https://ntrs.nasa.gov/search.jsp?R=19800025274 2020-01-26T20:59:22+00:00Z

NASA Conference Publication 2151

N80-33782

Ninth NASTRAN® Users' Colloquium

Proceedings of a colloquium held
at John F. Kennedy Space Center,
Kennedy Space Center, Florida
October 22-23, 1980

NASA Conference Publication 2151

Ninth NASTRAN® Users' Colloquium

Proceedings of a colloquium held
at John F. Kennedy Space Center,
Kennedy Space Center, Florida
October 22-23, 1980

FOREWORD

NASTRAN® (NASA STRUCTURAL ANALYSIS) is a large, comprehensive, nonproprietary, general purpose finite element computer code for structural analysis which was developed under NASA sponsorship and became available to the public in late 1970. It can be obtained through COSMIC (Computer Software Management and Information Center), Athens, Georgia, and is widely used by NASA, other government agencies, and industry.

NASA currently provides continuing maintenance of NASTRAN® through COSMIC. Because of the widespread interest in NASTRAN®, and finite element methods in general, the Ninth NASTRAN® Users' Colloquium was organized and held at the Kennedy Space Center, October 22-23, 1980. (Papers from previous colloquia held in 1971, 1972, 1973, 1975, 1976, 1977, 1978, and 1979 are published in NASA Technical Memorandums X-2378, X-2637, X-2893, X-3278, X-3428, and NASA Conference Publications 2018, 2062, and 2131.) The Ninth Colloquium provides some comprehensive general papers on the application of finite element methods in engineering, comparisons with other approaches, unique applications, pre- and post-processing or auxiliary programs, and new methods of analysis with NASTRAN.

Individuals actively engaged in the use of finite elements or NASTRAN® were invited to prepare papers for presentation at the Colloquium. These papers are included in this volume. No editorial review was provided by NASA or COSMIC, however, detailed instructions were provided each author to achieve reasonably consistent paper format and content. The opinions and data presented are the sole responsibility of the authors and their respective organizations.

Cochairmen:

Robert L. Brugh
COSMIC
University of Georgia
Athens, GA 30602

and

Henry Harris
John F. Kennedy Space Center
Kennedy Space Center, FL 32899

CONTENTS

NEW CAPABILITIES AND MODIFICATIONS FOR NASTRAN

LEVEL 17.5 AT DTNSRDC

Myles M. Hurwitz
David W. Taylor Naval Ship Research and Development Center

SUMMARY

Since 1970 DTNSRDC has been modifying NASTRAN to suit various Navy requirements. These modifications have included major new capabilities as well as user conveniences and error corrections. This paper describes the new features added to NASTRAN Level 17.5 at DTNSRDC. The subject areas of the additions include magnetostatics, piezoelectricity, fluid-structure interactions, isoparametric finite elements, and shock design for shipboard equipment.

INTRODUCTION

The David W. Taylor Naval Ship Research and Development Center (DTNSRDC) has been involved with NASTRAN since 1968. In the ensuing 3-4 years, prior to the first public release of the program in 1972, engineers at DTNSRDC gained valuable experience with NASTRAN, often interacting with the program developers on various theoretical, programming, and user aspects. The result of that early effort was a detailed NASTRAN evaluation report, which included a brief description of our first modification to NASTRAN--the addition of a heat transfer finite element to the NASTRAN element library (ref. 1).

In subsequent years, the DTNSRDC modifications to NASTRAN were many and varied, ranging from error correction and user conveniences to new finite elements and new functional modules and rigid formats.

Since Level 17.5 was released in the Spring of 1979, our NASTRAN modification effort has remained vigorous. The subject areas of new capabilities and uses include magnetostatics, piezoelectricity, fluid-structure interactions, isoparametric finite elements, and shock design of shipboard equipment. This paper describes these subject areas as we have implemented them into NASTRAN, sample applications of some of these areas, and the correction of an important program error. All of this work will be transferred to the DTNSRDC version of Level 17.6 after the standard version is released.

MAGNETOSTATICS

The prediction of static magnetic fields around ships and submarines is of

concern to the Navy because of the need for these craft to remain undetected. A numerical procedure which can predict these fields can also be used to evaluate systems which might reduce the fields, e.g. degaussing coils. Such a procedure, making use of a scalar potential rather than the less efficient vector potential, was described in reference 2. Reference 3 describes a capability for computing the magnetostatic fields about axisymmetric structures that was added to NASTRAN. However, that work was limited to the TRAPRG and TRIARG finite elements and to axisymmetric current coils. In Level 17.5, the analysis has been extended to general built-up and continuum structures with general current coil configurations. The finite elements allowed are those available for NASTRAN heat transfer analysis (ref. 4), for reasons which may be seen from the brief description of the applicable theory which follows.

The applicable Maxwell equations governing the magnetostatic case are

$$\nabla \times H = J \tag{1}$$

$$\nabla \cdot B = 0 \tag{2}$$

where

H = magnetic field strength or intensity
B = magnetic induction or flux density
J = current density

The constitutive relation

$$B = \mu H \tag{3}$$

where μ is the magnetic permeability is also required. If H is separated into two parts

$$H = H_c + H_m \tag{4}$$

where H_c, the field in a homogeneous region due to current density J (as might occur in a current coil), may be computed using the Biot-Savart law, (ref. 5), then H_m becomes the only unknown. By equations (1) and (4),

$$\nabla \times H_m = 0 \tag{5}$$

so that

$$H_m = \nabla\phi \tag{6}$$

where ϕ is a scalar potential. By equations (2), (3), (4), and (6),

$$\nabla \cdot \mu\nabla\phi + \nabla \cdot \mu H_c = 0 \tag{7}$$

which can be put into the standard form

$$K\phi = F \tag{8}$$

where

$$k_{ij} = \int_V (\nabla N_i)^T \mu (\nabla N_j) dV \tag{9}$$

$$f_i = -\int_V (\nabla N_i)^T \mu H_c dV \tag{10}$$

N_i being the displacement function for a finite element at the i^{th} grid point. Equation (9) is of the same form as that required to compute the conductivity matrix in heat transfer, with μ representing magnetic permeability rather than thermal conductivity. Equation (10), which is dependent on the finite element type, is not in a standard heat transfer form and was added to NASTRAN along with the new bulk data cards needed to specify H_c. Current coils may be defined, from which NASTRAN computes H_c using the Biot-Savart law, or H_c may be specified as coming from an ambient field, or a combination of both sources of H_c may be given.

Equation (6) gives the unknown H_m, which, in standard NASTRAN terminology, is the thermal gradient, and equations (4) and (3) yield the final result.

