analysis, can be realized. In any event, the capacity of the onsite computer is always a first order consideration.

The following sections explain Subroutine STORIJ in which the size of the major working array has to be defined. Examples are given to illustrate how to estimate core memory CM required to solve a given problem as well as certain features which are unique to a front solution. A summary of case studies based on execution on a CDC 6600 computer is included in tabulated form with core memory requirements indicated in Table 1. Finally, a list of typical array sizes is given. From these data, the reader can select appropriate array specifications and core size when a novel problem arises.

Subroutine STORIJ

It has been noted that only the stiffness coefficients [G]* associated with the active variables MVABL need to be readily available in the working area (see solution method). The core space of the primary working array ELPA, for instance, should be able to cover, among others, the maximum range of coefficients that can be included in a front [G] and the element stiffness contributions corresponding to a maximum element size (LVMAX).

The space allocation of the working array ELPA is segmented as shown in Figure 11. Key positions are marked as NELZ, NPAR, NPAX, etc. As each element in succession is being processed, the stiffness coefficients of the current element are read into the first segment and distributed into proper location of the third segment which begins from position NELZ + 1. The address or position index along the chainlike vector array, corresponding to a coefficient located at (I,J) of an upper triangular matrix, can be obtained with the aid of a line function NFUNC (I,J):

$$\text{NFUNC}(I,J) = I + \frac{J(J-1)}{2}$$

$$= \sum_{L=1}^{J} \text{ICOL}(L)$$

*For definition see NOTATION.

```
*DECK STORIJ
      SUBROUTINE  STORIJ
C
C     THE SUBROUTINE ESTABLISHES STORAGE REQUIREMENTS AND BOUNDS WITHIN
C     THE WORKING ARRAY ELPA
C
      COMMON /VAB1/NELPAZ,LVEND,MVEND,NIXEND,LVMAX,NIZZ,MAXNIC
      COMMON /VAB2/MAXPA,NVABZ,LCUREQ,MAXELT,NTIREX,LDES,KL,NSTRES
      COMMON /VAB3/NELEM,NELEMZ,KUREL,NIC,LPREQ,NEW,NPAR,NBAXO,NBAXZ
      COMMON /VAB6/CONST,NJ,NRUNO,LHSRHS,L,KOUNT,NELZ,NDELT
C
       NFUNC(I,J)=I+(J*(J-1)/2)
C
       NELZ = NFUNC(0,LVMAX+1) + LVMAX
       IF(NELZ.GT.MAXELT)  NELZ = MAXELT
       NPAR = NFUNC(0,MAXPA+1) + NELZ
       NPAZ = LVMAX + MAXPA
       IF(NTIREX.NE.0)  NPAZ = MAXNIC + MAXPA
       N = NPAR + (MAXPA*2)
       IF(N.GT.NPAZ) NPAZ = N
       NBAXO = NPAZ + 1
       NBAXZ = NBAXO + 3*MAXELT
       IF(NBAXZ.GT.NELPAZ) NBAXZ = NELPAZ
       NBUFFA = NBAXZ - NBAXO
       N1 = MAXPA + 4
       WRITE (6,26) NTIREX
   26  FORMAT(1H0, *NTIREX =  *,I4)
       WRITE (6,23)
   23  FORMAT(1H0,10X,* ELEMENT STORAGE REQUIREMENTS*,//,9X,6H LVMAX,9X,
     1 6H MAXPA,9X,6HMAXNIC,9X,6H  NELZ,9X,6H NPAR ,9X,6H NBAXO,9X,6H NB
     2AXZ,9X,6HNELPAZ ,//)
       WRITE (6,22) LVMAX,MAXPA,MAXNIC,NELZ,NPAR,NBAXO,NBAXZ,NELPAZ
   22  FORMAT(1H0,///(8I15))
C
       IF(NBUFFA.LT.N1)  GO TO 20
       NRUNO = NPAZ - MAXPA
       RETURN
   20  NBAXZ=N1+NBAXO
       WRITE(6,21) NBAXZ
   21  FORMAT(1H1,////,10X,* "" TROUBLE IN SUBROUTINE STORIJ. DIMENSION
     1 OF ELPA IS INADEQUATE. INCREASE NELPAZ TO *,I6,/,1H1)
       STOP
```

which can be represented in tabular form as:

ICOL (J)							
I \ J	1	2	3	4	5	6	...
0		1	3	6	10	15	...
1	1	2	4	7	11	16	
2		3	5	8	12	17	
3			6	9	13	18	
.				10	14	19	
.					15	20	
.						21	

In this line function:

NFUNC (I,J) — The index in an equivalent column array of terms (I,J) in an upper triangular matrix

NFUNC (0,J) — The location index of the diagonal term of the preceding column

LVMAX — Maximum number of variables (KUREL) per element actually encountered, or the length of the longest LVABL array

NELZ — End location of the element record in ELPA including the load column, or the element right-hand side (RHS)

MAXELT — $\frac{\text{NDOFPE} + (\text{NDOFPE}+1)}{2} + \text{NDOFPE}$, = maximum length of an element segment

"G" — Subvector of ELPA which contains the coefficients of assembled equations and the associated right-hand sides

MAXPA — Maximum number of active variables ever encountered at the front; MAXPA $\leq$ MVEND

NPAR — Location preceding the assembled right-hand sides

NPAZ — Last available location for the coefficients of the assembled right-hand sides

NBAXO — Location immediately preceding buffer area in ELPA reserved for equations using (e_s), etc.

NELPAZ — Effective dimension of ELPA

TABLE 1 - CORE-MEMORY REQUIREMENTS, A
SUMMARY OF CASE STUDIES

(Core Memory (CM_8) in Octal)

TWO-DIMENSIONAL CONTINUUM ELEMENT (8 NODE)						
Item	NELEM	NJ	NROW	MVEND	NELPAZ	CM_8*
A	10	50	3	24	1000	40,000
B	50	180	6	36	1200	42,000
THREE-DIMENSIONAL CONTINUUM ELEMENT (20 NODE)						
A	6	80	Single	60	3905	55,000
B	12	132	2	75	4870	77,000
C	15	148	3	90	6260	120,000
D	24	221	4	105	8330	115,000
E	48	409	6	135	11,480	131,000
F	64	531	8	165	16,090	145,000

*Add 5000_8 on CDC 6600 Scope 3.4.

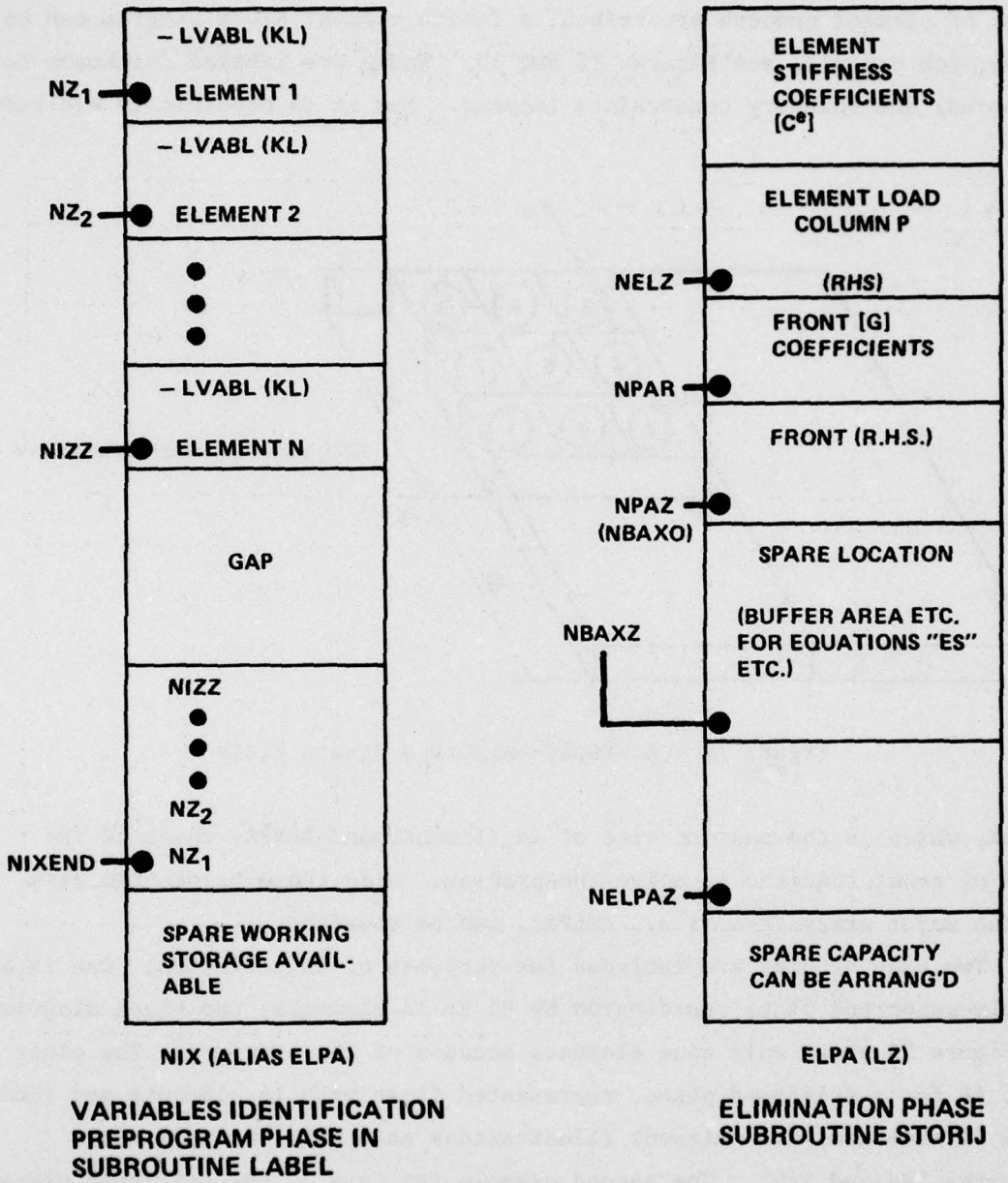


Figure 11 - Storage Allocations Used in the Working Column, Vector ELPA

Examples of Core-Size Calculation

After the element mesh for a given problem has been laid and the order of element numbers prescribed, a finite element block diagram can be drawn; for example, see Figures 12 and 13. Nodes are labeled (nickname for numbered) and boundary constraints imposed. Now it is possible to evaluate

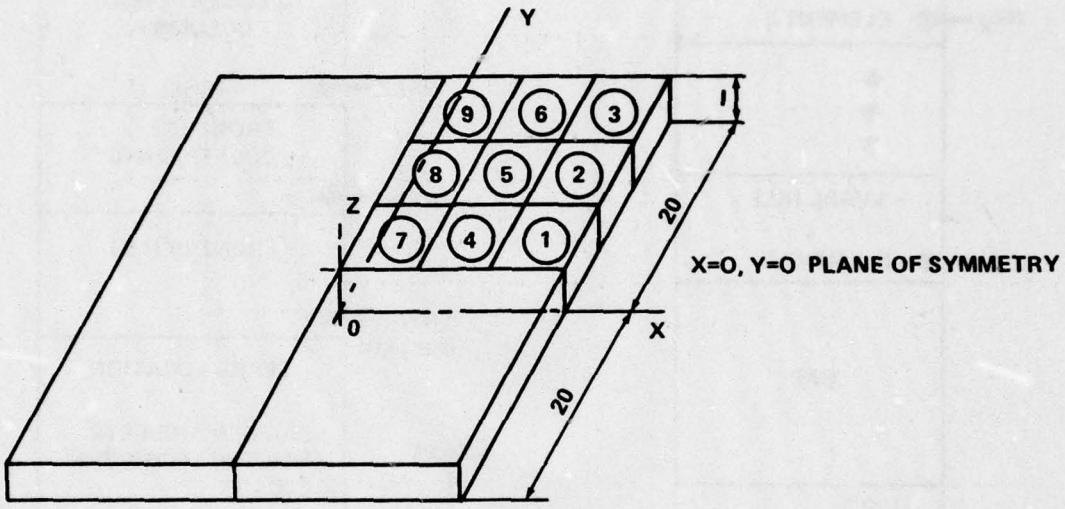


Figure 12 - A Simply-Supported Square Plate

LVMAX, which is the maximum size of an element, and MAXPA, which is the size of front required to solve the problem. With these values the size of the major array ELPA, i.e., NELPAZ, can be computed.

Two case studies are included for purposes of illustration. One is a simply-supported plate represented by 36 solid elements; the block diagram in Figure 13 shows only nine elements because of the symmetry. The other case is for a stiffened plate, represented first with 16 elements and then with 24 elements; the relevent illustrations show only one-half spans (Figures 14a and 14b). The second example for Case 2, the stiffened plate shown in Figure 14b, also illustrates a special feature of frontal solution, i.e., element numbers are critical whereas nodal numbers are immaterial.