One unanticipated addition to NASTRAN was required when it was discovered that the program did not compute thermal gradients for the isoparametric solids IHEX1, IHEX2, and IHEX3, as needed by equation (6). An example of this capability is shown in figure 1. The finite element model depicts a solid sphere (shaded part) which has been placed into a uniform, ambient, axial magnetic field. TRIARG elements were used and only the upper half was modeled due to symmetry. The NASTRAN results are compared with theoretical results in Table 1.

PIEZOELECTRICITY

The analysis of sonar transducers requires accounting for the effects of their piezoelectric materials. The theory for these materials couples the structural displacements to electric potentials (ref. 6). This theory was incorporated into NASTRAN only for the TRIAAX and TRAPAX finite elements (ref. 7). These elements, trapezoidal and triangular in cross-section respectively, are solid, axisymmetric rings whose degrees-of-freedom are expanded into Fourier series, thus allowing nonaxisymmetric loads.

The piezoelectric constitutive relations may be written as

$$\begin{Bmatrix} \{\sigma\} \\ \{D\} \end{Bmatrix} = \begin{bmatrix} [c^E] & [e] \\ [e]^T & -[\varepsilon^S] \end{bmatrix} \begin{Bmatrix} \{\varepsilon\} \\ \{E\} \end{Bmatrix} \tag{11}$$

where $\{\sigma\}$ = stress components = $\left[\sigma_{rr}, \sigma_{zz}, \sigma_{\theta\theta}, \sigma_{rz}, \sigma_{r\theta}, \sigma_{z\theta}\right]^T$

$\{D\}$ = components of electric flux density = $\lfloor D_{rr}, D_{zz}, D_{\theta\theta} \rfloor^T$

$\{\varepsilon\}$ = mechanical strain components

$\{E\}$ = electric field components

$[c^E]$ = elastic stiffness tensor evaluated at constant electric field

$[e]$ = piezoelectric tensor

$[\varepsilon^S]$ = dielectric tensor evaluated at constant mechanical strain

The displacement vector of a point within an element is taken to be

$$\{\bar{u}\} = \begin{Bmatrix} u \\ v \\ w \\ \phi \end{Bmatrix} \tag{12}$$

where u, v, and w are the ring displacements in the radial, tangential, and axial directions, respectively, and ϕ is the electric potential. The latter degree-of-freedom is taken to be the fourth degree-of-freedom at each ring. Each of these quantities is expanded into a Fourier series with respect to the azimuth position θ. The Fourier series for the electric potential ϕ has the same form as the Fourier series for radial displacement u, as given in the NASTRAN Theoretical Manual (ref. 4).

The "stiffness" matrix for the N^{th} harmonic is

$$[K^{(N)}] = \pi \int_r \int_z [B^{(N)}]^T \begin{bmatrix} [c^E] & [e] \\ [e]^T & -[\varepsilon^S] \end{bmatrix} [B^{(N)}] \, r\,dr\,dz \tag{13}$$

where $[B^{(N)}]$ is the matrix of "strain-displacement" coefficients for the N^{th} harmonic.

Equations (12) and (13) indicate that the matrix equation to be solved for static analysis may be partitioned as follows:

$$\begin{bmatrix} [K_{\delta\delta}] & [K_{\delta\phi}] \\ [K_{\phi\delta}] & [K_{\phi\phi}] \end{bmatrix} \begin{Bmatrix} \{\delta\} \\ \{\phi\} \end{Bmatrix} = \begin{Bmatrix} \{F_\delta\} \\ \{F_\phi\} \end{Bmatrix} \tag{14}$$

where $\{\delta\} = \lfloor u_1, v_1, w_1, \ldots, u_n, v_n, w_n \rfloor^T$

$\{\phi\} = \lfloor \phi_1, \ldots, \phi_n \rfloor^T$

$\{F_\delta\}$ = vector of structural forces

and $\{F_\phi\}$ = vector of electrical charges

In addition to the new data cards describing the piezoelectric materials, many modifications and corrections to NASTRAN were made, including the computation of complex stresses and forces for the TRAPAX and TRIAAX elements.

An example of a piezoelectric problem is shown in figure 2. This is an axially polarized PZT-4 piezoelectric disk, whose natural frequencies are to be determined. Table 2 compares the NASTRAN results with experimental and MARTSAM results (ref. 8). MARTSAM uses finite elements similar to NASTRAN's TRIAAX and TRAPAX elements, but with quadratic displacement functions rather than the linear displacement functions in NASTRAN. The MARTSAM results were obtained with a much coarser mesh.

FLUID-STRUCTURE INTERACTION

Investigation of the coupling of fluid and structural effects has been an important part of the DTNSRDC program during the past few years. Applications include vibrations of submerged structures (refs. 9 and 10), shock response of submerged structures (refs. 11 and 12), and the response of fluid-filled pipes.

Although these new applications did not require additions to NASTRAN, they did involve inventive use of DMAP and unusual use of existing data cards. This relatively new subject area shows the power of NASTRAN and its DMAP capability to adapt to new uses without requiring modification of the source code.

ISOPARAMETRIC FINITE ELEMENTS

A number of additions and modifications have been made to NASTRAN in the area of isoparametric finite elements.

1. A two-dimensional membrane element IS2D8, with quadratic displacement functions, was added to the finite element library. This element has complete NASTRAN capability with the exception of piecewise-linear analysis. The element has been used in a number of applications where the CQDMEM1 element would have required a much finer mesh.

2. The standard version of NASTRAN computes grid point stresses of the isoparametric solids IHEX1, IHEX2, and IHEX3 directly at the grid points. However, it has been shown that the stresses computed at the grid points are inferior to stresses extrapolated to the grid points from stresses calculated at the Gauss integration points (ref. 13). This extrapolation method has been added to the program for the IHEX1, IHEX2, IHEX3, and IS2D8 elements.

3. The isoparametric solid elements are limited to isotropic materials in the standard version of NASTRAN. We have added a capability for rectangular anisotropy for those elements.

4. As mentioned in the Magnetostatics section, a thermal gradient computation has been added for the isoparametric solids.

5. Although Level 17.5 allows for the choice of single precision or double precision arithmetic for some computations, including element matrix generation, it did not allow this choice for the isoparametric solids; only double precision was allowed. Since DTNSRDC uses CDC computers with 60-bit single precision words, a single-double choice for these elements was added. Generation time for the single precision stiffness matrix for one IHEX2 element with three Gauss integration points was reduced on the CDC 6400 computer from 12 CPU seconds to 4.

SHOCK DESIGN OF SHIPBOARD EQUIPMENT

The Dynamic Design-Analysis Method (DDAM) was developed for the shock design of shipboard equipment (ref. 14). This method is similar in many respects to the techniques used in earthquake analysis. In fact, an earthquake analysis using NASTRAN has been performed (ref. 15). However, DDAM, rather than some variation of it, is required by naval shipbuilding specifications for shipboard equipment. Therefore, we are presently developing a DMAP procedure and a functional module to perform DDAM analyses.

Briefly, the steps in the DDAM method are as follows:

1. Compute the normal modes and natural frequencies.
2. For each mode, the i^{th}, for example, compute the participation factor

$$P_i = \frac{1}{m_i} \{\phi_i\}^T [M]\{D\} \tag{15}$$

where P_i = participation factor for the i^{th} mode

m_i = modal mass term for the i^{th} mode = $\{\phi_i\}^T [M]\{\phi_i\}$

$[M]$ = mass matrix

$\{\phi_i\}$ = i^{th} mode shape

$\{D\}$ = direction cosine vector defining desired direction (DDAM analyzes one direction at a time)

3. Calculate the effective mass and effective weight in each mode.

$$M_i^{eff} = P_i \{\phi_i\}^T [M]\{D\} = m_i P_i^2 \tag{16}$$

$$W_i^{eff} = g\, M_i^{eff} \tag{17}$$

where

M_i^{eff} = effective mass in i^{th} mode

W_i^{eff} = effective weight in i^{th} mode

g = acceleration of gravity

4. Using the effective weights just computed, locate the design spectrum value V_i for each mode in the desired direction.

5. Compute the effective static force for each mode.

$$\{F_i\} = P_i V_i \omega_i [M]\{\phi_i\} \tag{18}$$

where ω_i is the i^{th} natural frequency.

6. Perform a static analysis with each load to compute stresses. (There will be one static analysis for each desired mode in each desired direction.)

7. Compute the so-called NRL sum (ref. 16) of the stresses at each desired point (element centroids) as follows:

$$S_j = \left| S_{jm} \right| + \sqrt{\sum_{b \neq m}^{b} (S_{jb})^2}$$

where S_{jm} = the maximum stress at the j^{th} point (taken over the modes under consideration)

Two NASTRAN runs will be required for a complete DDAM analysis; the first will perform steps 1-3, and the second, steps 5-7. The D and V terms will be input through DMI cards, although some default values will be available for V. A post-processor, possibly included in NASTRAN as a new functional module, will perform the NRL sums in step 7.

ERROR CORRECTIONS

Numerous error corrections have been made to Level 17.5 by DTNSRDC and reported to COSMIC, but perhaps the most important involved the stiffness matrix computation for the six elements QDMEM1, QDMEM2, SHEAR, TWIST, TRAPAX, and TRIAAX. The method of matrix computation for these elements was changed from SMA-type in Level 17.0 to EMG-type in Level 17.5. The change introduced an error which occurred only for certain combinations of grid point numberings for these elements.

All the error corrections reported to COSMIC are expected to appear in the forthcoming Level 17.6.

REFERENCES

1. Hurwitz, M.M.: An Addition to the NASTRAN Element Library. David W. Taylor Naval Ship Research and Development Center Report 3628, March 1971.

2. Zienkiewicz, O.C.; Lyness, J.; and Owen, D.R.J.: Three-Dimensional Magnetic Field Determination Using a Scalar Potential - A Finite Element Solution. IEEE Transactions on Magnetics, Vol. MAG-13, No. 5, Sept. 1977, pp. 1649-1656.

3. Hurwitz, M.M.; and Schroeder, E.A.: Solving Magnetostatic Field Problems with NASTRAN. Seventh NASTRAN Users' Colloquium, Oct. 4-6, 1978, Huntsville, Alabama, NASA CP-2062.

4. The NASTRAN Theoretical Manual (Level 17.5), NASA SP-221(05), Dec. 1978.

5. Jackson, J.D.: Classical Electrodynamics, Second Edition. John Wiley & Sons, Inc., New York, 1975.

6. Allik, H.; and Hughes, T.J.R.: Finite Element Method for Piezoelectric Vibration. Int. J. Num. Methods in Engr., Vol. 2, No. 2, pp. 151-157 (April-June 1970).

7. Lipman, R.R.; and Hurwitz, M.M.: Piezoelectric Finite Elements for NASTRAN. David W. Taylor Naval Ship Research and Development Center Report DTNSRDC-80/045, April 1980.

8. Allik, H.; Cacciatore, P.; Gauthier, R.; and Gordon, S.: MARTSAM 4, A Version of GENSAM, MARTSAM Users' Manual. General Dynamics/Electric Boat Division Report U440-74-043, May 1974.

9. Everstine, G.C.; Schroeder, E.A.; and Marcus, M.S.: The Dynamic Analysis of Submerged Structures. NASTRAN: Users' Experiences, NASA TM X-3278, Sept. 1975.

10. Marcus, M.S.: A Finite Element Method Applied to the Vibration of Submerged Plates. J. of Ship Research, Vol. 22, No. 2, June 1978.

11. Everstine, G.C.: A NASTRAN Implementation of the Doubly Asymptotic Approximation for Underwater Shock Response. NASTRAN: Users' Experiences, NASA TM X-3428, Oct. 1976.

12. Huang, H.; Everstine, G.C.; and Wang, Y.F.: Retarded Potential Techniques for the Analysis of Submerged Structures Impinged by Weak Shock Waves. Computational Methods for Fluid-Structure Interaction Problems, ed. by T. Belytschko and T.L. Geers, AMD-Vol. 26, ASME, New York, Nov. 1977.

13. Zienkiewicz, O.C.: The Finite Element Method, Third Edition. McGraw-Hill, London, 1977.

14. Belsheim, R.O.; and O'Hara, G.J.: Shock Design of Shipboard Equipment. NAVSHIPS 250-423-30, May 1961.

15. Pamidi, M.R.; and Pamidi, P.R.: Modal Seismic Analysis of a Nuclear Power Plant Control Panel and Comparison with SAP IV. Fifth NASTRAN Users' Colloquium, Ames Research Center, NASTRAN: Users' Experiences, NASA TM-X-3428.