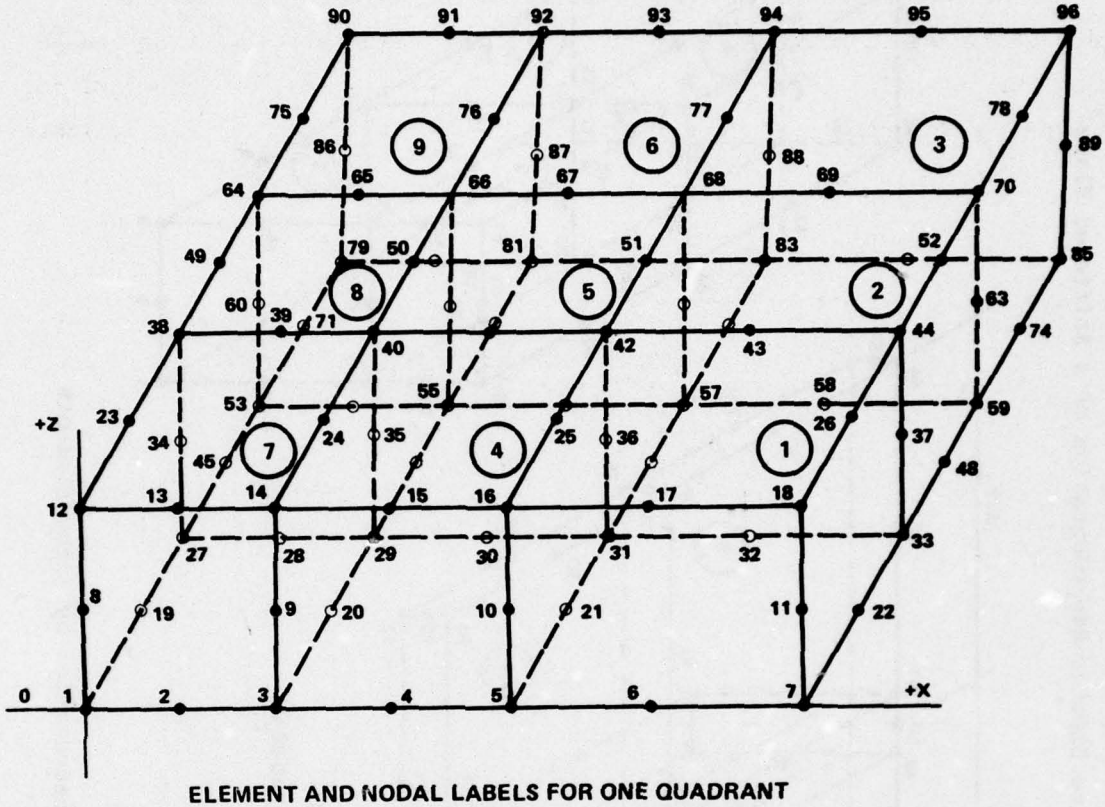


Figure 13 - Finite Element Representation of a Plate Quadrant

Figure 14 - Two Schemes for Finite Element Representation of a Stiffened Plate

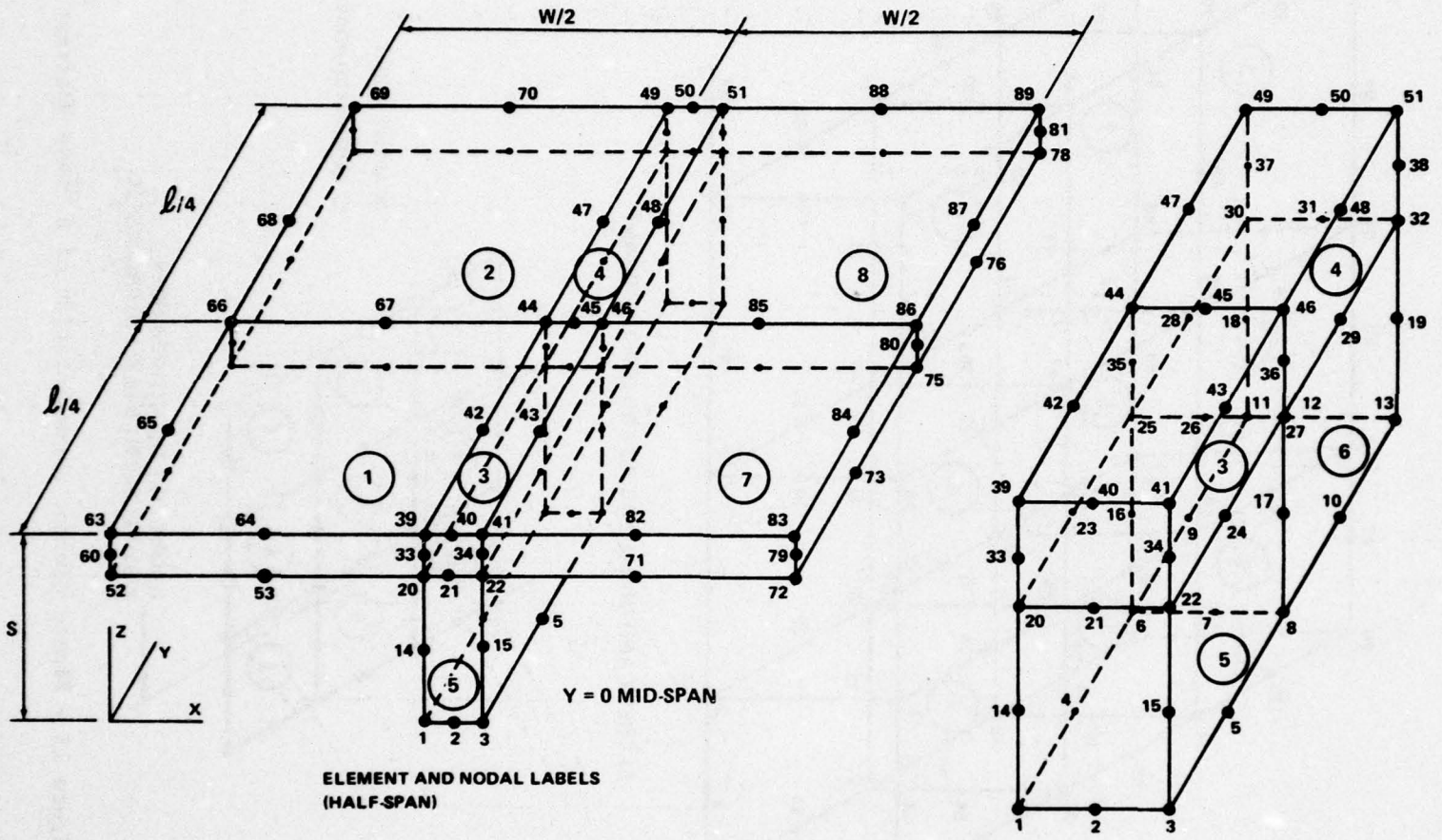
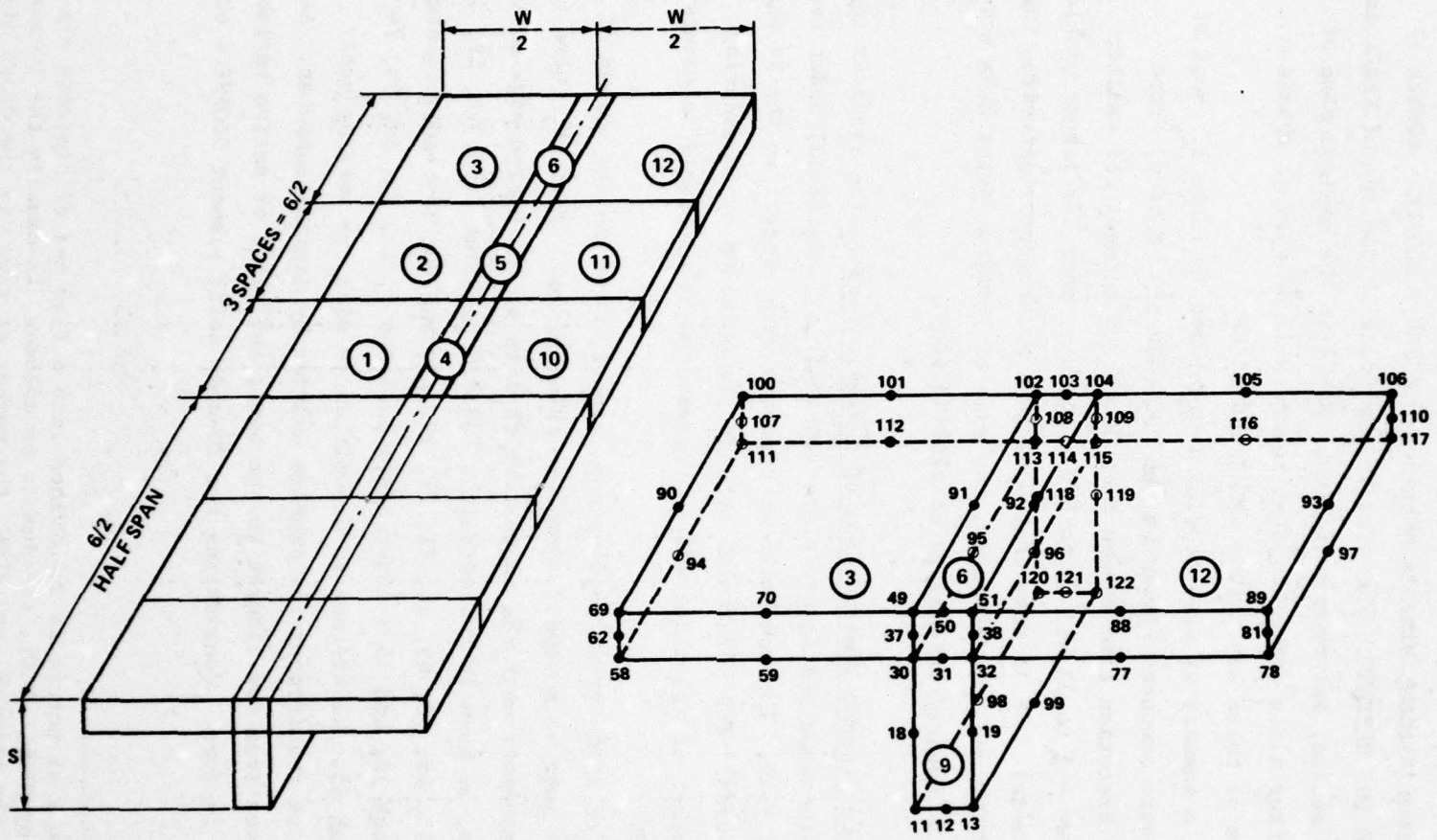


Figure 14a - Representation by Eight Elements

Figure 14 (Continued)



ELEMENT AND NODAL LABELS OF A STIFFENED PLATE

Figure 14b Representation by Twelve Elements

The calculation is fairly routine. Following each element in succession, the current element size is obtained: KUREL = NDOFPE - NUMBER OF CONSTRAINTS ON THE ELEMENT. The size of the front is the sum of KUREL and the number of active, existing variables. MAXPA is the maximum size of the front. After a few exercises, the reader will be able to obtain a quick estimate of these key values LVMAX and MAXPA.

Finally, a summary of case studies is included in Table 1. Most of the problems were executed either on the CDC 6400 or CDC 6600, Scope Version 3.3. Execution time ranged from 30 to 100 seconds for smaller problems (NROW* = 1 to 3) and up to 500 seconds or more for larger problems (NROW = 6 or more). It is seen that NROW is a prime factor affecting the front width MVEND and, consequently, the size of working vector ELPA and core memory CM required to analyze an elastic body.

Case 1: Finite Element Idealization of a Simple Plate. The analysis involves a simply-supported square plate (40×40×1 in., Figure 12) under transverse bending load. The condition of symmetry with respect to the X- and Y-axes is assumed. The structural response is obtained by considering only one quadrant of the plate and is represented by nine solid elements; see Figure 13.

Because of symmetry in loading and support, nodes on the planes of symmetry will have no movement normal to these planes. Further, there is no vertical movement over the support which lies at the bottom edge of the plate. Hence, we have 18 X-constraints "JREST" on Nodes 1, 8, 12, 19, 23, 27, 34, 38, 45, 49, 53, 60, 64, 71, 75, 79, 86, and 90; 18 Y-constraints on Nodes 1 through 18; and 13 Z-constraints on Nodes 7, 22, 33, 48, 59, 74, and 79 through 85. We are now in a position to size the working array (ELPA) and core requirement for problem solving by digital computer. As the front moves from one element to the next, the range of active variables is evaluated in turn. Considering the 20-node solid element NDOFPE = 60, we have:

---

*Suppose a structure is subdivided into a flow net of discrete elements, NCOL by NROW. NCOL, the number of columns, is usually the longer dimension of the network, and NROW, the number of rows, is the cross dimension of the same network.

Element No.	Count of "JREST"	"KUREL" (LVMAX)	Existing Active Variable	Number of Positions in Front (LDES)
1	$0+8_y+3_z$	49	0	49
2	$0+0+3_z$	57	+12	69
3	5_z	55	+12+15	82
4	8_y	52	+15+14	81
5	0	60	+12+14	86*
6	3_z	57	+12+15	84
7	8_x+8_y	44	+15+14	73
8	8_x	52	14	66
9	8_x+3_z	49	--	49

*The maximum front width occurs on processing Element 5, and maximum "LDES," i.e., MAXPA = 86. Now

$$\text{NELZ} = \frac{61\ (60)}{2} + 60 = 1890$$

$$+ \frac{87\ (86)}{2} = \underline{+)\ 3741}$$

$$5631\ (=\text{NPAR})$$

$$2 \times 86 = 172$$

$$1 + 86 + 4 = \underline{\quad 91}$$

$$\text{Total} \qquad 5894\ (=\text{NBAXZ})$$

$$< \text{NELPAZ} = 6260$$

Case 2: Idealization of a Plate Beam (or Stiffened Plate). Figure 14 illustrates two schemes for nodal labeling. The first node number may begin at any convenient location, for example, at the bottom edge of the midplane section, and then follow the positive directions of the global X-, Y-, and Z-axes. In Figure 14a, the labeling treads through Elements

6, 3, 4, 1, 2, 7 and ends with the last node 89 on the top side of Element 8. Element labels are also shown on the same figure. Because the optimal element numbering to obtain a minimum front width is not always obvious, the type of preliminary estimate of the range of active variables given here can be helpful in setting up an effective scheme.

First, all constraints, structural supports, and boundary constraints must be delineated. Vertical end supports ($Y=\pm\ell/2$) are provided along the bottom edge of the plate, the stem, and also on both side nodes of the stem. The centerplane ($X=b/2$) is held against lateral movement and the midplane ($Y=0$) is a plane of symmetry. This results in 12 JREST(Z), 6 JREST(X), and 23 JREST(Y). From evaluation of Figure 14a we have:

Element Number	JREST	KUREL (LVMAX)	Existing Active "X"	Front Width (MAXPA)
1	8_y	52	0	52
2	3_z	57*	12	69
3	2_x+8_y	50	14	64
4	2_x+3_z	55	11+7	73
5	2_x+8_y	50	13+14+6	83
6	2_x+8_z	50	12+9	71
7	8_y	52	14	66
8	3_z	57		57

*Hence NELZ $= \frac{57\ (58)}{2} + 57 =$ 1710

$+ \frac{84\ (83)}{2} =$ +) 3486

5196 (=NPAR)

$83 \times 2 =$ 166

$83 + 4 + 1 =$ 88

Total 5450 (=NBAXZ)

< NELPAZ = 7660

The core requirement is 105,000 octal where using CDC 6400 (Scope Version 3.3).

The stiffened plate is reanalyzed in Figure 14b by adding four more elements to the existing model. The existing nodal labels and element incidences (INCEL) referring to these nodes are reusable - a time saver for complex models. A set of 33 new nodes are augmented, Figure 14b. The order of new nodal labels is immaterial with respect to the existing scheme of nodal numbers.* In the current example, elements are renamed so that a smaller front width can be achieved. In fact, elements 10, 11, and 12 can be deleted owing to symmetry of the beam with respect to its centerplane $x = w/2$.

The maximum front occurs at the processing Element 8 of Figure 14b. In this case, KUREL = 60 and MAXPA = 60 + 12 + 15 + (11+9) = 107. The summation in the last expression includes terms 12, 15, and 20 representing the number of nodal variables introduced by Elements 4, 5, and 6, respectively, and which remains active. Hence

$$\text{NELZ} = \frac{60\ (61)}{2} + 60 = 1890$$

$$+ \frac{107 + 1}{2}\ (107) = \underline{+)5778}$$

$$7668\ (=\text{NPAR})$$

$$107 \times 2 = 214$$

$$107 + 4 + 1 = \underline{112}$$

$$\text{Total} \quad 7994\ (=\text{NBAXZ})$$

$$< \text{NELPAZ} = 8330$$

The core requirement is 111,500 octal when using CDC 6600 (Scope Version 3.3).

---

*Here the program user is free to choose any convenient nodal label. This is in contrast to many finite element programs where restrictive node-numbering rules must be followed in order to conserve the bandwidth of a problem, resulting in a substantial revision of nodal numbers.

ARRAY SPECIFICATION

The dimensions of the standard array listed in Table 2 are for a certain number of rows (NROW): NROW = 6 in the case of two-dimensional continuum elements and NROW = 8 for three-dimensional continuum elements. For a specific class of problems, the dimensions of the system array, such as NJ* and NEL* may be adjusted accordingly. The values of MVEND and NELPAZ can be computed as described in the preceding section or obtained by inspection. The dimension of the long vector ELPA is a function of NEL, NJ, and NROW among which NROW is one dominant factor.

INSTRUCTIONS FOR DATA INPUT

INPUT ITEMS AND OPTIONS

Input items are read by the main program PBLADE and are listed in Table 3. Input cards consist of a set of 10 items: Cards 1 through 7 are single-card items, whereas the remainders are multicard items that vary with the number of elements or nodes employed to represent a structure. For input definition and problem specification see the sections entitled, respectively, NOTATION and PROBLEM FORMULATION.

Certain options, such as generation of intermediate nodes and/or equivalent nodal loads, are available. These options are governed by the logic parameters:

LNEL	Number of the last element which is loaded by a surface pressure, applicable when NF = 0, or 1.
LPB	Nodal number at which printout of PLOAD-array begins, for NF = 0, or 1.
LPT	Nodal number at which printout of PLOAD-array terminates, for NF = 0, or 1.
NBLA = 0	Stresses with reference to global coordinate axes are computed, element by element, at their integration points (which were used to calculate element stiffness matrices).
NBLA = 1	Additional stresses (σ_r, σ_θ, $\tau_{r\theta}$) are computed on (element) body surfaces in cylindrical coordinates. Principal surface stresses are also given.

*See NOTATION for definition.

TABLE 2 - DIMENSIONS OF STANDARD ARRAY

Element Array	Two-Dimensional Element	Three-Dimensional Element
X(ND,NNPE)	X(2,8)	X(3,20)
P(NDOFPE)*	P(16)	P(60)
STFNS(NDOFPE,NDOFPE)	C(16,16)	C(60,60)
EL(MAXELT)**	EL(152)	EL(1890)
JDIS(NDOFPE)	JDIS(16)	JDIS(60)
LDEST(NDOFPE)	LDEST(16)	LDEST(60)
LVABL(NDOFPE)	LVABL(16)	LVABL(60)
System Array		
COORD(ND,NJ)	COORD(2,175)	COORD(3,531)
PLOAD(ID,JOINT)	PLOAD(2,175)	PLOAD(3,531)
DISP(ID,JOINT)	DISP(2,175)	DISP(3,531)
JDISP(ID,JOINT)	JDISP(2,175)	JDISP(3,531)
INCEL(NNPE,NEL)	INCEL(8,50)	INCEL(20,64)
MVABL(MVEND)	NVABL(66)	MVABL(165)
ELPA(NELPAZ)	ELPA(3000)	ELPA(16085)
NIX(NIXEND)	NIX(1600)	NIX(6000)

*NDOFPE = ND × NNPE

**MAXELT = NFUNC(O,NDOFPE+1) + NDOFPE

TABLE 3 - INPUT ITEMS FOR PBLADE COMPUTER PROGRAM

Card Item	Input Data	Data Format*
1	Title: Problem, Scope, Date, etc.	55-col,H-Format
2	Blank	
3	E, PRATIO, T, DEN	6F10.5
4	NNPE, NDOFPN, LVEND, MVEND, NELPAZ, NIXEND	6I5
5	NELEMZ, NJ, JREST, NSVJ, INCID, NBLA	6I5
6	NF, NLOADS, LPT, LNEL, NPT, NPZ	6I5
7	NVABZ, LCUREQ, LVMAX, MAXNIC, MAXPA, NIZZ	6I5
8	LPB, NPUNCH	6I5
8a	PRES (L), L = 1, NELEMZ, for NF = 0 only	6F10.5
9	NELEM, INCEL (J,NELEM), J = 1, NNPE, NELEM = 1, NELEMZ	21I3
10	JOINT (L), COORD (J,JOINT(L)), J = 1, NDOFPN L = 1, NJS	I5,5X,3F10.3
11	JOINT (I), IDIREC I = L, JREST	2I5
11a	JOINT (L), PLOAD (ID, JOINT(L)), for NF = 2 only, ID = 1, NDOFPN; L = 1, NLOADS	I5,5X,3F10.3

*Format: I is an integer and F is a real decimal number.

NF = 0 — A distributed pressure is integrated over the contiguous element surface of an arbitrary body. (The value of average pressure over each individual element may be prescribed as input. See Card Item 8a, Table 3.) The equivalent nodal forces which are the algebraic sums of loads from all contributing elements attached to these nodes will be tabulated and printed. The complete set of equivalent nodal loads will be used as input in the equilibrium equation, Equation (3).

NF = 1 — The pressure load is input by an assigned pressure distribution for each element prescribed in data set PRES.

NF = 2 — Load input is read from a set of NLOADS, one for each loaded joint (Card Item 11a). A joint load is designated for each node where an external force is explicitly in effect (or indirectly through contributions of connecting elements). The joint load is defined by its three components along the global coordinate axes.

NSVJ = 0 — Regular nodal coordinates (X,Y,Z) are read for every joint required to define the geometry of the individual elements.

NSVJ > 0 — Subroutine GEOORD is called to generate midsurface nodal coordinates of a shell-like structure. These coordinates in conjunction with surface coordinates (which were read as INPUT) form the complete geometric input. NSVJ numerically equals the number of nodes on the midsurface.

INCID* = 1 — See Card Item 9. Standard incidence format (21I3) is used to insert nodal sequence of the 20-node hexahedron.

INCID* = 2 — Incidence format (13I5) is used. Local coordinate axis - ξ will be rotated by one quadrant of the surface (ζ=0).

Other terms defined by the main program include:

INTGER = NELEMZ

NTIREX = 1

NJS = NJ - NSVJ

T, DEN = CONSTANTS

EPSLON = 0.5E - 10

---

*INCID = 0 indicates that options are to be defined. (The existing option accepts the 16-node three-dimensional shell element input format which is to be converted to a standard 20-node hexahedron.)

$$C_1 = \frac{\nu E}{(1+\nu)\ (1-2\nu)}$$

$$C_2 = \frac{E}{2(1+\nu)}$$

} Lame' Constants

$$DK = \frac{E}{(1+\nu)\ (1-2\nu)}$$

SAMPLE PROBLEMS

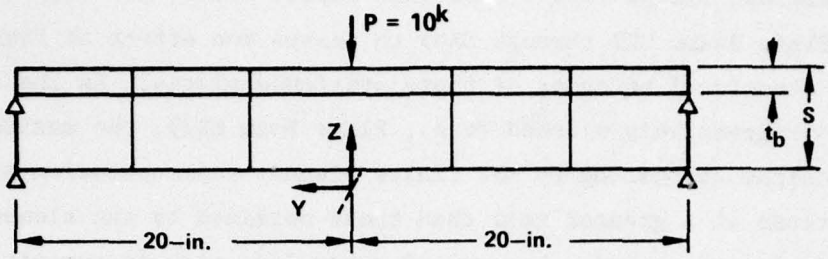
Stresses in Plate Beams, Problem 1

A plate beam is an integral structural element essentially consisting of a plate with a beam stiffener placed underneath it. Plate beams (Figure 15) have the shape of a T-beam and are often treated as such; elementary beam theory is applicable. When the beam has an extended flange or the stiffened plating has widely spaced stiffeners, the effective width concept has been found useful in design considerations. It has been pointed out earlier* that because the flexural stress distribution is not (as assumed) uniform across the width of plating, stiffness as calculated on the basis of the conventional design rule is often overestimated. Where the fatigue and fracture mechanism becomes a factor in design criteria, it is particularly important to have an accurate evaluation of the state of stress. In such cases, the plate beam should be treated like a three-dimensional elastic body, as described below.