16. Shock Design Criteria for Surface Ships. Naval Sea Systems Command Report NAVSEA 0908-LP-000-3010, May 1976.

Table 1. Ferromagnetic Sphere Results

DATA POINT	ANALYTIC SOLUTION		NASTRAN SOLUTION		DEVIATION	
1	0.396 TESLA	56.3°	0.396 TESLA	59.0°	0.0%	2.7°
2	0.843	52.7	0.840	53.5	0.3	0.8
3	1.488	0.0	1.537	0.0	3.3	0.0
4	0.523	86.9	0.527	87.1	0.7	0.2
5	0.941	76.5	0.921	76.3	2.1	0.2
6	0.417	79.2	0.409	80.6	1.9	1.4
7	0.571	74.6	0.579	74.5	1.5	0.1
8	0.705	83.7	0.697	83.6	1.1	0.1
9	0.479	86.7	0.483	86.8	.7	0.1
10	0.526	85.3	0.532	85.4	1.1	0.1
11	0.566	88.3	0.562	88.5	0.7	0.2
12	0.492	89.3	0.499	89.3	1.5	0.1
13	0.499	88.6	0.505	88.6	1.3	0.0
14	0.516	89.1	0.523	89.0	1.6	0.1

Table 2. Natural Frequencies of Piezoelectric Disk

Mode	Natural Frequencies (cps)		
	Experimental	MARTSAM Mesh	NASTRAN Mesh
1	22042	23298	24323
2	-----	59805	61114
3	-----	103048	104689