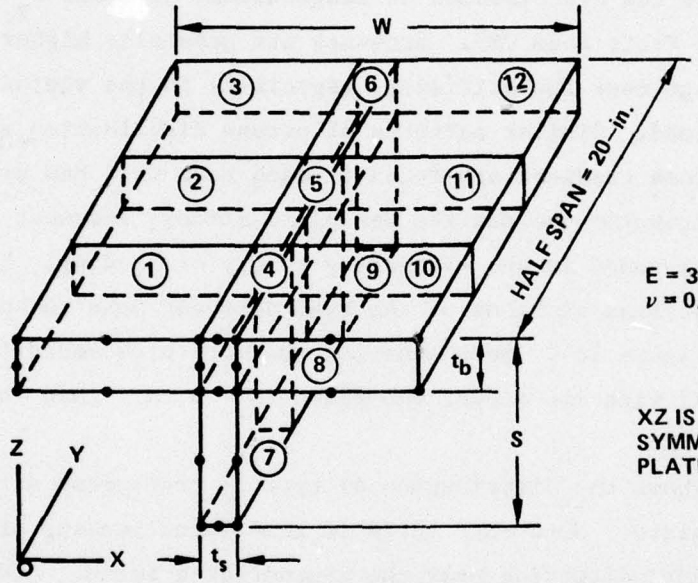
Figure 15 indicates a general scheme used to analyze a laterally loaded plate beam. The plate beam is designated as a simply supported span carrying a center load P. Because of symmetry, a half-span of the beam is idealized and twelve 90-node hexahedron elements are used to represent it. At each end, vertical supports are provided for both flanges and web. Input data are prepared according to instructions given for data input; these are listed in Appendix B.

---

*See Reference 11, page 91.



A SIMPLY SUPPORTED PLATE BEAM



E = 30,000 ksi
ν = 0.30

XZ IS THE PLANE OF SYMMETRY OF PLATE BEAM

A FINITE ELEMENT REPRESENTATION

PLATE BEAM MARK	BEAM PARAMETER				CENTER DEFLECTION (in.)	
	w	t_b	s	t_s	FINITE ELEMENT	BEAM THEORY
CX1	0.6	0.6	0.6	0.6	0.0425	0.0412
CX2	6.0	0.6	0.6	0.6	0.0214	0.0191
CX3	12.0	0.6	0.6	0.6	0.0185	0.0158
CX4	1.8	0.6	3.0	0.6	0.213	0.210
CX5	6.0	0.6	3.0	0.6	0.147	0.142
CX6	12.0	0.6	3.0	0.6	0.129	0.123

PARAMETERS OF BEAM CROSS SECTION AND COMPUTED DEFLECTIONS

Figure 15 - Plate Beam Sample Problem

Five different flange versus stiffener aspect ratios are used as parameters (Plate Beams CX2 through CX6) to assess the effect of flange width on the structural behavior of these stiffened plates. As the flange of a beam is progressively widened (e.g., Plate Beam CX3), the maximum center deflections calculated by the finite element representation are found to increase at a greater rate than those obtained by the elementary beam formula. In other words, because of shear lag, each incremental flange material is engaged in a lesser capacity than assumed by classical theory.

Figure 16 depicts the distribution of longitudinal stresses σ_y on a transverse section of Plate Beam CX3. Stresses are generally higher in the parts of the flange near the stiffener, especially in the vicinity of the concentrated load. Similar patterns of stress distribution with varying degree of stress gradient are found in each beam that has projected flanges. Stress distribution across the web plate differs somewhat from the linear variation assumed in the elementary theory of bending. The neutral axis at these cross sections of the beam does not pass through the geometric centroid (Figure 16). Such behavior was more pronounced for a plate beam such as CX3 with its deeper web plate (S = 6 in.) than for the others, for example, CX4.

Figure 16 also shows the distribution of typical transverse stresses σ_x along the flange plate. Note that there is some local bending of the flange in the transverse direction near the concentrated force. Additional stress data of interest are given in Table 4. These stresses are obtained at the Gaussian integration points (used to form the stiffness matrix of the isoparametric element). Stresses computed at such points have been shown to be generally of high accuracy.

A Supercavitating Blade, Problem 2

Figure 17 illustrates another application of the general three-dimensional element. An 18-in. diameter, stainless steel model of supercavitating Propeller DTNSRDC P-3604 with wedge-shaped blade sections was selected. The propeller is not raked or skewed and is composed of wide blades with an aspect ratio (based on mean chord length) of about 1. The

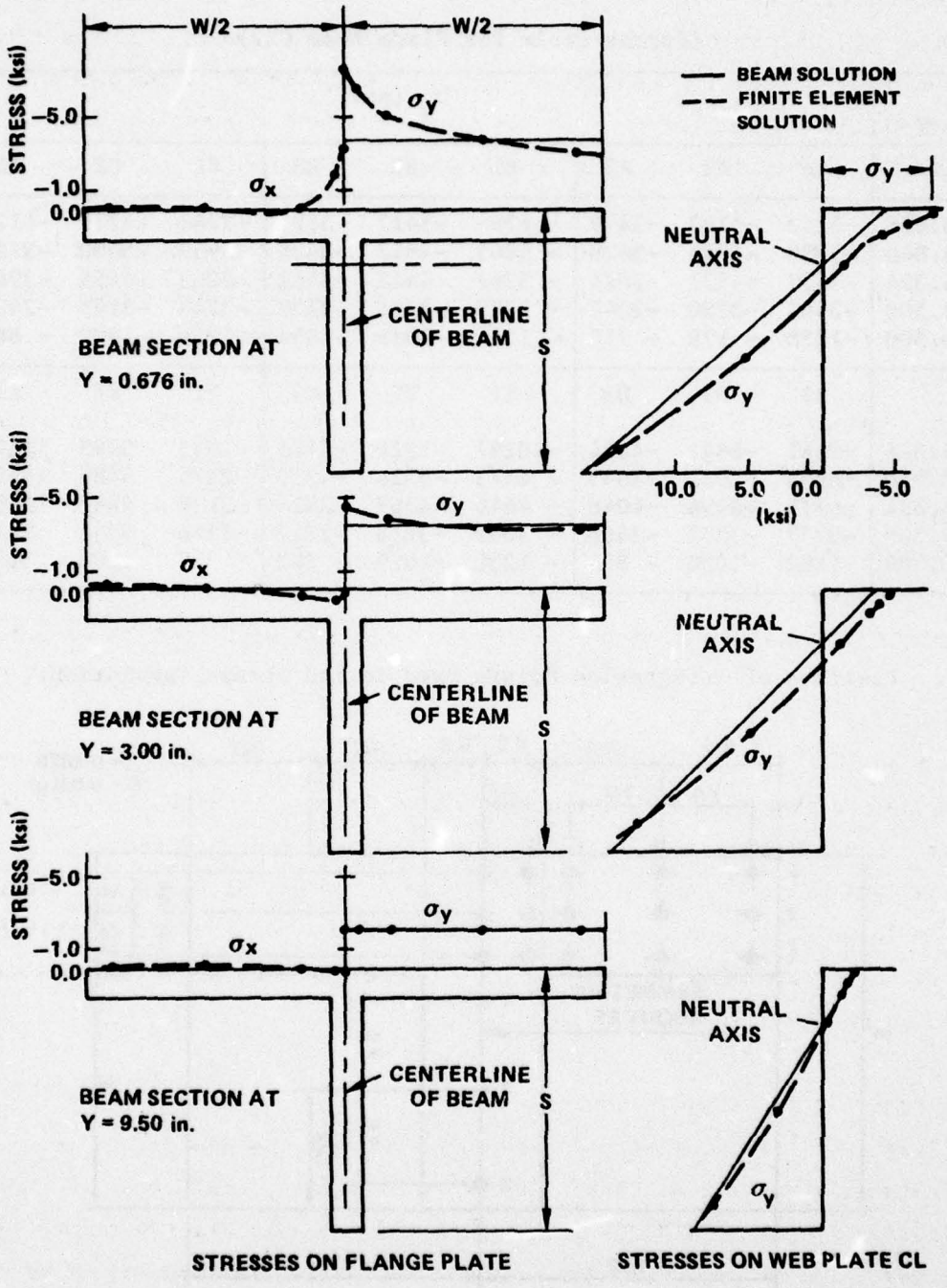


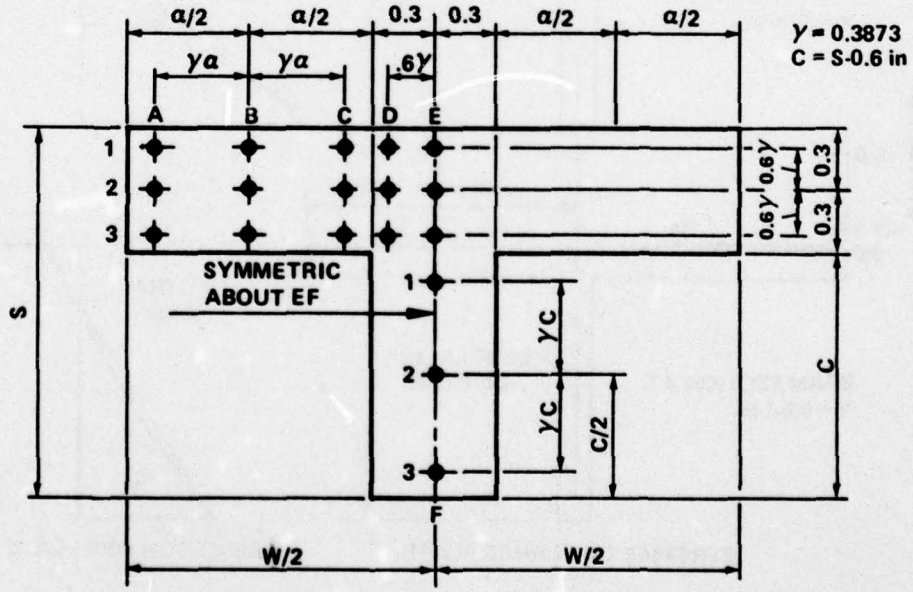
Figure 16 - Distribution of Normal Stresses in a Plate Beam for Sample Problem 1

TABLE 4 - LONGITUDINAL STRESSES (σ_y) IN A PLATE BEAM

(Stress Table for Plate Beam CX2)

Y	σ_y (psi)								
(in.)	A1	A2	A3	B1	B2	B3	C1	C2	C3
0.676	-6252	-5103	-3479	- 6795	-5417	-3525	-7768	-6215	-4127
3.000	-5989	-4945	-3529	- 5845	-4817	-3450	-5959	-5000	-3755
5.324	-5557	-4471	-3074	- 5267	-4442	-3411	-5063	-4555	-3961
9.500	-3683	-3180	-2547	- 3725	-3201	-2537	-3748	-3199	-2498
16.500	-1158	- 979	- 752	- 1214	-1039	- 814	-1255	-1082	- 863
	D1	D2	D3	E1	E2	E3	F1	F2	F3
0.676	-9161	-6841	-4624	-10247	-7926	-5705	-2533	5995	18193
3.000	-6093	-5051	-4045	- 6575	-5528	-4519	-2225	5486	16420
5.324	-4773	-4496	-4048	- 4646	-4367	-3916	-2135	4687	14265
9.500	-3757	-3232	-2466	- 3639	-3113	-2317	-1126	3541	9634
16.500	-1200	-1050	- 854	- 1230	-1079	- 882	- 387	1203	3182

Position of Integration Points Used in the Stress Tabulation:



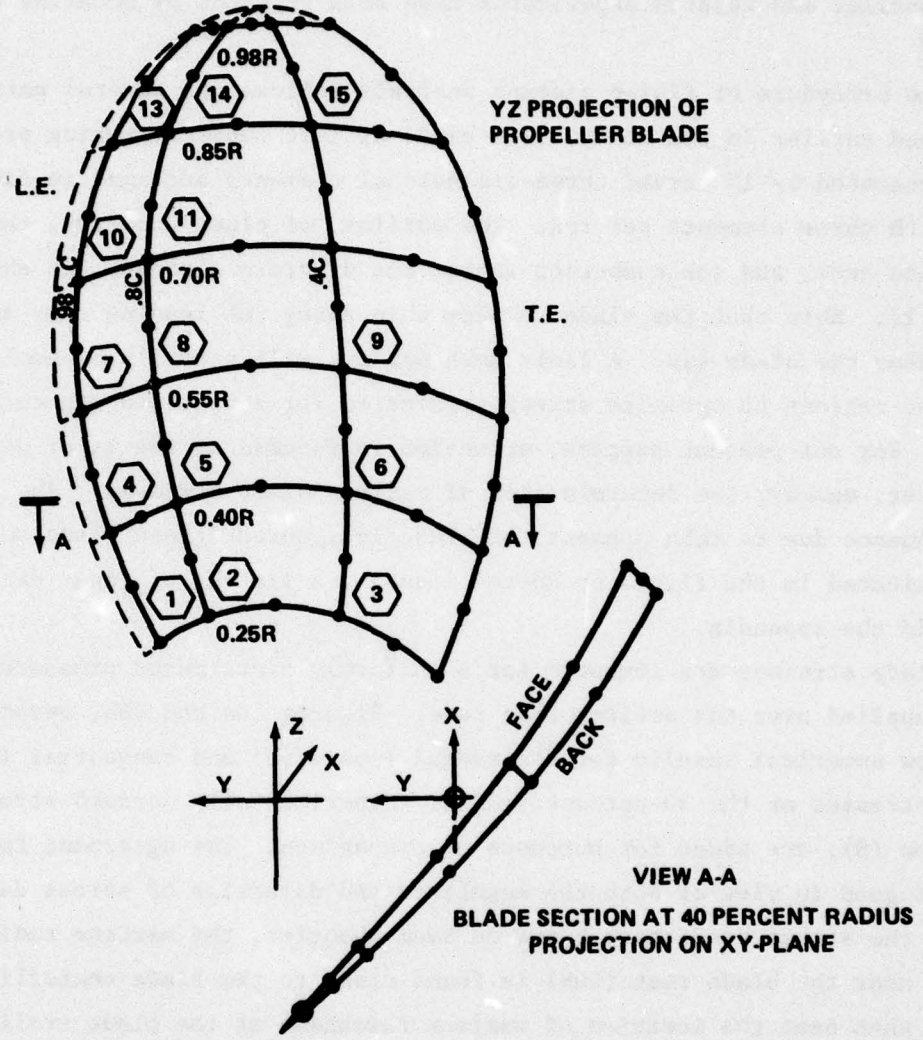


Figure 17 - Curved Finite Element Representation of a Supercavitating Propeller DTNSRDC P-3604

maximum chord length occurs at the hub and ramains about the same to 60 percent of the tip radius. The blade is very highly pitched. Its maximum section thickness occurs at or near the trailing edge. Other details of the propeller and related experiments have been reported by McCarthy and Brock.[8]

The procedure of finite element analysis follows the general pattern described earlier in the study. The blade of this supercavitating propeller is represented by 15 curved three-dimensional elements arranged in five rows with three elements per row. The outlines of blade geometry, the reference axes, and the numbering scheme for discrete elements are shown in Figure 17. Note that the blade is very thin along the leading edge and again near the blade tip. A finer mesh pattern will normally be employed at these regions to optimize stress prediction for these thin segments of blade. For our present purpose, attention is focused on the major design parameter, namely, the determination of maximum blade stresses. The limited influence due to thin segments of blade is ignored (these blade segments are indicated in the figure by dotted lines). A listing of input data is given in the appendix.

Blade stresses are computed for a uniformly distributed pressure of 1 psi applied over the entire blade face. Figures 18a and 18b, respectively, show numerical results for the radial (spanwise) and tangential (chordwise) stresses at the 30-percent radius. Experimentally derived stresses, Equation (8), are added for purposes of comparison. The agreement is considered good in view of both the magnitude and direction of stress data. Unlike the stress prediction based on beam theories, the maximum radial stress near the blade root (hub) is found close to the blade centerline rather than near the location of maximum thickness at the blade trailing edge; for large radii, the locations of maximum spanwise stresses tend to shift toward the blade trailing edge. Also note that spanwise stresses are predominant near the blade root and that chordwise stress components become more prominent near the blade tip.

Figure 18 - Supercavitating Blade Stresses for a Uniformly Distributed 1-Pound per Square Inch Pressure Over the Entire Blade Face

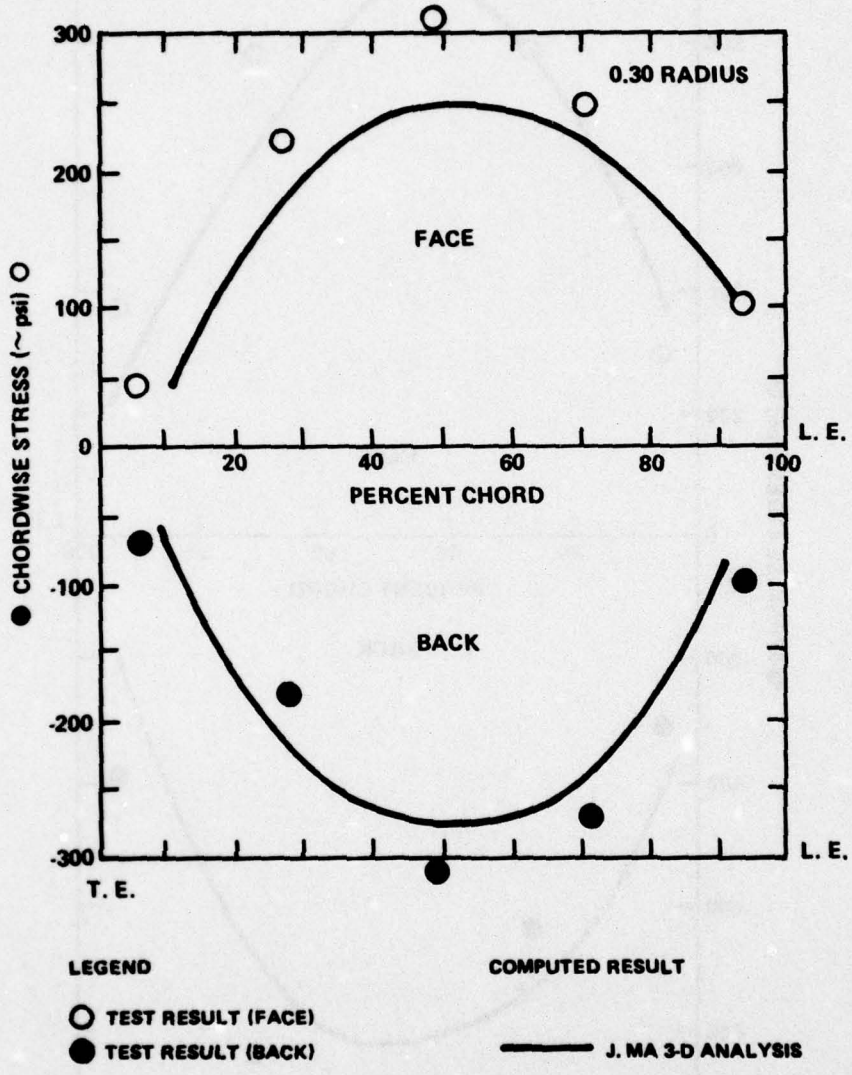


Figure 18a - Tangential (Chordwise) Stresses

Figure 18 (Continued)

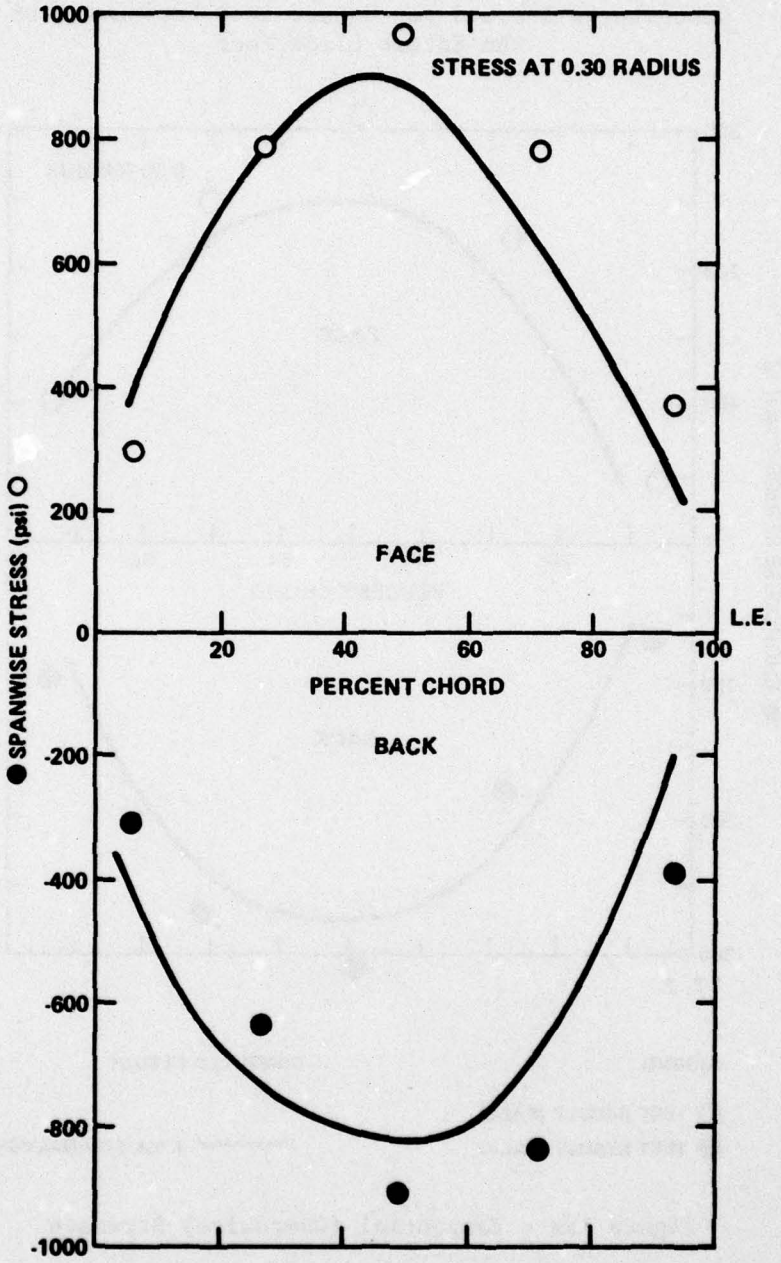


Figure 18b - Radial (Spanwise) Stresses

The Solid Foil of TAP-1, Problem 3

The next sample problem concerns validation of the feasibility of a preliminary foil design[21] for a desirable hydrodynamic cross section of a high-speed hydrofoil test craft. It is important to have a reliable estimate of stresses in the critical areas of the foil-strut system to ensure adequate strength. The areas of concern are the stresses at the thin leading edge of the foil and near the foil root interface with the strut (support).

Figure 19 is a pictorial view of a supercavitating foil model made of HY-130 steel (or 17-4 PH stainless steel). This solid supercavitating foil was analyzed by using curved three-dimensional finite elements. The TAP-1 foil has a typical wedge-shaped chord section with maximum thickness at the wetted trailing edge. A varying chord length tapers linearly toward the tip as shown in the foil planform (Figure 20). At its centerline, the foil receives support from a generously proportioned strut. The foil and its support are represented by 43 curved solid elements. In an alternative design which includes the foil annex, 49 solid elements are employed. A fine element mesh is adopted at the leading edge and also along the fillet area near the foil-strut interface to provide a clear picture of stress distribution at those structurally critical regions.

The nodal coordinates for the top and bottom faces of the foil are derived from the design layout for the TAP-1 hydrofoil,* supplied by the DTNSRDC Design Engineering Division, Code 294. The XY-plane of the global coordinate system corresponds to the reference foil chord plane** with center of the coordinate system located at the 0.7 chord point on the foil centerline (see Figure 20). A total of 421-XYZ coordinate points is used in the finite element mesh to describe the foil including annex.

Load Condition 1, which corresponds to the maximum lift at a speed of 80 knots, has been used because it results in maximum foil bending moments

---

*Twist of the foil was removed to simplify computing structural load and stress.

**A plane which passes through both the leading edge and the trailing edge.

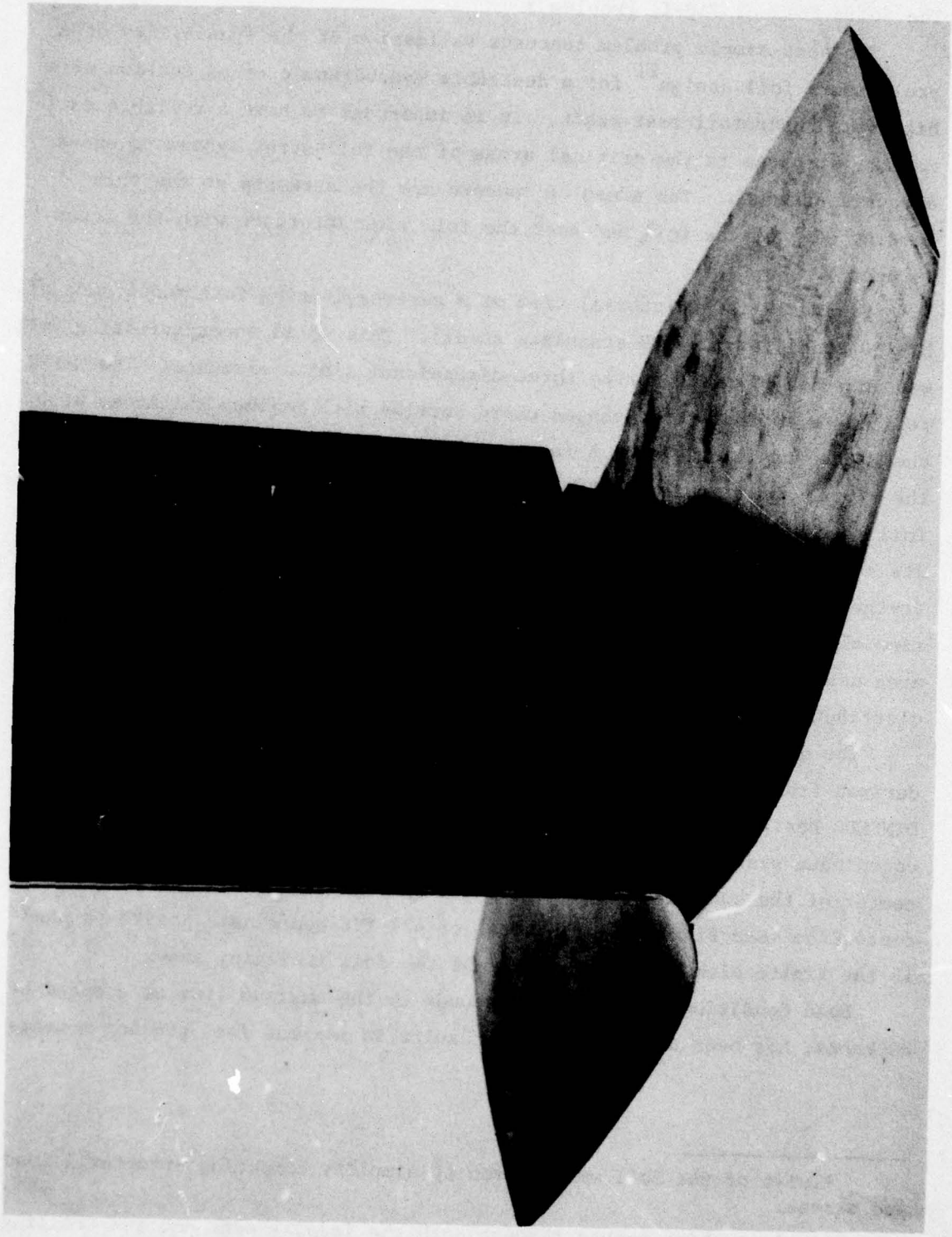


Figure 19 - Pictorial View of a Supercavitating Foil Model

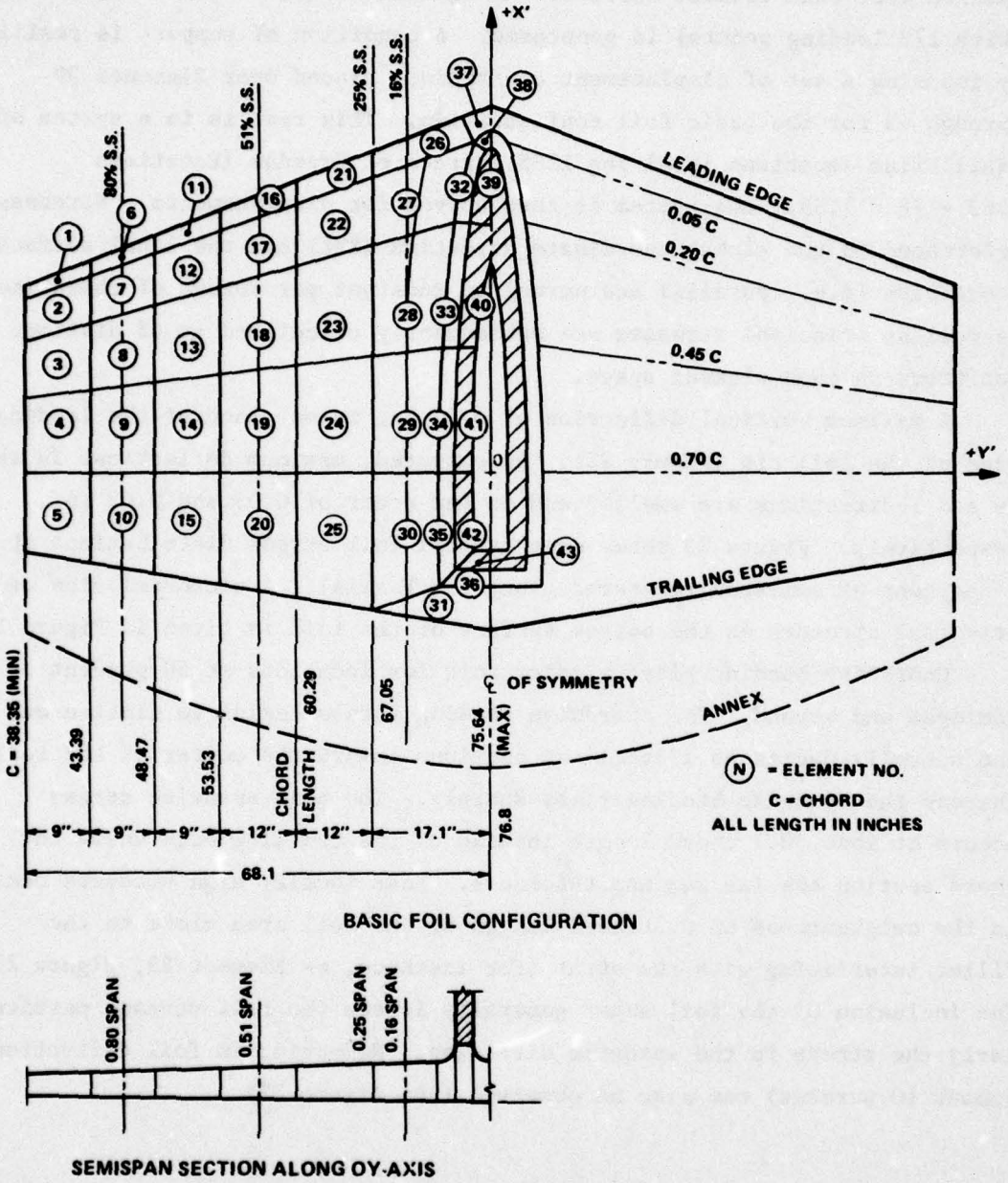


Figure 20 - Planform of TAP-1 Foil and Strut

as well as maximum loading on the foil leading edge. The pressure distribution of Figure 21 is inserted into the program in a linear piecewise fashion over each element surface. An equivalent set of nodal load vectors (with 178 loading points) is generated. A condition of support is realized by imposing a set of displacement constraints placed over Elements 39 through 43 for the basic foil configuration. This results in a system of equilibrium equations involving 1185 degrees of freedom (Equations 1263 - 78 = 1185); the system is then solved for displacements. Stresses referenced to the global coordinate direction (XYZ) and the local surface coordinate (i.e., parallel and normal to constant percentage of chordlines) as well as principal stresses are subsequently calculated at 45 distinct positions in each element space.

A maximum vertical deflection of 0.57 in. takes place at the leading edge of the foil tip (Figure 22). As expected, maximum deflections in the X- and Y-directions are smaller and on the order of 0.03 and 0.02 in., respectively. Figure 23 shows some typical foil stress distributions at 25 percent of semispan (measured along the Y-axis). A pictorial view of principal stresses on the bottom surface of the foil is given in Figure 24.

Chordwise bending plays a major role for locations at 50 percent of semispan and beyond. The chordwise bending stress begins to flatten out and actually decreases slightly at sections toward the center of the foil whereas the spanwise bending rises sharply. The peak spanwise stress occurs at about 0.7 chord length instead of the trailing edge where the chord section has its maximum thickness. Some locally high stresses occur in the neighborhood of 0.3 chord length of the foil area close to the fillet interfacing with the strut (for instance, at Element 28, Figure 20). The inclusion of the foil annex generally lowers the foil stress, particularly the stress in the spanwise direction. Reduction in foil deflection (about 10 percent) can also be observed (see Figure 22).

Note: Computation time on the CDC 6600 for the basic foil configuration is a little over 5 minutes (CPA-320 Sec).