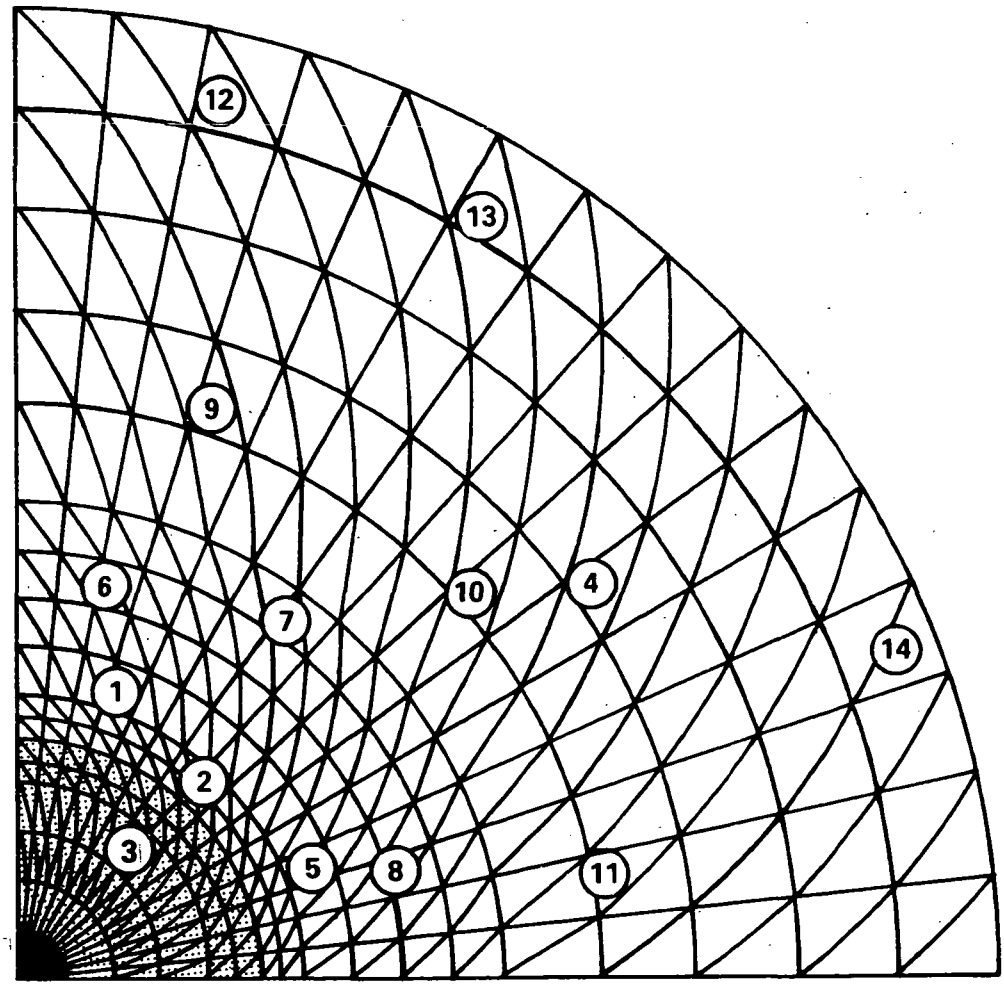


Figure 1 — Finite Element Mesh of Ferromagnetic Sphere

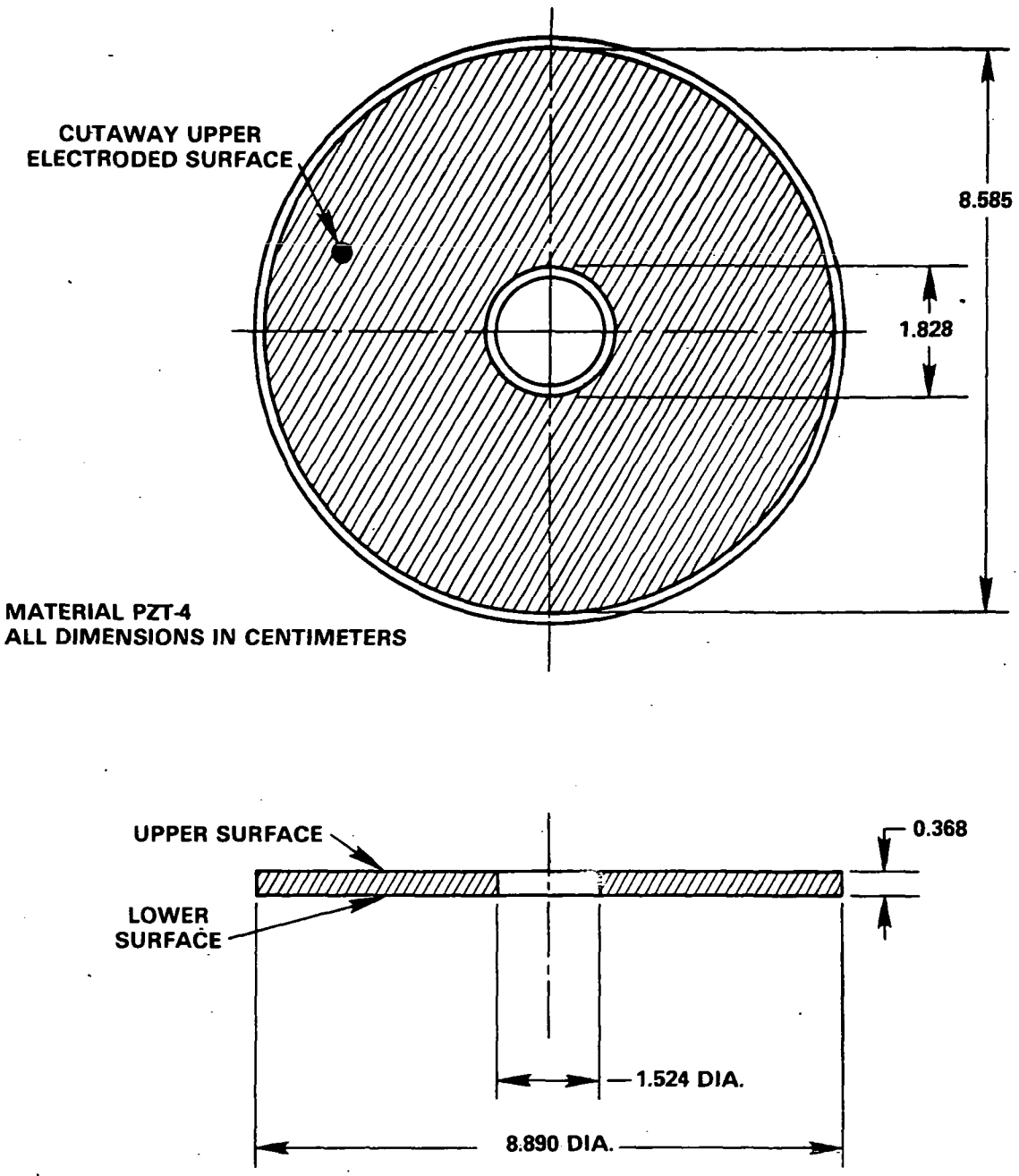


Figure 2 - Piezoelectric Disk

IMPROVEMENTS IN SPARSE MATRIX OPERATIONS OF NASTRAN

Shinichiro Harano
Hitachi, Ltd.

SUMMARY

This paper describes improvements in sparse matrix operations for the NASTRAN program achieved by Hitachi, Ltd.(Japan). To solve a large scale problem at a high speed, a great emphasis was laid on how to make a reduction in execution time needed by matrix operations, since the size of the problem depends largely on speed of matrix operations as well as on hardware and program performance. The descriptions in this paper are presented under Introduction plus five subjects: Sparse Matrix and Matrix Packing, Matrix Decomposition, Forward Elimination and Backward Substitution, Eigenvalue Extraction Methods and Parallel Processing Oriented Matrix Operations. These improvements can be applied to other versions of NASTRAN with a slight modification by using several subroutines which we have developed.

INTRODUCTION

Since the introduction of NASTRAN level 15.5.1 in 1974, we have improved it by a series of program enhancements. Highlights of them are development of the IG/OG (Input Generator/Output Generator) program to perform automatic meshing and edit the results of calculation, and addition of isoparametric elements of two dimensions, three dimensions or axi-symmetry. Dealt with in this paper is another highlight of them.

Recent drastic improvements in hardware performance have brought a gradual moving from the third generation computers, typified by IBM 370 to distributed computers, also typified by IBM 3033. Being in step with such worldwide trends, Hitachi has developed HITAC M-180 closely comparable with IBM 370/168 and HITAC M-200H providing a throughput three times that of IBM 370/168. Along with these hardware breakthroughs, the parallel processing feature appearing with vector and array processors will be increasingly brought in, changing a current software environment greatly.

Accordingly, in putting a further refined processing system for matrix operations into practice under such situation, one must direct his attentions to hardware as well as software dimensions of the breakthroughs. We have confirmed that a more effective use of a vector processor is well attainable by the Gaussian elimination of inner product type, also called "the Skyline method", which was proposed by Prof. Wilson (UCB), rather than by the conventional band matrix algorithm.

SPARSE MATRIX AND MATRIX PACKING

Generally, matrices for structural analysis are characterized by sparsity. To take full advantage of this characteristic in matrix operations, NASTRAN carries out matrix data packing. The way of matrix data packing is especially important for a problem where a large scale matrix is to be handled efficiently. So far, in transmission of matrix data between a secondary storage device and a main memory via an input/output buffer, packing routines have been used to transmit data from the input/output buffer to allow matrix data to be referenced. However, this method is less advantageous to handle a large volume of matrix data since it needs much overhead time for the transmission.

New "non-transmit" packing routine has been added to our version of NASTRAN to allow matrix data to be refered to directly from the input/output buffer. The matrix packing format obtained as a result is shown in figure 1. In this figure, the string is a set of successive non-zero terms, plus the row number and length of string ahead of these terms. Further, the number of strings in one column that are resident at one input/output buffer is given to control how to refer to matrix data resident at the buffer. Padding information is also given to adjust a word boundary for data provided in double precision. At the present, the new packing routine to perform a direct reference to the input/output buffer makes only the READ option effective. This non-transmit type of routine called string by string receives or sends:

(1) the start adress of a string in a buffer
(2) the foremost row number of a string
(3) the length of a string
(4) the instruction to show whether or not EOL (end of column detection is to be made.)
(5) type of matrix data

Use of the packing routine permits various routines for matrix handling to perform a direct reference to the input/output buffer if once they have received data addresses. The packing routine offers a buffer-by-buffer backspace feature for efficient backspacing in sequential access. Unlike a conventional backspacing that needs twice back record for a single read of one record (one column), as shown in figure 2, this feature omits overlapping of READ operation and back record, as also shown in figure 3. This feature eliminates the necessity of writing, in decomposition of a symmetric matrix, of a portion of the matrix to its upper triangular matrix from the last to the first columns of the symmetric matrix, thus saving time for generating the upper triangular matrix. Furthermore, the feature requires the writing of only a lower triangular matrix onto the secondary storage device, bringing 10 to 30% reduction in use of the disk space of the storage device.

This new matrix packing technique is fully employed in the matrix decomposition described in MATRIX DECOMPOSITION. Figure 4 reveals how the technique is superior to conventional techniques by comparisons of

packing routines to pack and unpack one column of the matrix in respect to CPU time versus non-zero term densities. The CPU time for packing/unpacking of one column is on the ordinate, while non-zero term densities are on the abscissa. The length of one column is 2000 and the whole CPU time has been obtained as the result of 300 iterations. Packing routines mutually compared in this figure are:

(1) INTPK : Clears the area for one column to zero to perform element-by-element unpacking. (This is a conventional unpacking routine.)

(2) UNPACK : Unpacks a column at one time. (This is a conventional unpacking routine.)

(3) PACK : Packs a column at one time. (This is a conventional packing routine.)

(4) INPNT : Clear the area for one column to zero and transmit string data directly from the buffer to the appropriate location of the column. (This is the new packing routine.)

Figure 4 also shows CPU time for READ and WRITE operations in case of GINO (General Input Output Routines) as additional information. Although the READ and WRITE operations may be performed irrespectively of non-zero term density of the matrix, they cannot take advantage of sparsity in case of low density due to unsatisfactory efficiency. Further, all elements including non-zero ones are written out onto the secondary storage device, making an increase in disk storage space needed.