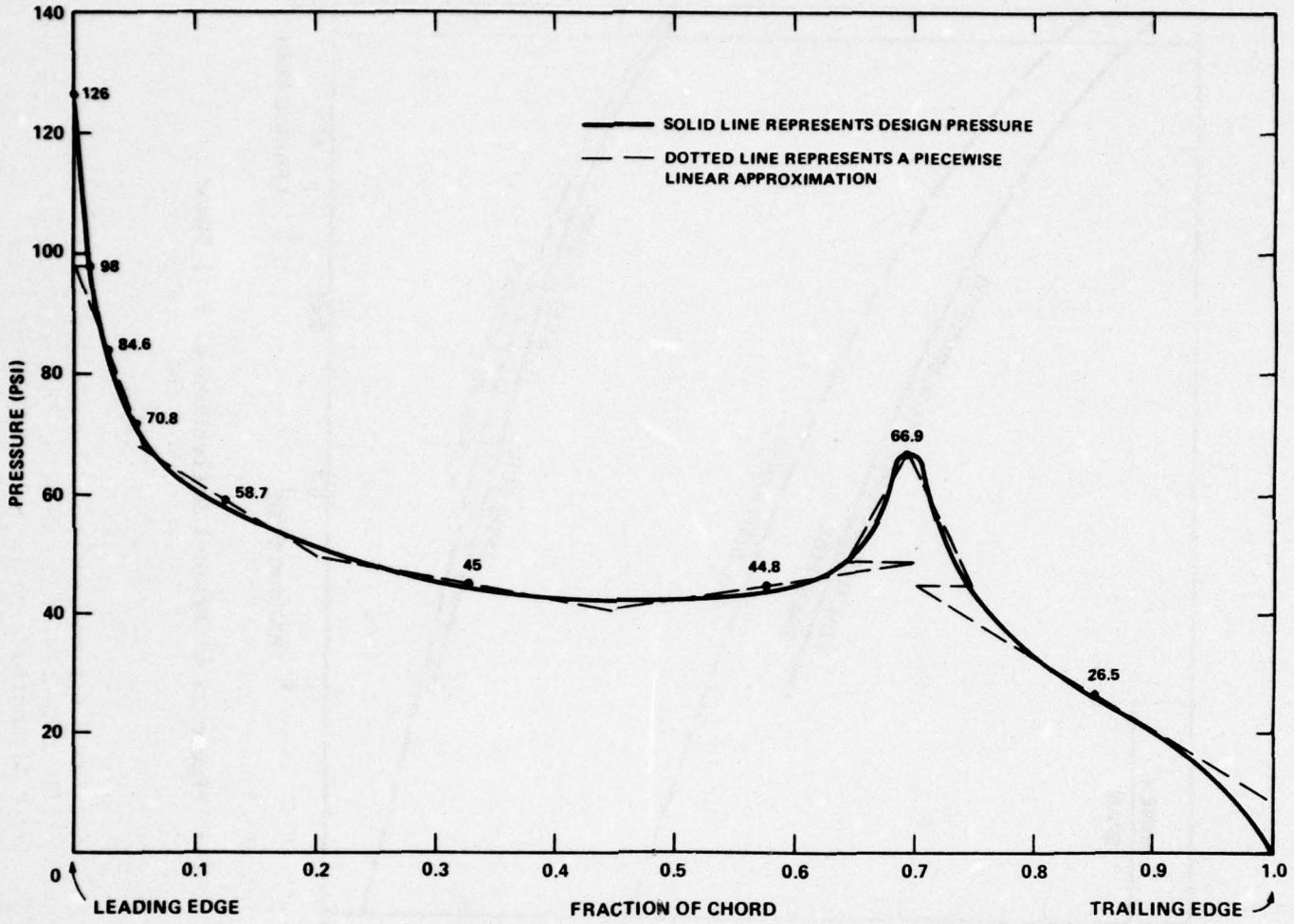


Figure 21 - Typical Pressure Distribution across a Chord Section of TAP-1

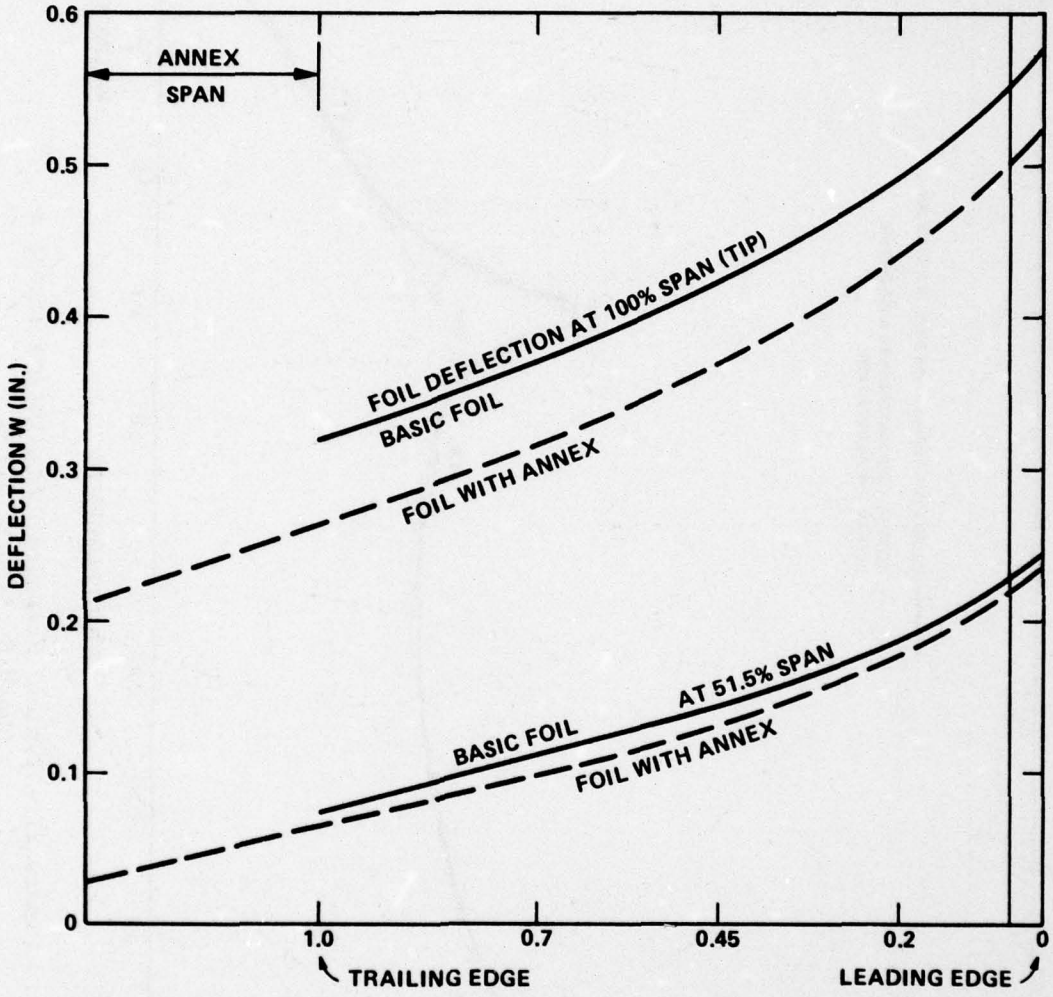


Figure 22 - Vertical Deflection of Foil Spans

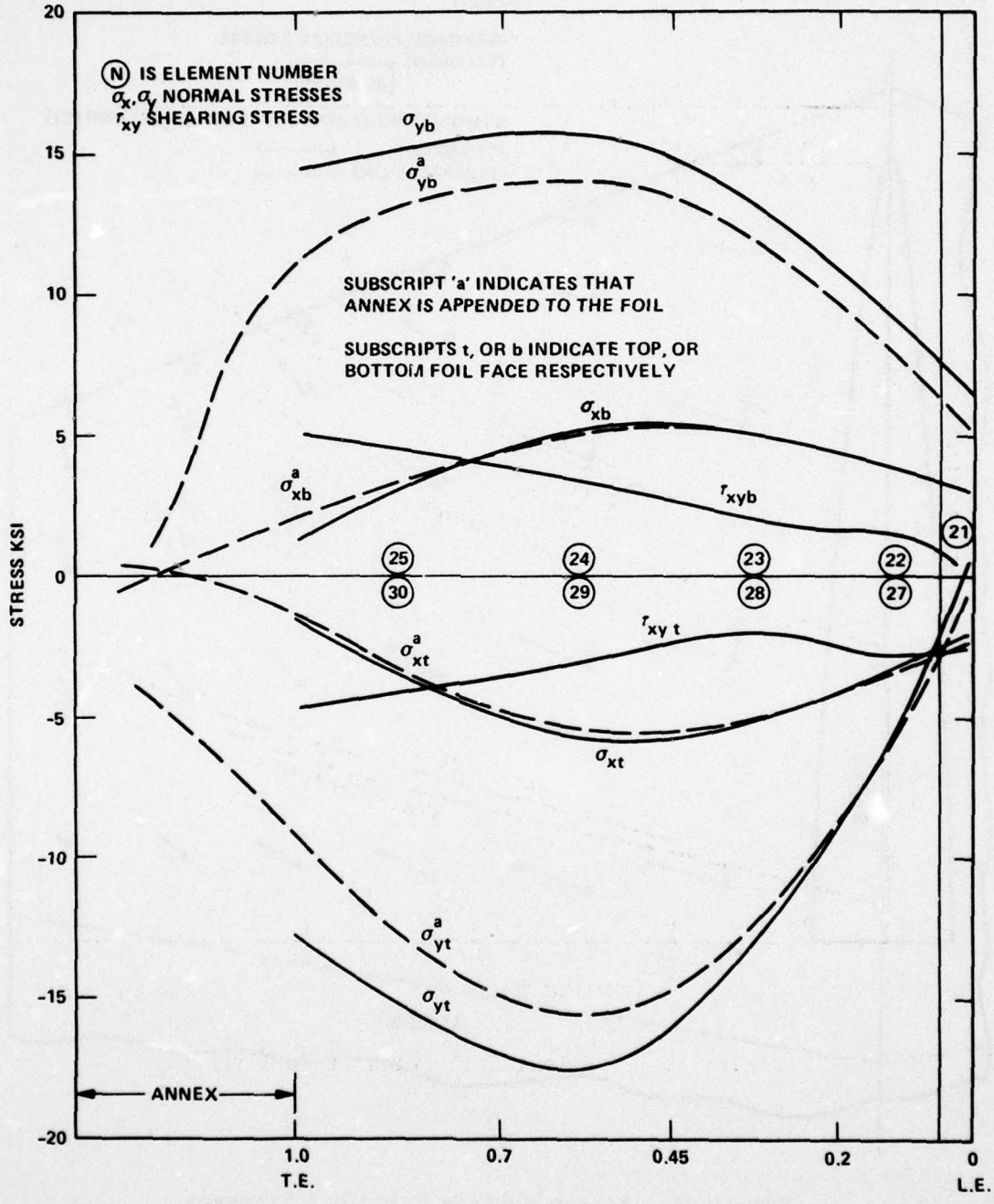


Figure 23 - Foil Stresses along Chord Section, 25 Percent Semispan, Top and Bottom Faces

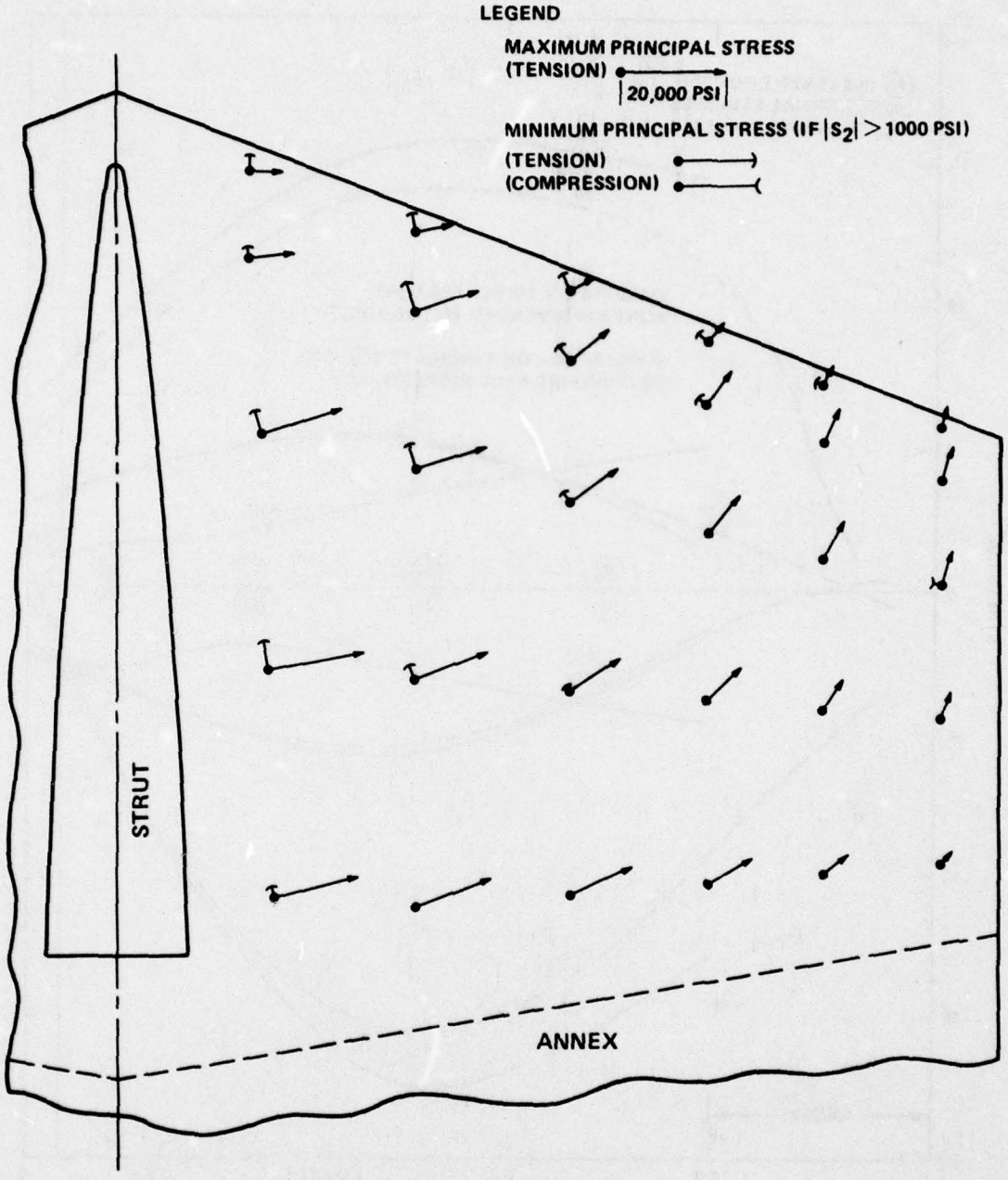


Figure 24 - Bottom Surface Principal Stresses
(Courtesy E. Hoyt)

APPENDIX A

SOLUTION PROCEDURE

DISCUSSION OF ALGORITHM

The solution algorithm warrants attention, particularly by those concerned with making data more manageable. The logic of data organization has not been given the recognition it deserves, and standard texts on numerical methods seldom include techniques for relating the identity and position of matrix elements. Yet it is such techniques that enable the advantage of matrix-sparsity to be exploited and that have enabled notable progress to be made during the last decade in the scope and efficiency with which large matrix problems can be solved on a digital computer. An introductory description of a carefully planned and developed bookkeeping process of a frontal algorithm (a modified version of Irons[14]) is included to illustrate the type of data organization that is essential to effect a numerical solution.

Now consider a direct solution* of the system of equations; from Equation (3)

$$\underset{n\times 1}{B} = \underset{n\times n,\ n\times 1}{[C]\ X}$$

Typically, a Gaussian elimination procedure consists of two operations:

1. Factorization

$$[C] = [L][U]$$

2. Forward elimination and back substitution

$$[L]\ V = B$$

$$[U]\ X = V$$

Let [C] be represented as

---

*A direct solution is the process in which a solution results directly from a fixed number of arithmetic operations.

$$[C]_{n\times n} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & \cdot & \cdot & C_{1n} \\ C_{21} & C_{22} & C_{23} & \cdot & \cdot & C_{2n} \\ C_{31} & C_{32} & C_{33} & \cdot & \cdot & C_{3n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ C_{n1} & C_{n2} & C_{n3} & \cdot & \cdot & C_{nn} \end{bmatrix}$$

and let [U], the upper triangular matrix resulting from the forward operation of a Gaussian elimination, be expressed as

$$[U] = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & \cdot & \cdot & C_{1n} \\ & C_{22}^{(1)} & C_{23}^{(1)} & C_{24}^{(1)} & & \cdot & C_{2n}^{(1)} \\ & & C_{33}^{(2)} & C_{34}^{(2)} & \cdot & \cdot & C_{3n}^{(2)} \\ & & & C_{44}^{(3)} & \cdot & \cdot & C_{4n}^{(3)} \\ & & & & \cdot & \cdot & \cdot \\ & & & & & \cdot & \cdot \\ & & & & & & C_{nn}^{(n-1)} \end{bmatrix}$$

The superscripts of element $C_{ij}^{(m)}$, where m = 1, 2, ..., n - 1, give the exact state of elements during the forward eliminations.

It is known that elimination proceeds in steps; at each step, an unknown X_i is eliminated by performing a set of row operations. This operation can be expressed in terms of an elementary triangular matrix $[L_i]$

$$[L_i]_{n\times n} = \begin{bmatrix} 1 & & & & & & \\ & 1 & & & & & \\ & & 1 & & & & \\ & \ell_{i+1,i} & & 1 & & & \\ & \vdots & & & 1 & & \\ & \vdots & & & & 1 & \\ & \ell_{n,i} & & & & & 1 \end{bmatrix}$$

all blank spaces are zeros

i^{th}-column

with corresponding elements $\ell_{i+1,i} = -\dfrac{C_{i+1,i}^{(i-1)}}{C_{i,i}^{(i-1)}}$

$$\ell_{i+2,i} = -\frac{C_{i+2,i}^{(i-1)}}{C_{i,i}^{(i-1)}}$$

. .

$$\ell_{n,i} = -\frac{C_{n,i}^{(i-1)}}{C_{i,i}^{(i-1)}}$$

for i = 1, 2, ..., n - 1. Now, write the Gaussian elimination process in terms of the notation

$$[U] = [L^{-1}][C]$$

$$= [L_{n-1}] \ldots [L_i] \ldots [L_2][L_1][C]$$

and each step elimination is represented as

$$[C_1] = [L_1]^* [C]$$

$$[C_2] = [L_2]^* [C_1] \quad \text{etc.}$$

and $[U] = [C_{n-1}] = [L_{n-1}]\,[C_{n-2}]$. For instance,

$$[C_1] = \begin{bmatrix} C_{11} & C_{12} & C_{13} & \cdots & C_{1n} \\ & C_{22}^{(1)} & C_{23}^{(1)} & \cdots & C_{2n}^{(1)} \\ & C_{32}^{(1)} & C_{33}^{(1)} & \cdots & C_{3n}^{(1)} \\ & \vdots & \vdots & \ddots & \vdots \\ & C_{n2}^{(1)} & C_{n3}^{(1)} & & C_{nn}^{(1)} \end{bmatrix}$$

Note that $[L] = [L_1^{-1}][L_2^{-1}] \ldots [L_{n-1}^{-1}] = [\ell_{ij}]$ is a lower triangular matrix with one's on the diagonal.

The forward substitution is carried out following the process of factorization $[U] = [L^{-1}][C]$. In this case, the intermediate vector $V = [L^{-1}]B$ will result; explicitly:

---

*Physically, the operations L_1, L_2, etc. are analogous to a succession of relaxations of the constraints held on nodal degrees of freedom, such as joint release in a structural system.

$$V_1 = b_1$$

$$V_2 = b_2 - \ell_{21} V_1$$

. . . . .

$$V_i = b_i - \sum_{j=1}^{i-1} \ell_{ij} V_j$$

And finally, a backward substitution, [U] X = V solves for unknown displacements X.

THE FRONT APPROACH

It has been known that the stiffness matrix [C] is positive definite for a kinematically stable structure under static loads. The Gaussian elimination is guaranteed to be numerically stable irrespective of the order in which the equations are eliminated. No pivot search is necessary, and full advantage of symmetry is realized.

$$C_{ij}^* = C_{ij} - C_{is} \cdot \left(\frac{C_{sj}}{C_{ss}} \right)$$

$$= C_{ji}^*$$

For $C_{ij} = C_{ji}$, $C_{is} = C_{si}$ etc.

In practice, [C] is large and sparse. The important task is to determine the proper order in which the columns of [C] are eliminated to result in minimum growth[17] of nonzeros in [U]. For a banded matrix, the problem is simple; no growth of nonzero terms outside the band can be achieved by choosing diagonal elements as pivots. This provides us with the motivation for permuting [C] in such a way that the permuted matrix

$[P][C][P^t]$ is a minimum bandwidth form, where [P] is a permutation matrix. In the front method, the bandwidth is optimized (or minimized) by virtue of the discretization in the finite element representation. Nodal variables are naturally grouped by element; as a matter of fact, the process involves performing the same function as those claimed by matrix partitioning, or substructuring, used to improve the solution of the banded structure matrix.

An important guideline in dealing with sparse matrices is to store and process only nonzero matrix coefficients to save both space and computer time. The structure stiffness matrix of many practical problems can have from 200-2000 unknowns and thus the conventional full storage of all matrix coefficients has to be abandoned. Instead, a condensed columnwise storage form* retains only nonzero coefficients and some additional information necessary to retrieve them. The following section of this report includes an example to illustrate the data organization. The data are chain-listed to an element sequel. At the individual element level, the data contain, in an orderly manner, the element size (KUREL)** and orientation together with a set of element stiffness coefficients and their addresses (or destination) in front of the active variables.*** The unknown nodal variable (I, J = 1, 2, ..., NDOFPF) which is associated with the coefficient C_{ij} is labeled individually, is coded with identifiers that mark its first, intermediate, and last occurrence during the whole sequel of element-processing (see LVABL, LDEST); and its position in the order of elimination is defined. The number of equations (or variables) being eliminated at each element stage is also recorded for later use in back substitution.

The choice of a computation algorithm for efficient solution of the governing linear equation, Equation (3), is a crucial question, especially for analyzing large structures.[19,20] There does not appear to be a simple

---

*Alternatively, a row-by-row store can be arranged.

**See NOTATION.

***A variable becomes active on its first appearance, namely, the first nonzero term in the corresponding column matrix of [C]. These variables are the constituents of the front.

universal approach to the problem. The method and programing techniques used are often computer and/or problem dependent to some extent, and this results in constraints on the range of applicability of many computer programs. The front algorithm described here, however, is practically independent of the peculiar sparsity pattern of the matrices processed. The procedure is equally effective whether the global structure matrix is dense or sparse.

It is generally admitted that bookkeeping in the frontal algorithm[14] is extensive compared with ordinary band solvers. The implementation of the frontal technique is within the capability of most computers currently in use. The extra effort is well compensated for by the increased efficiency and capability (see Reference 1) of the program development.

Some basic characteristics of the front bookkeeping system are now described and key organization traits are illustrated so that experienced users will be able to modify the procedure if so desired or incorporate their own elements into the program system.

DESCRIPTION OF PROCEDURE

A clear understanding of the solution process is essential to an intelligent application of a numerical procedure such as computer program, PBLADE. As stated earlier, the load-displacement equations, Equation (3), are symmetric, positive, definite, and numerically well-conditioned. Retention of the pivotal terms along the diagonal of the structure matrix enables the full advantage of symmetry to be realized. Good results are obtained by the direct method of elimination and, further, it is immaterial to the order in which the nodal variables are to be eliminated.

In a standard method for solving simultaneous equations, the variables are customarily arranged in the order of elimination. But for a digital computation, it is important to minimize the amount of core storage for intermediate calculations and to avoid physically interchanging rows and columns since this operation can be quite time-consuming, particularly when it involves many recurrences as additional variables are admitted to

the front of active variables. (The intent of the front algorithm described here is to illustrate a procedure which minimizes manipulation of the coefficients of the matrix of the working array, which is the front.)

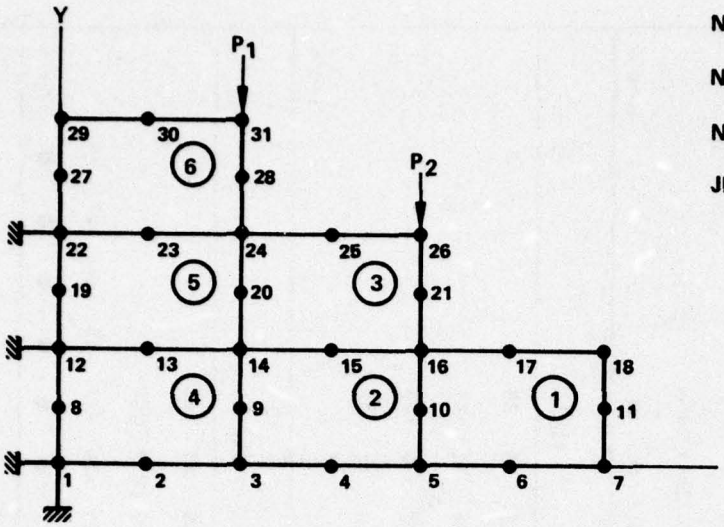
A nodal variable becomes active when it appears for the first time as an element is assembled. This is identified by entering a nonzero stiffness coefficient to the column matrix corresponding to that variable. It remains active until its last appearance. At each intermediate appearance (if any), as the front proceeds from one element to the next, new contributions to that row and column coefficient matrix are collected for the given variable. On the last appearance of an active variable, the coefficients of its row matrix are fully summed and immediately eliminated. The elimination of a variable (accomplished by a set of operations on the row matrix) affects only the coefficients immediately below them (Figure 9). Now, the space which was occupied by the ex-variable (the one being eliminated) is made available (MVABL(LDES)=0*) for active variables introduced by the next element.

The process of the front solution can be viewed literally as the progress of a wavefront propagating through a network of finite elements in the order of increasing element number. Each element introduces new, active variables to the front. The size of the front which, at any one moment, extends over the whole range of active variables, is kept small by filling the void spaces left by variables just eliminated, e.g., Figure 25 and Table 5. (This is done without a bodily shift of variables.) The front width, which in a way is analogous to the bandwidth of a banded matrix is, in most cases, smaller than a bandwidth.

The frontal process employed here has a distinct trait of composite action consisting of the accumulation of stiffness coefficients on the one hand and their elimination on the other. The alternation between assembly and elimination of nodal variables (or labeled unknowns) is repeated as each element is processed in turn. The elements are naturally taken as the unit of implementation. The size of the working vector (also known as

---

*See bookkeeping algorithm, page 72.



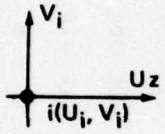
ND = 2

N&LEM = 6

NJ = 31

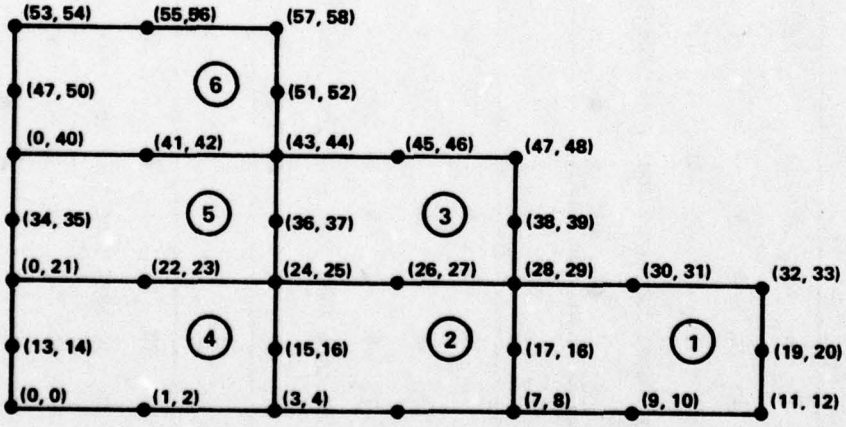
JREST = 4

ELEMENT MESH AND NODAL LABLE



U_i, OR V_i = 0

MEANS FIXED BOUNDARY



COUNT OF NODAL VARIABLES (=JDISP)

Figure 25 - A finite Element Representation of a Two-Dimensional Problem

TABLE 5 - DATA ORGANIZATION OF A FRONT PROCEDURE

FRONT OF ACTIVE VARIABLES - (MVABL)

NELEM 1															KL=16		NVABZ
LVABL(KL)	7	8	9	10	11	12	17	18	19	20	28	29	30	31	32	33	
LDES	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
MVABL(LDES)	-7	-8	-9	-10	-11	-12	-17	-18	-19	-20	-28	-29	-30	-31	-32	-33	
LDEST	3001	3002	2003	2004	2005	2006	3007	3008	2009	2010	4011	4012	2013	2014	2015	2016	
			●	●	●	●			●	●			●	●	●	●	10-e's

NELEM 2															KL=16		
LVABL	3	4	5	6	7	8	15	16	17	18	24	25	26	27	28	29	
LDES	3	4	5	6	1	2	9	10	7	8	13	14	15	16	11	12	
MVABL	-7	-8	-3	-4	-5	-6	-17	-18	-15	-16	-28	-29	-24	-25	-26	-27	
LDEST	3003	3004	2005	2006	1001	1002	3009	3010	1007	1008	5013	5014	3015	3016	11	12	
	●	●			●	●	●	●									6-e's

NELEM 3															KL=16						
LVABL	24	25	26	27	28	29	36	37	38	39	43	44	45	46	47	48					
LDEST	13	14	1015	1016	1011	1012	3001	3002	2005	2006	4007	4008	2017	2018	2019	2020					
MVABL	-36	-37	-3	-4	-38	-39	-43	-44	-15	-16	-28	-29	-24	-25	-26	-27	-45	-46	-47	-48	
					●	●					●	●			●	●	●	●	●	●	10-e's

Note:

1. LDES is the decoded destination, LDEST(LDES is simply the last two digits of the latter).
2. LDES has been deleted from elements 3, 4, 5, and 6.

TABLE 5 (Continued)

FRONT OF ACTIVE VARIABLES - (MVABL)

NELEM 4															KL=13			NVABZ
LVABL	x	x	1	2	3	4	13	14	15	16	x	21	22	23	24	25		
LDEST			2005	2006	1003	1004	2011	2012	1009	1010		3015	3016	3017	13	14		
MVABL	-36	-37	-3	-4	-1	-2	-43	-44	-15	-16	-13	-14	-24	-25	-21	-22	-23	
			●	●	●	●			●	●	●	●						8-e's
NELEM 5															KL=14			
LVABL	x	21	22	23	24	25	34	35	36	37	x	40	41	42	43	44		
LDEST		1015	1016	1017	1013	1014	2003	2004	1001	1002		3005	3006	3009	7	8		
MVABL	-36	-37	-34	-35	-40	-41	-43	-44	-42				-24	-25	-21	-22	-23	
	●	●	●	●									●	●	●	●	●	9-e's
NELEM 6															KL=15			
LVABL	x	40	41	42	43	44	49	50	51	52	53	54	55	56	57	58		
LDEST		1005	1006	1009	1007	1008	2001	2002	2003	2004	2010	2011	2012	2013	2014	2015		
MVABL	-49	-50	-51	-52	-40	-41	-43	-44	-42	-53	-54	-55	-56	-57	-58			
	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●			15-e's

Note:

x denotes a boundary constraint; no label is furnished.

the range of active variables, or front width, which is closely associated with element size) in many cases compares favorably with core requirement for a band-solver for the solution of a given structural problem.

A Simple Example

To illustrate the details of data structure, consider a simple example. Figure 25 shows a two-dimensional continuum under a set of loads and P2 and being supported at three nodal joints. The structure is discretized and represented by six 8-node elements with a grid system of 31 nodes. The system has 58 degrees of freedom; the order of nodal labels and nodal variables (or unknowns) is given; and the elements are labeled.*

After the element stiffness coefficients $[C_{ij}^{e}]$ have been calculated individually for each element with respect to its boundary nodes (I,J,... etc.), those values representing element contributions must be properly labeled with reference to a system of nodal numbers. In assembly then, these element contributions will be distributed correctly to connecting nodes so that the resulting system of equations, Equation (3), will properly represent the load-displacement relation of the structure.

An extensive system of bookkeeping is required to track the labeled nodal variables and their positions in the front as well as to identify variables which are readied for elimination. The program records these data on tape together with the reduced stiffness coefficient and the number of equations eliminated during each element processing cycle. An outline of the on-going process is listed in Table 5.

Start with element number one - NELEM one. The first line gives LVABL(KL),** which is an ordered list of nodal variables which were used in formation of element matrix $[C^{e}]$. Line 2 is an address matrix, LDES,*** which specifies the positions where these nodal variables are assigned in

---

*The order and position of a numbered element affect the front width and, hence, the effectiveness of a solution. Careful attention is required in numbering these elements, especially when a complex structure is analyzed.

**KL = 1, ..., KUREL

***See NOTATION for definitions of these terms.

the front of active variables. It is at these locations that stiffness contributions of the current element will be allocated, or accumulated. The third line, MVABL(LDES), gives the ordered list of active variables which constitute the front. The maximum value of LDES sets the front width, a dominating factor in deciding the core size for a computer run. The fourth line; LDEST,* is a coded version of LDES. It indicates the number of repeated occurrences of a given nodal variable and whether it is the first, intermediate, or the last appearance of the variable.

For the first element, Nodes 6, 7, 11, 17, and 18 (with corresponding global nodal variables 9, 10, 11, 12, 19, 20, 30, 31, 32, and 33) are free standing. These are boundary nodes which are not linked to other elements and hence receive no further contributions of stiffness coefficients from others. For these 10-nodal variables, it is their first as well as their last appearance in the front of active variables; these ten equations are fully summed and are readied for elimination, NVABZ = 10. The elimination phase completes the process for Element 1, and the front moves to the next element.

Now, the second element is assembled. Its stiffness contributions are distributed to positions of active variables of the front according to the sequence prescribed by LDES which sets the addresses for global nodal variables. Those positions in the front which were vacated by the elimination phase of the preceding element are replaced by new active variables (those appeared for the first time) during the assembly of the current element. Those unknown active variables which received final contributions and for which coefficients are fully summed make their last appearance and their six equations are eliminated immediately, NVABZ = 6.

The spaces in the front of active variables is not always fully packed, as can be seen in the assembly of element five; in this case, there are five new active variables and eight open spaces were available. Consequently, three spaces remain vacant.

---

*See NOTATION for definitions of these terms.

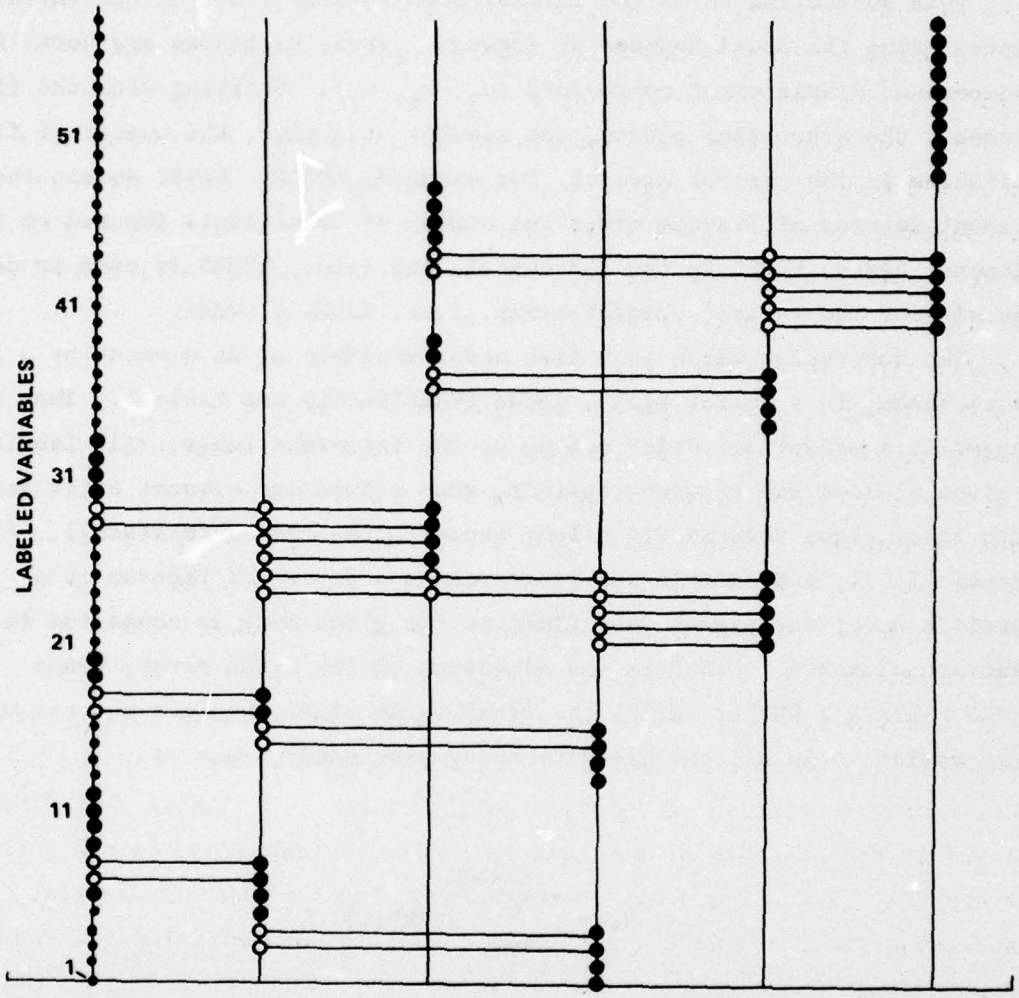
The movement of the front, also known as the wavefront, proceeding from element to element is graphically represented in Figure 26 together with a summary of operational counts (for symbol definitions see the NOTATION).

The solution proceeds from Element 1 to Element 6. At each stage, the front width (or LDES) can be evaluated (see STORIJ) or more directly from an idealized element block diagram (Figure 26). Each circle represents an element variable. Solid circles are those fully summed nodal variables (i.e., no further contribution from other elements) which are eliminated immediately. The circles with attached bar links are variables that remain active through the addition of a follow-on element. The front width can be approximated with adequate accuracy by counting the number of circled variables plus crossing links, if any, over the column directly above the element in question. The actual front width is also given in the figure. The number is, at times, slightly higher than the minimum number counted because of the time delay in filling the free spaces, or vacancies, left by the eliminated variables of the preceding element.

Half bandwidth (HBW=27) is also indicated in the same figure. The bandwidth can be reduced either by exercising care in ordering the nodal number or by utilizing band-reduction algorithms. The density of nonzero terms, or the sparsity of a structure matrix, is dependent on the geometry or topology of a structure. In most cases, front widths are smaller than bandwidths and consequently a front solution is more efficient in terms of computational time and storage requirements.

Bookkeeping Algorithm

The salient features and computational sophistication of the frontal technique can be better assessed through a close inspection of the bookkeeping process which is used to keep track of nodal displacement variables described in Equation (3). In the coded format, the procedure was carried out by Subroutines LABLE and APPEAR.



ELEMENT	(1)	(2)	(3)	(4)	(5)	(6)
KUREL =	16	16	16	13	14	15
LDES =	16	16	20	17	17	15
NVABZ =	10	6	10	8	9	15
LCUREQ =	0	10	16	26	34	43
HBW =	27	27	25	25	24	19

NOTE:
HBW IS THE HALF BANDWIDTH OF MATRIX [C].

Figure 26 - Occurrence of Nodal Variables in a Two-Dimensional Continuum Problem

Subroutine LABLE

This subroutine forms the initial step to keep track of the variables representing the nodal degrees of freedom. These variables are normally independent displacement components (u_i, v_i, w_i). Starting with the first element, the subroutine counts, one element at a time, the number of free variables in the current element, for example, KUREL. KUREL equals the element degrees of freedom minus the number of constraints imposed on that element; hence, KUREL is the current element size. LVMAX is used to denote the size of the largest element array, i.e., KUREL $\leq$ LVMAX.

The subroutine marks each free nodal variable of an element by a label, or nickname, in a vector array, named LVABL(KUREL) see Table 5. They are logged in a predefined order set up by the incidence table. All labels for a given element are grouped together, then stored one element after another with their signs reversed in a long vector array called (NIX(NEW)). A certain label, for example one representing a degree of freedom of a specific node, may repeat many times as the given node is connected to many discrete elements. LVEND is the dimension of the LVABL array; hence LVEND $\geq$ LVMAX $\geq$ KUREL. NIZZ, the final value of NZ, becomes the sum of free variables in all the elements being processed. That is

$$NZ = \sum_{L=1}^{NELEM} KUREL(L)$$

Subroutine APPEAR

This subroutine records the first, last, and intermediate appearances of all element variables. At successive stages, as each element is processed in turn, it examines the front (=MVABL(MVEND*)) for space location as well as the location of free space, i.e., the space vacated by each

---

*MVEND is the size of the working vector of active variables, MVABL(LDES) and sets the upper limit of front width.

variable previously eliminated. At the DO loop 60, each nodal variable which had formerly been given an individual label in LVABL(1,KUREL) will be assigned a space within the working vector MVABL and its position designated by LDES. This is done at statement 30.