As a result, we adopted a combination of the clearing a core space to zero and the new routine of non-transmit type to unpack columns in matrix decomposition. In short, figure 4 reveals that the new unpacking routine permits a speedup approximately 2.1 times the conventional unpacking routines in such a situation that densities of unpacked one column usually fall in the range of 0.4 to 1.0, owing to the method of matrix decomposition described below.

MATRIX DECOMPOSITION

In solving a given system of linear equations, where the coefficient matrix is a large scale sparse matrix, it is general to decompose the matrix as,

$$A = L*D*U \qquad (1)$$

where, A is the original matrix (e.g. stiffness matrix), L is a unit lower matrix, U is a unit upper matrix and D is a diagonal matrix which is usually part of the diagonal portion of L or U . The discussion below assumes that the original matrix is symmetric. To decompose the matrix, the Skyline method was used. The name of "Skyline" is derived from the fact that the contour line of column's all foremost none-zero elements is similar to a skyline. This method divides a portion enclosed in a skyline and a diagonal line into two groups whose

sizes are such that the groups are capable of being resident at a free area of a virtual storage space. The contents of these groups may be read from the secondary storage device which the results of calculation may be written out onto as needed. This is shown in figure 5.

Unlike conventional techniques, the Skyline method employs an upper triangular matrix. The diagonal matrix is generated on a diagonal terms of triangular matrix U. The original symmetric matrix is decomposed in the following algorithm.

$$u^*_{ij} = a_{ij} - \sum_{k=1}^{i-1} u_{ki} u^*_{kj} \quad (i=2, \ldots, j-1) \tag{2}$$

$$u_{ij} = u^*_{ij}/d_{ii} \quad (i=1, \ldots, j-1) \tag{3}$$

$$d_{jj} = a_{jj} - \sum_{k=1}^{j-1} u_{kj} u^*_{kj} \tag{4}$$

The method carries out (2) through (4) in succession for $j=2,\ldots,n$ where, n is the dimension of matrix A. $d_{11} = a_{11}$ is assumed. First, the algorithm for the Skyline method used for an incore routine is described with the help of the explanatory illustration of figure 6. This method employs the Gaussian elimination of inner product type, as shown in this figure. All column's elements from foremost non-zero ones to diagonal ones, including zero elements, are stored on the memory. The store address at that time is resident at array M. The address of the I-th row diagonal term is expressed by M(I). The value at the I-th row and the J-th column (position pointed by IJ) is determined by the inner product of vectors P and Q. Vector P has a length of:

$$JH = M(J) - M(J-1) \tag{5}$$

If JH=1, the diagonal terms are those obtained by matrix decomposition, and therefore, processing is skipped. Vector Q, a party of inner product with P, satisfies:

$$J > I > J-JH \tag{6}$$

Let NT be a length of the intersection of vectors P and Q. Since the length of vector Q is,

$$IH = M(I) - M(I-1) \tag{7}$$

length NT is,

$$NT = \text{Min}\{I - (J - JH), IH\} - 1 \tag{8}$$

The following processing is performed only if NT is positive. Let NS be a start address of element being involved in inner product calculation of vectors P and Q. Then, NS may be expressed as:

$$NS = M(I) - NT \tag{9}$$

The address of the I-th row and the J-th column element to which the inner product value is to be added is,

$$IJ = M(J) - (J-I) \tag{10}$$

The distance (IC) between the addresses of foremost elements of vectors P and Q is,

$$IC = IJ - M(I) \tag{11}$$

Thus, letting X be a vector storing elements on the memory, the foremost elements for inner product calculation of vectors Q and P are,

$$X(NS) \text{ and } X(NS+IC), \text{ respectively.} \tag{12}$$

The length of inner product, then, is NT. Assuming that the value of inner product between vectors P and Q is S, the I-th row and the J-th column is obtainable by:

$$X(IJ) = X(IJ) - S \tag{13}$$

By performing the calculation for all I restricted by (6), the entire J-th column may be obtained. Notice that the value obtained by (13) corresponds to that by (2). In a practical LU-decomposition, as shown by (3), each element of the J-th column must be divided the corresponding diagonal element (See (15)). The J-th row and the J-th column diagonal term is,

$$X(IJ) = X(IJ) - \sum_{K=NS}^{NE} \frac{X(K)*X(K)}{X(KD)} \qquad (I=J,\ KD=M(K)) \tag{14}$$

The last result for the I-th row and the J-th column is,

$$X(K) = \frac{X(K)}{X(KD)} \qquad (K=NS,\ \ldots,\ NE) \tag{15}$$

By carrying out the above process for $2 \le J \le n$, the upper triangular matrix of matrix A may be generated in X.

The above algorithm is well applicable to matrix calculation if all matrix elements are capable of being stored on the memory. Otherwise, matrix grouping is needed prior to implement the Skyline method.

Assume that the original matrix is such a matrix as shown in figure 7. First, this matrix is divided into some groups, each of which should not have more elements than those restricted by a core space. In the example of figure 7, the matrix is divided into four groups, each of which has not more than 15 elements. These divided groups provide the

information of table I ; headings of the table are:

(1) K(1) : the current group number
(2) K(2) : minimum group number needed by calculation of group K(1)
(3) K(3) : minimum column number in group K(1)
(4) K(4) : maximum column number in group K(1)
(5) M(J) : pointer array of the diagonal term in the J-th column

With the matrix grouped above, the algorithm of the Skyline is proceeded as follows. Symbols used in the description are:

X : open core array
IA : start address of group A element
IB : start address of group B element
IM : start address of pointer array of diagonal terms
KA(1), KB(1) : group numbers for groups A and B
KA(2), KB(2) : minimum group numbers involved in calculation of groups A and B
KA(3), KB(3) : minimum column numbers of groups A and B
KA(4), KB(4) : maximum column numbers of groups A and B

Apart from the explanation of how to determine these values, we begin our discussion with the following assumptions. The areas are already assured for groups A and B (headings are X(IA) and X(IB)) and for the address of diagonal terms (heading is X(IM)). The diagonal term addresses are set in advance. Let NEQ be the number of unknowns of a given system of linear equations, and NGP be the number of groups. All control information for NGP groups, except those for diagonal term addresses, are generated on a scratch file in advance.

Then, the following steps are repeated for P until NGP.

P = 1, ... , NGP (P : group number in current calculation)

Mutually permutated, for P≠1, are the address of A and that of B, and group information of A and that of B, that is,

$$IA \rightleftarrows IB, \quad KA(k) \rightleftarrows KB(k) \quad (k=1,\ldots,4) \tag{16}$$

Let Q be the minimum group number needed to generate group P. Then,

Q = KA(2)

For P≠Q, group Q is already on B if Q = P-1; otherwise the control information about group Q is read from the scratch file to be set to KB(k), where k=1,..,4. Then, the correlation values between groups P and Q are added to group P. This process is carried out for Q by an increment of 1 until Q = P-1. For Q = P, the correlation between P and Q becomes that between P and itself on A. After the completion of the above step for group P, the elements of it are written out onto the secondary storage device. The implementation of these procedures for individual groups (1≤P≤NGP) allows an upper triangular matrix to be generated in the column direction on the secondary storage device.

The correlation between groups P and Q is obtained as follows.

Assume that group P is on A (heading : X(IA) area), while group Q is on B (heading : X(IB) area). Let NTA be the number of columns of group P on A, and NTB be the number of columns of group Q on B. Then,

$$NTA = KA(4) - KA(3) + 1 \tag{17}$$

$$NTB = KB(4) - KB(3) + 1 \tag{18}$$

Similar to figure 6, handling of groups P and Q is shown in figure 8. Let J=1, ... , NTA be column numbers of group P on A. Then, the column number of P on the entire matrix is,

$$JJ = KA(3) + J - 1 \tag{19}$$

The length of column J is,

$$JH = M(JJ) - M(JJ-1) \tag{20}$$

where, if JJ=1,

$$JH = M(JJ) \tag{21}$$

Similarly, let I=1, ... , NTB be column numbers of group Q on B. Then, the column number of Q on the entire matrix is,

$$II = KB(3) + I - 1 \tag{22}$$

The length of column I is,

$$IH = M(II) - M(II-1) \tag{23}$$

where, if II=1,

$$IH = M(II) \tag{24}$$

Again, let NT be the length of inner product of column J in group P and column I in group Q. Then,

$$NT = \text{Min}\left\{IH,\ JH-(JJ-II)\right\} - 1 \tag{25}$$

Also, let NS be the start address involved in the inner product of column I in group Q and NE be the address of the last portion. Then,

$$NS = M(II) - NT \tag{26}$$

$$NE = M(II) - 1 \tag{27}$$

The displacement (IJ), where the inner product value is added on area A, is expressed as:

$$IJ = M(JJ) - JJ + II \tag{28}$$

If NS>NE, column I in group Q has nothing to do with the calculation for column J in group P; otherwise, inner product must be calculated. Since the start addresses of areas A and B on the open core are IA and IB, respectively, by putting,

$$IC = IJ - M(II) \tag{29}$$

the inner product is,

$$X(IJ) = X(IJ) - \sum_{K=NS}^{NE} X(IB+K-1)*X(IA+K+IC-1) \tag{30}$$

The calculation of contributions from group Q to group P is completed if the above steps are carried out for all columns in group Q.

After calculations on all groups such that,

$$KA(2) \leq Q \leq P-1 \tag{31}$$

the autocorrelation of group P itself is obtained by the same procedure as that used by previous calculation of the correlation between groups P and Q by regarding that area A is the same as area B. In this calculation, I varies within the range:

$$1 \leq I \leq J \tag{32}$$

If I=J, (30) must be replaced by the procedure below. For K such that,

$$NS \leq K \leq NE \tag{33}$$

ID = IJ - KJ + 1 is obtained. If KD = M(ID) is established, the column J is completed by (14) and (15).

The implementation of the above procedure for the range of (32) gives the autocorrelation of group P. The results are written out onto an upper triangular matrix data-block for each column.

The interchange of addresses and control information by (16) are needed for handling a succeeding group. This takes the place of moving group P completed on area A to area B simply by interchanging addresses and control information. This eliminates a data transmission, and further allows one group to be read in the core only once if the correlation between two groups affects only adjacent groups.

As already suggested, the matrix must be prepared for grouping prior to the implementation of the algorithm of the Skyline method if matrix data overflows the core space available. First, the size of each group must be determined to provide such control information as in table I. The size of a group depends upon how the open core is large at execution. Figure 9 presents an open core layout. Given the open core size (NX), the size of the memory to be allocated to one group may be obtained by

bisectioning the area excluding a working area needed. This allows up to MAXT elements of one group to be stored. By use of the new unpacking routine, only start row numbers of each matrix column are picked up to create table I information. Such information are stored on a scratch file in such a way that each group is on one record. At the same time, the addresses of diagonal terms of each group at a working area are also stored on array M. Thus, the preparation is completed.

To read group P, The unpacking method must be used in which the input/output buffer can be directly referred to from an input matrix data-block. This is also applied to reading Q, where the buffer is directly referred to from an upper triangular matrix data-block. After the completion of calculation, each group is written out following the end of the upper triangular matrix. The diagonal elements are also written out onto another output data-block for succeeding forward elimination and backward substitution.

The following are the results obtained by applying the Skyline method to practical examples. Table II gives matrix characteristics for four data with a comparision of matrix characteristics in case of the conventional band matrix method. Figure 10 compares CPU time for the band matrix method with that for the Skyline method. The Skyline method in these examples gives two cases in which a vector processor has been applied and it has not been applied. The vector processor has been also applied to the band matrix method, resulting in no improvement of CPU time. This figure, therefore, does not that case. Figure 10 reveals that the Skyline method consumes 33 to 66% of CPU time needed by the band matrix method. If the vector processor is applied to the Skyline method, this value drops to as many as 16 to 28% of the CPU time. Further, for data of 7000 degrees of freedom, the Skyline method has needed CPU time on the same percent basis as the band matrix method.

FORWARD ELIMINATION AND BACKWARD SUBSTITUTION

NASTRAN is designed to proceed forward elimination and backward substitution while retaining vectors of load terms on the memory as much as possible. As the result of matrix decomposition by the Skyline method, an upper triangular matrix is generated and diagonal terms are stored on another file. This means that forward elimination is an inner product type and that store-type calculation is to be carried out after division of load terms by diagonal elements. The procedure for solving a system of linear equations:

$$U^t DUX = B \tag{34}$$

is separated into the forward elimination process

$$U^t Y = B \tag{35}$$

and the backward substitution process

$$UX = D^{-1}Y \tag{36}$$

The forward elimination process is shown in figure 11. In this figure, $l_{i,j}$, $l_{i,j+1}$, $l_{i,j+2}$ forms a string starting with the i-th column and the j-th element of the upper triangular matrix (string length is 3 here). For this string, the following calculation is carried out.

$$b_{ia} = b_{ia} - \sum_{k=j}^{j+ST-1} l^*_{ik} * y_{ka} \qquad (a=1,2,3) \tag{37}$$

This is performed for all i-th column strings in the upper triangular matrix. Then, this procedure is repeated after setting i=i+1. Since (37) is an inner product type, high speed calculation is possible. Further, as for string's elements, the new packing routine refers to the input/output buffer, saving time for data transmission and reducing time for calling the unpacking routine due to string-by-string call unlike conventional element-by-element call. Thus, the forward elimination process is inner-product-type operation for matrix's factors decomposed by the Skyline method, while the forward elimination process of the conventional method is store-type calculation.