Further, it inspects the variable list for recurrence of a variable in the long vector NIX(NIZZ) and counts the number of repetitions of a given label (or nickname). Let KOUNT equal 1000 for each appearance plus 1000. For instance, KOUNT = 2000 indicates a single appearance and KOUNT = 3000 indicates a dual appearance. (This means that the given nodal variable is connected to exactly two elements, etc.) KOUNT = 1000 is reserved, however, for the last appearance of a variable. For each intermediate appearance, KOUNT is suppressed in the coded format of the destination vector LDEST(KUREL), see Table 5.

$$\text{LDEST(L)} = \text{LDES(L)} + \text{KOUNT(L)} \qquad \text{L} = 1, \text{KUREL}$$

This leaves LDEST(L), NIX(LAS) = LDES, a positive quantity; see listing of the subroutine.

The pair LVABL(KL) and LDEST(KL) provides each variable (KL) with a label and an address for the element stiffness coefficient, i.e., its forwarding position at the front of the global stiffness matrix. With the provision of knowledge as to when a given variable is ready for elimination, the information is now complete for assembling the basic element data into a finished system of simultaneous equations, Equation (3).

For instance, LDEST(7) = 1025 implies that a variable (KL=7) is on the twenty-fifth location (LDES=25) of the working vector MVABL(LDES). Its label is given by LVABL(7) or MVABLE(25), and the variable is ready for elimination. In other words, the seventh row and column of the current element matrix $[C^e]$ must accumulate onto the twenty-fifth row and column of the front matrix [G], Figure 9. MVEND is the dimension of front MVABL, or matrix [G].

At DO loop 70, Subroutine CODEST interprets the coded destination vector LDEST(KL). It recovers the element nickname, LVABL(KL), for later

use in Subroutine ELDATA and erases nicknames representing variables eliminated in the front. NVABZ counts the variables eliminated so far, and at the end of LOOP, it gives the total number of equations (or variables) eliminated.

At DO loop 90, input data are regrouped for the current element to be used by Subroutines STIFF and ELDATA. From geometric input (nodal coordinates X(NNPE), Y(NNPE)) and material properties (E, GNU), Subroutine STIFF computes element stiffness coefficients STFNS = $[C^e]$ or

$$\underset{n\times n}{[K_e]} = \int \underset{n\times m}{[B]^T}\ \underset{m\times m}{[D]}\ \underset{m\times n}{[B]} d\ \text{vol}$$

In the case of an 8-node element of two-dimensional continuum where n = 16 and m = 3, the displacement vector $\underline{U}^a = (u_1, u_2, u_3, \ldots u_8, \ldots v_8)^T$.

The matrix $[K_e]$ is later resequenced to a displacement convention of $(u_1, v_1, \ldots u_8, v_8)^T$ compatible with output convention. The coefficients of the stiffness matrix (the upper triangular matrix of $[K_e]$, with columns and rows corresponding to fixed nodes being deleted) as well as the load vector [P] are then grouped together to form a column matrix [EL] by Subroutine ELDATA. These data units are recorded in TAPE 1 for later use in Subroutine FORWRD (Figure 10).

```
*DECK LABLE
      SUBROUTINE LABLE(INCEL,JDISP,LVABL,NIX,NDOFPN,NNPE)
C
C
      COMMON /VAB1/NELPAZ,LVEND,MVEND,NIXEND,LVMAX,NIZZ,MAXNIC
      COMMON /VAB3/NELEM,NELEMZ,KUREL,NIC,LPREQ,NEW,NPAR,NBAXC,NBAXZ
      COMMON /VAB6/CONST,NJ,NRUNO,LHSRHS,L,KOUNT,NELZ,NDELT
       DIMENSION NIX(1),INCEL(NNPE,1),JDISP(NDOFPN,1),LVABL(1)
C
      DO 600 NELEM =1,NELEMZ
       KUREL = 0
       DO 400 J=1,NNPE
       K = INCEL(J,NELEM)
       DO 300  L=1,NDOFPN
       IF(JDISP(L,K).EQ.0) GO TO 300
       KUREL = KUREL + 1
       LVABL(KUREL) = JDISP(L,K)
  300  CONTINUE
  400  CONTINUE
       IF(KUREL.LE.LVMAX) GO TO 401
       LVMAX = KUREL
      IF(LVMAX.GT.LVEND) GO TO 700
  401  DO 500 M=1,KUREL
       NIC = LVABL(M)
       NIZZ = NIZZ + 1
       NIX(NIZZ) = - NIC
  500  CONTINUE
       L = NIXEND + 1 - NELEM
       NIX(L) = NIZZ
  600  CONTINUE
       RETURN
  700  WRITE(6,701) NELEM
  701  FORMAT(1H1,///,10X,* **** TROUBLE IN SUBROUTINE LABLE OCCURED WHIL
     1E PROCESSING ELEMENT NO. *,I4,/,10X,* THE DIMENSION OF LVABL(I) IS
     2INADEQUATE. INCREASE LVEND. *,/,1H1)
       STOP
       END
```

```
*DECK APPEAR
      SUBROUTINE APPEAR(MVABL, NIX,LDEST,LVABL,INCEL,JDISP,COORD,
     1 PLOAD,X,P,JDIS,STFNS,EL,NDOFPN,NDOFPE,NNPE,NF)
C
C     THIS  SUBROUTINE RECORDS FIRST, LAST, AND INTEMEDIATE
C     APPEARNCES  OF A VARIABLE
C
      COMMON /VAB1/NELPAZ,LVEND,MVEND,NIXEND,LVMAX,NIZZ,MAXNIC
      COMMON /VAB2/MAXPA,NVABZ,LCUREQ,MAXELT,NTIREX,LDES,KL,NSTRES
      COMMON /VAB3/NELEM,NELEMZ,KUREL,NIC,LPREQ,NEW,NPAR,NBAXD,NBAXZ
      COMMON /VABL5/JDIREC,JOINT,LEAD,ID,PRATIO,E,T,DET,A,B,C1,C2
      COMMON /VAB6/CONST,NJ,NRUNO,LHSRHS,L,KOUNT,NELZ,NDELT
       COMMON /VAB7/ INTGER,ND,NPT,NPTS,NPZ
      COMMON /VCON/PRES(64),LPT,LNEL
      COMMON /BLAV/GH(3,5),PX(3),PY(3),Q,FF(3,8,2)
       DIMENSION COORD(NDOFPN,1),PLOAD(NDOFPN,1),  X(NNPE,NDOFPN)
       DIMENSION P(NDOFPE),JDIS(NDOFPE),STFNS(NDOFPE,NDOFPE)
       DIMENSION NIX(1),INCEL(NNPE,1),JDISP(NDOFPN,1)
       DIMENSION LVABL(1),EL(1),LDEST(1),MVABL(1),ISWT(8)
      DATA ISWT/5,6,7,8,15,16,19,20/
       IF(NF.EQ.2) GO TO 19
C
C   NF EQUALS TO 1 OR 0
      LL= 3*LPT
      CALL ZEROZ(PLOAD,LL)
  19  N1 =1
      CALL ZEROZ (MVABL,MVEND)
       DO 100  NELEM=1,NELEMZ
       LPREQ = LCUREQ
       LCUREQ = NVABZ
       L = NIXEND + 1 - NELEM
       NZ  = NIX(L)
       KUREL = NZ - N1 + 1
       DO 60  NEW=N1,NZ
       LDES = NIX(NEW)
       NIC = LDES
       IF(NIC.GT.0) GO TO 50
       ISUM = MAXNIC + NIC
       IF(ISUM.LT.0) MAXNIC= - NIC
       LDES = 1
   20  IF(MVABL(LDES) .EQ.0) GO TO 30
       LDES = LDES + 1
       GO TO 20
   30  MVABL(LDES) = NIC
       IF(LDES.GT.MAXPA) MAXPA=LDES
       IF(MAXPA.GT.MVEND) GO TO 110
C
C     MAXPA RECORDS MAXIMUM SIZE OF FRONT FOR DYNAMIC STORAGE
C     FIND OUT THE NUMBER OF REPETITIONS OF LABELS.OR NICKNAMES,I.E. NIC
C     KOUNT EQUALS 1000 FOR EACH APPEARANCE PLUS 1000, KOUNT EQUALS 1000
C     AS GIVEN IN LDEST. IS RESERVED FOR THE LAST APPEARANCE OF A VARIABLE
C
```

```
      KOUNT = 1000
      DO  40 LAS=NEW,NIZZ
      IF(NIX(LAS).NE.NIC) GO TO 40
      NIX(LAS) = LDES
      KOUNT = KOUNT + 1000
      LAST = LAS
   40 CONTINUE
      NIX(LAST) = LDES + 1000
      NIX(NEW) = LDES + KOUNT
      LDES = NIX(NEW)
   50 L = NEW - N1 + 1
      LDEST(L) = LDES
   60 CONTINUE
      N1 = NZ + 1
C     WRITE(6,91) NELEM,(MVABL(I),I=1,MVEND)
   91 FORMAT(1H1, 'ELEMENT NO',I10,'MVABLS' //(13X,8I12))
      DO  70 KL=1,KUREL
      CALL CODEST(LDEST)
      LVABL(KL) = - MVABL(LDES)
      NIC = LVABL(KL)
      IF(NSTRES.NE.0.AND.NSTRES.NE.1) GO TO 70
      MVABL(LDES) = 0
      NVABZ = NVABZ + 1
   70 CONTINUE
      KOUNT=0
      DO 90 I=1,NNPE
      IJ=INCEL(I,NELEM)
      DO 80 J=1,NDOFPN
      KOUNT=KOUNT+1
      X(I,J)=COORD(J,IJ)
      JDIS(KOUNT)=JDISP(J,IJ)
      IF(NF.NE.2) GO TO 80
      P(KOUNT)=PLOAD(J,IJ)
      PLOAD(J,IJ)=0.0
   80 CONTINUE
   90 CONTINUE
      CALL STIFFB (X,E,PRATIO,STFNS,NF)
      IF(NF.EQ.2) GO TO 17
     IF(NELEM.GT.LNEL)  GO TO 17
     DO 18 L=1,8
     IT = ISWT(L)
     JT = INCEL(IT,NELEM)
     DO 18 M=1,3
     LK= M+3*(IT-1)
      P(LK)= PRES(NELEM)*FF(M,L,1)
      PLOAD(M,JT)= PLOAD(M,JT)+P(LK)
C     WRITE (61,94) M, JT,PLOAD(M,JT)
C 94  FORMAT (1X, 'PLOAD(M,JN) M=1,3 '/(2I10, F15.4))
   18 CONTINUE
   17  CALL ELDATA(P,JDIS,LDEST,LVABL,STFNS,EL,NDOFPE)
     IF(NELEM.GT.3) GO TO 100
     WRITE (6,703) NELEM
  703 FORMAT(1X,//,' ELEMENT DATA VIA APPEAR',I6)
```

```
      WRITE (6,26) ((X(NN,L),L=1,ND),NN=1,NNPE)
   26 FORMAT (1H0,29HCOORDINATES ARE XYZ(NNPE,ND)//(3F13.5,10X,3F13.5))
      WRITE (6,92) JDIS
   92 FORMAT(1H0,* JDIS VARIABLES*/(3I13,3X,3I13))
      WRITE (6,25) P
  100  CONTINUE
       REWIND 1
       REWIND 4
   25 FORMAT (1H0, *ELEMENTAL FORCE COMPTS FX  FY  FZ */(3F12.5,10X,
     1 3F12.3))
       RETURN
  110  WRITE(6,111) NELEM
  111  FORMAT(1H1,////,10X,* """ TROUBEL IN SUBROUTINE  "APPEAR" OCCURED
     1 WHILE PROCESSING ELEMENT NO. *,I4,/,10X,*DIMENSION OF MVABL IS IN
     2ADEQUATE. INCREASE MVEND. *,/,1H1)
       STOP
       END



*DECK CODEST
       SUBROUTINE  CODEST(LDEST)
C
C
      COMMON /VAB2/MAXPA,NVABZ,LCUREQ,MAXELT,NTIREX,LDES,KL,NSTRES
       DIMENSION LDEST(1)
C
       LDES = LDEST(KL)
       NSTRES =(LDES/1000) - 1
       LDES = LDES - (NSTRES+1) * 1000
       RETURN
       END
```

APPENDIX B

INPUT LISTINGS OF SAMPLE PROBLEMS*

```
     //  BEGIN SAMPLE PROBLEM NO 1 ///
C
 PROB PLATE BEAM-CX2  L/H=40/6 NELEM12 JUL72 MOD 76

 30000000.    0.30
   20     3    60   110 8330 5890
   12   122    41     0    1     0
    2     1     0     0    3     3
    0     0     0     0    0     0
    0     0
  1 52 20 55 25 63 39 66 44 60 33 61 35 54 23 65 42 53 56 64 67
  2 55 25 58 30 66 44 69 49 61 35 62 37 57 28 68 47 56 59 67 70
  3 58 30111113 69 49100102 62 37107108 94 95 90 91 59112 70101
  4 20 22 25 27 39 41 44 46 33 34 35 36 23 24 42 43 21 26 40 45
  5 25 27 30 32 44 46 49 51 35 36 37 38 28 29 47 48 26 31 45 50
  6 30 32113115 49 51102104 37 38108109 95 96 91 92 31114 50103
  7  1  3  6  8 20 22 25 27 14 15 16 17  4  5 23 24  2  7 21 26
  8  6  8 11 13 25 27 30 32 16 17 18 19  9 10 28 29  7 12 26 31
  9 11 13120122 30 32113115 18 19118119 98 99 95 96 12121 31114
 10 22 72 27 75 41 83 46 86 34 79 36 80 24 73 43 84 71 74 82 85
 11 27 75 32 78 46 86 51 89 36 80 38 81 29 76 48 87 74 77 85 88
 12 32 78115117 51 89104106 38 81109110 96 97 92 93 77116 88105
    1        5.7       0.0       0.6
    2        6.0       0.0       0.6
    3        6.3       0.0       0.6
    4        5.7       3.0       0.6
    5        6.3       3.0       0.6
    6        5.7       6.0       0.6
    7        6.0       6.0       0.6
    8        6.3       6.0       0.6
    9        5.7       9.5       0.6
   10        6.3       9.5       0.6
   11        5.7      13.0       0.6
   12        6.0      13.0       0.6
   13        6.3      13.0       0.6
   14        5.7       0.0       3.3
   15        6.3       0.0       3.3
   16        5.7       6.0       3.3
   17        6.3       6.0       3.3
   18        5.7      13.0       3.3
   19        6.3      13.0       3.3
   20        5.7       0.0       6.0
   21        6.0       0.0       6.0
   22        6.3       0.0       6.0
   23        5.7       3.0       6.0
   24        6.3       3.0       6.0
   25        5.7       6.0       6.0
   26        6.0       6.0       6.0
   27        6.3       6.0       6.0
   28        5.7       9.5       6.0
   29        6.3       9.5       6.0
   30        5.7      13.0       6.0
   31        6.0      13.0       6.0
   32        6.3      13.0       6.0
   33        5.7       0.0       6.3
```

SAMPLE PROBLEM (1) PLATE BEAM STRUCTURE

*Listing of Sample Problem No. 3 is voluminous and may be furnished by addressing a need to the author.

34	6.3	0.0	6.3
35	5.7	6.0	6.3
36	6.3	6.0	6.3
37	5.7	13.0	6.3
38	6.3	13.0	6.3
39	5.7	0.0	6.6
40	6.0	0.0	6.6
41	6.3	0.0	6.6
42	5.7	3.0	6.6
43	6.3	3.0	6.6
44	5.7	6.0	6.6
45	6.0	6.0	6.6
46	6.3	6.0	6.6
47	5.7	9.5	6.6
48	6.3	9.5	6.6
49	5.7	13.0	6.6
50	6.0	13.0	6.6
51	6.3	13.0	6.6
52	3.0	0.0	6.0
53	4.35	0.0	6.0
54	3.0	3.0	6.0
55	3.0	6.0	6.0
56	4.35	6.0	6.0
57	3.0	9.5	6.0
58	3.0	13.0	6.0
59	4.35	13.0	6.0
60	3.00	0.0	6.3
61	3.00	6.0	6.3
62	3.00	13.0	6.3
63	3.0	0.0	6.6
64	4.35	0.0	6.6
65	3.0	3.0	6.6
66	3.0	6.0	6.6
67	4.35	6.0	6.6
68	3.0	9.5	6.6
69	3.0	13.0	6.6
70	4.35	13.0	6.6
71	7.65	0.0	6.0
72	9.0	0.0	6.0
73	9.0	3.0	6.0
74	7.65	6.0	6.0
75	9.0	6.0	6.0
76	9.0	9.5	6.0
77	7.65	13.0	6.0
78	9.0	13.0	6.0
79	9.0	0.0	6.3
80	9.0	6.0	6.3
81	9.0	13.0	6.3
82	7.65	0.0	6.6
83	9.0	0.0	6.6
84	9.0	3.0	6.6
85	7.65	6.0	6.6
86	9.0	6.0	6.6
87	9.0	9.5	6.6
88	7.65	13.0	6.6
89	9.0	13.0	6.6
90	3.0	16.5	6.6

91	5.7	16.5	6.6
92	6.3	16.5	6.6
94	3.0	16.5	6.0
95	5.7	16.5	6.0
96	6.3	16.5	6.0
93	9.0	16.5	6.6
97	9.0	16.5	6.0
105	7.65	20.0	6.6
106	9.0	20.0	6.6
110	9.0	20.0	6.3
116	7.65	20.0	6.0
117	9.0	20.0	6.0
98	5.7	16.5	0.6
99	6.3	16.5	0.6
100	3.0	20.0	6.6
101	4.35	20.0	6.6
102	5.7	20.0	6.6
103	6.0	20.0	6.6
104	6.3	20.0	6.6
107	3.0	20.0	6.3
108	5.7	20.0	6.3
109	6.3	20.0	6.3
111	3.0	20.0	6.0
112	4.35	20.0	6.0
113	5.7	20.0	6.0
114	6.0	20.0	6.0
115	6.3	20.0	6.0
118	5.7	20.0	3.3
119	6.3	20.0	3.3
120	5.7	20.0	0.6
121	6.0	20.0	0.6
122	6.3	20.0	0.6

2	1
21	1
40	1
103	1
114	1
121	1
111	3
112	3
113	3
114	3
115	3
116	3
117	3
118	3
119	3
120	3
121	3
122	3
1	2
2	2
3	2
14	2
15	2
20	2
21	2

```
 22     2
 33     2
 34     2
 39     2
 40     2
 41     2
 52     2
 53     2
 60     2
 63     2
 64     2
 71     2
 72     2
 79     2
 82     2
 83     2
 40          0.0         0.0      -5000.
0000000000000000000000000
```

AD-A056 878 DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 9/2
CURVED FINITE ELEMENTS COMPUTER PROGRAM - PBLADE USER'S MANUAL.(U)
APR 78 J H MA

UNCLASSIFIED DTNSRDC-78/016 NL

2 OF 2
ADA
056878



END
DATE FILMED
9-78
DDC

```
//  BEGIN SAMPLE PROBLEM NO 2 ///

PROB SUPER-CAV PROPELLER BLADE (P-3604) NEL15 MAR73MOD6

30000000.    0.30
   20     3    60    90 6260 5800
   15   148    24     0    1    1
    2    62     0     0    3    3
    0     0     0     0    0    0
    0     0
  1 12 14 38 40  1  3 27 29  8  9 34 35 23 24 19 20 13 39  2 28
  2 14 16 40 42  3  5 29 31  9 10 35 36 24 25 20 21 15 41  4 30
  3 16 18 42 44  5  7 31 33 10 11 36 37 25 26 21 22 17 43  6 32
  4 38 40 64 66 27 29 53 55 34 35 60 61 49 50 45 46 39 65 28 54
  5 40 42 66 68 29 31 55 57 35 36 61 62 50 51 46 47 41 67 30 56
  6 42 44 68 70 31 33 57 59 36 37 62 63 51 52 47 48 43 69 32 58
  7 64 66 90 92 53 55 79 81 60 61 86 87 75 76 71 72 65 91 54 80
  8 66 68 92 94 55 57 81 83 61 62 87 88 76 77 72 73 67 93 56 82
  9 68 70 94 96 57 59 83 85 62 63 88 89 77 78 73 74 69 95 58 84
 10 90 92116118 79 81105107 86 87112113101102 97 98 91117 80106
 11 92 94118120 81 83107109 87 88113114102103 98 99 93119 82108
 12 94 96120122 83 85109111 88 89114115103104 99100 95121 84110
131161181421441051071311331121131381391271281231241171431061132
141181201441461071091331351131141391401281291241251191451081134
151201221461481091111351371141151401411291301251261211471101136
   1       -2.748      1.300      1.837
   3       -1.863      0.843      2.086
   4       -0.670      0.192      2.242
   5        0.544     -0.435      2.208
   6        1.781     -0.988      2.021
   7        3.044     -1.438      1.731
  12       -2.776      1.256      1.867
  14       -1.928      0.725      2.130
  15       -0.783     -0.0275     2.250
  16        0.377     -0.7505     2.121
  17        1.563     -1.351      1.799
  18        2.778     -1.796      1.356
   8       -2.762      1.278      1.852
   9       -1.8955     0.784      2.108
  10        0.4605    -0.5925     2.1645
  11        2.911     -1.617      1.5435
  27       -2.367      1.816      3.108
  29       -1.648      1.163      3.407
  30       -0.659      0.261      3.591
  31        0.376     -0.602      3.549
  32        1.464     -1.372      3.328
  33        2.607     -2.019      2.981
  38       -2.400      1.781      3.128
  40       -1.729      1.069      3.438
  41       -0.803       .0856     3.599
  42        0.172     -0.846      3.499
```

SAMPLE PROBLEM (2) A SUPERCAVITATING PROPELLER

43	1.196	-1.668	3.191
44	2.272	-2.343	2.733
34	-2.384	1.799	3.118
35	-1.688	1.116	3.423
36	0.274	-0.724	3.524
37	2.440	-2.181	2.857
53	-1.965	2.095	4.485
55	-1.408	1.339	4.766
56	-0.624	0.315	4.940
57	0.225	-0.666	4.905
58	1.149	-1.558	4.699
59	2.151	-2.336	4.364
64	-2.001	2.066	4.498
66	-1.490	1.269	4.785
67	-0.768	.187	4.947
68	0.0167	-0.848	4.877
69	0.878	-1.783	4.618
70	1.819	-2.590	4.218
60	-1.983	2.081	4.492
61	-1.449	1.304	4.776
62	0.121	-0.757	4.891
63	1.985	-2.463	4.291
79	-1.547	2.111	5.936
81	-1.134	1.353	6.153
82	-0.541	0.341	6.291
83	0.122	-0.632	6.266
84	0.862	-1.537	6.110
85	1.684	-2.357	5.843
90	-1.585	2.086	5.945
92	-1.210	1.301	6.164
93	-0.666	0.254	6.295
94	-0.053	-0.753	6.255
95	0.637	-1.689	6.069
96	1.410	-2.534	5.768
86	-1.566	2.099	5.941
87	-1.172	1.327	6.159
88	0.035	-0.693	6.262
89	1.547	-2.446	5.806
131	-0.3409	0.665	8.795
133	-0.2364	0.439	8.809
134	-0.0932	0.139	8.819
135	0.056	-0.158	8.816
136	0.2122	-0.451	8.8084
137	0.3757	-0.7404	8.789
142	-0.358	0.6564	8.7955
144	-0.2843	0.4148	8.8102
145	-0.204	0.083	8.8196
146	-0.0881	-0.231	8.817
147	0.0769	-0.5197	8.8047
148	0.306	-0.7756	8.786
138	-0.3495	0.661	8.7952
139	-0.2604	0.427	8.8096
140	-0.0161	-0.195	8.818
141	0.341	-0.758	8.7875
105	-1.034	1.713	7.456
107	-0.771	1.1004	7.5704
108	-0.3864	0.293	7.6444

109	0.053	-0.486	7.635
110	0.5545	-1.224	7.551
111	1.121	-1.9146	7.4065
116	-1.0624	1.697	7.460
118	-0.826	1.069	7.575
119	-0.4665	0.2466	7.646
120	-0.0578	-0.550	7.630
121	0.405	-1.310	7.537
122	0.946	-2.0125	7.381
112	-1.048	1.705	7.458
113	-0.799	1.085	7.573
114	-0.0024	-0.518	7.6325
115	1.034	-1.9636	7.3938
2	-2.3063	1.0755	1.971
13	-2.3533	0.9943	2.012
28	-2.009	1.494	3.269
39	-2.065	1.430	3.296
54	-1.688	1.720	4.636
65	-1.747	1.670	4.653
80	-1.341	1.733	6.053
91	-1.399	1.696	6.063
106	-0.903	1.407	7.518
117	-0.9425	1.385	7.522
132	-0.2887	0.552	8.8025
143	-0.3296	0.5314	8.8037
19	-2.571	1.591	2.454
20	-1.764	1.025	2.740
21	0.463	-0.533	2.876
22	2.840	-1.765	2.333
23	-2.6014	1.552	2.479
24	-1.839	0.918	2.777
25	0.272	-0.812	2.810
26	2.529	-2.112	2.023
45	-2.168	1.986	3.786
46	-1.531	1.269	4.082
47	0.2955	-0.647	4.226
48	2.380	-2.211	3.659
49	-2.202	1.955	3.802
50	-1.615	1.187	4.107
51	0.0857	-0.8585	4.188
52	2.038	-2.503	3.466
71	-1.760	2.139	5.202
72	-1.277	1.369	5.456
73	0.1675	-0.6603	5.586
74	1.921	-2.387	5.0934
75	-1.797	2.112	5.213
76	-1.355	1.309	5.471
77	-0.030	-0.813	5.566
78	1.614	-2.601	4.987
97	-1.305	1.974	6.690
98	-0.964	1.268	6.859
99	0.088	-0.572	6.952
100	1.419	-2.205	6.617
101	-1.341	1.952	6.696
102	-1.036	1.223	6.867
103	-0.0564	-0.6628	6.943
104	1.192	-2.341	6.5704

123	-0.700	1.269	8.228
127	-0.721	1.257	8.230
124	-0.514	0.822	8.284
128	-0.560	0.798	8.287
125	0.0522	-0.3382	8.318
129	-0.0637	-0.400	8.315
126	0.762	-1.417	8.204
130	0.6457	-1.478	8.193
8	1		
9	1		
10	1		
11	1		
2	2		
4	2		
6	2		
13	2		
15	2		
17	2		
1	3		
2	3		
3	3		
4	3		
5	3		
6	3		
7	3		
12	3		
13	3		
14	3		
15	3		
16	3		
17	3		
18	3		
12	0.097	0.091	-0.052
13	-0.30	-0.306	0.185
14	0.270	0.283	-0.028
15	-0.746	-0.967	0.074
16	0.380	0.454	0.071
17	-0.654	-0.959	-0.420
18	0.183	0.240	0.073
23	-0.247	-0.223	0.146
24	-1.182	-1.288	0.362
25	-1.480	-1.909	-0.34
26	-0.672	-0.932	-0.463
38	0.209	0.164	-0.091
39	-0.694	-0.527	0.309
40	0.543	0.537	-0.069
41	-1.722	-1.70	0.152
42	0.791	0.866	0.112
43	-1.511	-1.799	-0.644
44	0.380	0.450	0.133
49	-0.290	-0.175	0.108
50	-1.397	-1.022	0.296
51	-1.742	-1.618	-0.179
52	-0.792	-0.861	-0.292
64	0.229	0.128	-0.069
65	-0.757	-0.408	0.231
66	0.601	0.430	-0.071

67	-1.897	-1.347	0.185
68	0.873	0.735	0.031
69	-1.668	-1.603	-0.338
70	0.422	0.387	0.068
75	-0.296	-0.134	0.081
76	-1.435	-0.784	0.255
77	-1.808	-1.338	0.005
78	-0.828	-0.771	-0.122
90	0.218	0.094	-0.054
91	-0.725	-0.301	0.178
92	0.574	0.325	-0.083
93	-1.826	-1.028	0.231
94	0.841	0.585	-0.036
95	-1.638	-1.340	-0.055
96	0.407	0.316	0.006
101	-0.262	-0.094	0.064
102	-1.267	-0.561	0.237
103	-1.612	-1.010	0.146
104	-0.755	-0.611	0.020
116	0.156	0.054	-0.039
117	-0.552	-0.189	0.136
118	0.394	0.194	-0.062
119	-1.401	-0.661	0.218
120	0.589	0.264	-0.078
121	-1.278	-0.917	0.113
122	0.287	0.193	-0.021
127	-0.146	-0.043	0.042
128	-0.701	-0.257	0.154
129	-0.909	-0.483	0.131
130	-0.417	-0.318	0.091
142	0.059	0.020	-0.014
143	-0.142	-0.044	0.041
144	0.198	0.080	-0.036
145	-0.368	-0.144	0.067
146	0.277	0.154	-0.050
147	-0.330	-0.228	0.071
148	0.134	0.087	0.012

000000000000000000000000

REFERENCES

1. Ma, J.H. et al., "Propeller Stress Calculation Using Curved Finite Elements," Society of Naval Architects and Marine Engineers Symposium - Propellers 75, Philadelphia, PA. (Jul 1975).

2. Taylor, D.W., "The Speed and Power of Ships," Ransdell, Inc., Washington, D.C. (1933).

3. "Principles of Naval Architecture," Edited by J.P. Comstock, SNAME, New York, N.Y. (1967).

4. Cohen, J.W., "On Stress-Calculations in Helicoidal Shells and Propeller Blades," Netherlands Research Center T.N.O. for Shipbuilding and Navigation, Report 21S, Delft, The Netherlands (Jul 1955).

5. Conolly, J.E., "Strength of Propellers," Transactions Royal Institute of Naval Architects, Vol. 10 (1961).

6. Atkinson, P., "On the Choice of Method for the Calculation of Stress in Marine Propellers," Transactions Royal Institute of Naval Architects, Vol. 17 (1968).

7. Wereldsma, R., "Stress Measurements on a Propeller Model for 42,000 DWT Tanker," International Ship Progress, Vol. 12 (Sep 1965).

8. McCarthy, J.H. and J.S. Brock, "Static Stresses in Wide-Bladed Propellers," Naval Ship Research and Development Center, NSRDC Report 3182 (Sep 1969).

9. Boswell, R.J., "Static Stress Measurements on a Highly Skewed Propeller Blade," NSRDC Report 3247 (Dec 1969).

10. Dhir, S.K. and J.P. Sikora, "Holographic Displacement Measurements on a Highly Skewed Propeller Blade," NSRDC Report 3680 (Aug 1971).

11. Ma, J.H., "Stress Analysis of Complex Ship Components by a Numerical Procedure Using Curved Finite Elements," NSRDC Report 4057 (Jul 1973).

12. Ralston, A., "A First Course in Numerical Analysis," McGraw-Hill, New York (1965).

PRECEDING PAGE BLANK-NOT FILMED

13. Zienkiewicz, O.C., "The Finite Element Method in Engineering Science," McGraw-Hill, London (1971).

14. Irons, B.M., "A Frontal Solution Program for Finite Element Analysis," Int. J. for Numerical Method in Engineering, Vol. 2 (1970).

15. Tinney, W.F. and J.W. Walker, "Direct Solutions of Sparse Network Equations by Optimally Ordered Triangular Factorization," Proceedings Institute of Electrical and Electronics Engineering, Vol. 55 (Nov 1967).

16. Jennings A. and A.D. Tuff, "A Direct Method for the Solution of Large Sparse Symmetric Simultaneous Equations," Conference on Large Sparse Set of Linear Equations, Saint Catherine's College, Oxford, England, (April 1970). Also published by Academic Press, Edited by J.K. Reid, London (1971).

17. Tewarson, R.P., "Sorting and Ordering Sparse Linear Systems," in "Large Sparse Set of Linear Equations," Edited by J.K. Reid, Academic Press, London (1971).

18. Mondkar, D.P. and P.H. Graham, "Toward Optimal Incore Equation Solving," Computers and Structures, Vol. 4 (1974).

19. Jensen, H.G. and G.A. Park, "Efficient Solutions for Linear Matrix Equations," J. Structure Division, American Society of Civil Engineers, Vol. 96, paper 7007 (Jan 1970).

20. Wilson, E.L. et al., "Direct Solution of Large Systems of Linear Equations," Computers and Structures, Vol. 4 (1974).

21. Hoyt, E.D. et al., "Structural Design Study of TAP-1 Supercavitating Foil and Strut," DTNSRDC Report 77-0094 (Dec 1977).

INITIAL DISTRIBUTION

Copies	
1	DDR&E Library
3	CNO 1 OP 971 1 OP 098 1 OP 098T
2	CHONR 1 Code 439 1 Code 463
3	CHNAVMAT 1 MAT 03T 1 MAT 0333 1 Lib
1	USNA
1	NAVPGSCOL
1	USNROTC & NAVADMINU, MIT
1	NAVWARCOL
1	DNL
1	NRL Tech Lib
8	NAVSEA 1 SEA 2052 1 SEA 03Z 1 SEA 031 1 SEA 034 1 SEA 03412 1 SEA 03413 1 SEA 0342 1 SEA 03423
2	NAVAIRSYSCOM 1 Eng Div (Code 520) 1 Str Br (Code 5302)
1	NOSC
1	NAVSHIPYD PUGET
1	NAVSHIPYD CHASN
1	NAVSHIPYD LBEACH
1	NAVSHIPYD MARE
1	NAVSHIPYD NORVA
1	NAVSHIPYD PEARL
1	NAVSHIPYD PHILA
1	NAVSHIPYD PTSMH
15	NAVSEC 1 SEC 6110 1 SEC 6112 1 SEC 6113 1 SEC 6114D 1 SEC 6115 1 SEC 6120D 2 SEC 6128 1 SEC 6129 1 SEC 6132 1 SEC 6137 1 SEC 6139 1 SEC 6140B 1 SEC 6144 1 SEC 6148
1	AFOSR Mechanics Div
1	Air Force Appl Mech Grp Wright-Patterson AFB
12	DDC
2	U.S. COGARD 1 Testing & Devel Div 1 Ship Structures Comm
1	Lib of Congress
2	MARAD 1 Div of Ship Design 1 Off of Res & Dev

Copies	
1	NASA, Langley Res Center, Lib
1	NASA, Lewis Res Cen Lib
1	NASA, Sci & Tech Info Fac
1	NSF Engr Div Lib
1	BUSTAND
2	Univ of California, Berkeley 1 J.R. Paulling 1 Eng. Lib
1	Cal Inst of Tech, Lib
1	Case Western Reserve Univ Dept Civ Engr
1	Catholic Univ Dept Mech Engr
1	George Washington Univ School of Engr & Applied Sci
2	Univ of Illinois 1 College of Engr, Lib 1 W.C. Schnobrich
1	Univ of Iowa Iowa Inst of Hydr Res
1	Lehigh Univ Dept Mech
1	Univ of Maryland Dept of Mech Engr
1	Mass Inst of Tech Dept Ocean Engr
1	Univ of Michigan Dept NAME
1	Southwest Res Inst

Copies	
1	Stanford Univ Dept Civ Engr
1	Stevens Inst Tech Davidson Lab
1	Virginia Poly Inst & State Univ/ Dept Engr Mech
1	Webb Inst
2	National Academy of Sci 1 National Res Council 1 Ship Hull Res Comm
1	SNAME
1	American Bureau of Shipping
1	Aerojet General
1	Bethlehem Steel
1	Boeing Company/ Aerospace Group
1	Douglas Aircraft
1	General Applied Science Lab
1	General Dynamics Corp Electric Boat Div
1	Gibbs & Cox
1	Grumman Aerospace Corp
1	Hydronautics
1	Newport News Shipbuilding, Lib
1	United Aircraft Corp Hamilton Standard Div

CENTER DISTRIBUTION

Copies	Code	Name
1	1100	W. Ellsworth
1	1150	R. Johnston
1	1151	W.C. O'Neill
1	1500	W.E. Cummins
1	1500	J.B. Hadler
1	1520	R. Wermter
1	1524	W.C. Lin
1	1532	G.F. Dobay
1	1540	W.B. Morgan
1	1544	R.A. Cumming
1	1552	J.H. McCarthy, Jr.
1	1560	G.R. Hagen
1	1600	H.R. Chaplin
1	1700	W.W. Murray
1	1720	M. Krenzke
1	1730	A. Stavovy
1	1730.5	J. Adamchak
25	1730.5	J. Ma
1	1780	L. Becker
1	1800	G.H. Gleissner
1	1805	E. Cuthill
1	1900	M.M. Sevik
1	1965	Y.N. Liu
1	2700	L.J. Argiro
1	274	L.J. Argiro
1	2810	R. Niederberger
1	2803	J.J. Kelly
1	2940	E. Screen
10	5214.1	Reports Distribution
1	522.1	Library (C)
1	522.2	Library (A)

*U.S. GOVERNMENT PRINTING OFFICE: 1978 261-250/0013 1-3