On the other hand, the backward substitution process begins with the generation of $D^{-1}Y$ for load term Y already generated by the forward elimination process. The process allows calculations independent of the following process because all diagonal term elements are already generated on a file as one vector on matrix decomposition. After the division of load terms by diagonal terms, backward substitution is performed by backspacing the upper triangular matrix file buffer-by-buffer. This is shown in figure 12. In the process,

$$y'_{ka} = y'_{ka} - u_{kj} * x_{ja} \qquad (k=i,i+ST-1;\ a=1,\ 2,\ 3) \tag{38}$$

is calculated for each string of each upper triangular matrix column. At that time, the non-transmit unpacking routine is called for each string, and only addresses in the input/output buffer are passed to the routine of forward elimination and backward substitution.

Figure 13 gives the results of forward elimination and backward substitution by use of data shown in table II. This figure reveals that the new method requires only 16 to 54% of CPU time needed by forward elimination and backward substitution of the conventional method. Taking it into account that the coding for both processes are not oriented to the vector processor, more satisfactory results will be expected in respect to CPU time by further improvements.

EIGENVALUE EXTRACTION METHODS

In real eigenvalue extraction methods we attemped to develop these

methods in two directions: that is, partly the speeding up of the Inverse Power method, partly the development of a simultaneous iteration method.

At first we describe the Inverse Power method. Eigenvalue extraction methods are generally divided into two groups: tracking methods (the Inverse Power method and the Determinant method) and transformation methods (the Householder method and the Givens method). Though transformation methods are able to solve rapidly an eigenvalue problem in the range of comparatively small scale problems, large scale problems are unfavourable to them. On the contrary, the Inverse Power method has been employed frequently owing to less restriction than transformation methods.

Since the Inverse Power method in NASTRAN is accompanied by movements of shift points, it needs to use iteratively matrix decomposition and FBS (forward elimination and backward substitution). Improvements that were previously mentioned were applied to the Inverse Power method, so that we could improved the CPU performance of the Inverse Power method which is two or three times as much efficient as that of the conventional Inverse Power method. Table III shows that the CPU performance of the new method without the vector processor amounts to 2.7 times that of the old one. If the vector processor is applied to the former, the CPU performance of it will be equal to about 3.3 times that of the conventional one.

Now we describe a simultaneous itetation method which is called the Jennings method. There is a problem to find q eigenvalues in ascending order from the lowest value and q eigenvectors corresponding to them for the general eigenvalue problem:

$$K \bar{x} = r M \bar{x} \tag{39}$$

where K is a symmetric matrix of positive definite type and M is a symmetric matrix of non-negative definite type. The Jennings method is useful for calculating a set of eigenvalues from the lowest value and has no weakpoint that some important eigenvalues are often missed in calculation. The algorithm is shown in figure 14. This method is different from the Subspace Iteration method on operations of orthogonalization shown in (i), (j), (k), (l).

The Jennings method needs the following input data:

(1) ND : number of eigenvalues to be extracted ($2 \leq ND \leq 90$).
(2) LMAX: the maximum number of subspace iterations (the default value is 16).
(3) IEP : convergence parameter (if IEP$<$0, then EPS=10^{IEP} ; otherwise EPS=0.0001).

The dimension "m" of subspace is decided on by

$$m = \min(n, 2q, q+8) \tag{40}$$

The selection of initial iteration vectors is most important for the

convergence of subspace and the convergence ratio is decided on by the "neighborhood" between subspace spanned by eigenvectors and subspace spanned by initial iteration vectors. Assume that

$$K = (k_{ij}) \ , \ M = (m_{ij}) \quad : k_{ij} = k_{ji} \ , \ m_{ij} = m_{ji} \tag{41}$$

Then, $k_{ii} \neq 0$ is always satisfied as K is positive definite. The matrix Go which is composed of m initial vectors will be generated as follows.

(1) At first, the first column of Go is a vector D, where
$D(I) = m_{ii}$

(2) check $k_{ii} \neq 0$, and

$$D(I) = D(I) \ / \ k_{ii} = m_{ii} \ / \ k_{ii} \tag{42}$$

(3) sort D(I) (I=1, ... , n) and select (m-1) values in decending order from the largest:

$$D(I_2) \geq D(I_3) \geq \ldots \geq D(I_m) \tag{43}$$

where the i-th vector of Go (i= 1, ... , m) is the unit vector, the i-th component of which is equal to 1.

The criterions of convergency are as follows:
(1) q eigenvalues and q eigenvectors are extracted.
(2) number of subspace iterations amounts to LMAX.
(3) there is no CPU time to execute three subspace iterations, because output of the results needs a little time.

Let the eigenvalues in the L-th iteration loop be on a vector E(I) (I= 1, ... , m). After reordering E(I) in ascending order, the eigenvalues in the (L-1)-th iteration loop are stored on a vector E*(I) (I= 1, ... , m). If E(I) satisfies the following relation,

$$\left| \frac{E(I) - E^*(I)}{E(I)} \right| < EPS \tag{44}$$

then E(I) is already converged; otherwise E(I) is not converged. If (44) holds true for all I ($1 \leq I \leq q$), the convergence will be achieved owing to criterion (1). If (44) doesn't hold true for some I and number of subspace iterations is equal to LMAX, the calculation of eigenvalues will be stopped due to criterion (2).

The orthogonalization of subspace is also important for a simultaneous iteration method. If a mass matrix M is non-positive definite and operations of orthogonalization isn't applied to subspace during iteration loops, the orthogonality of subspace will be breaking. For a eigenvalue problem with a non-positive definite mass matrix, the orthogonalization of subspace is necessary for iteration vectors to converge to eigenvectors. Consequently we adopted the Jennings method which orthogonalizes iteration vectors just after the calculation of eigenvalues in subspace. The generalized Jacobian method is adopted in the eigenvalue extraction on subspace.

Table III shows that the CPU performance of the Jennings method without a vector processor (or with it) is 4.0 (or 4.1) times as high as that of the conventional Inverse Power method. Thus, the Jennings method consumes about two-third of CPU time used by the new Inverse Power method. This fact results from the following reason. While the Inverse Power method needs several decompositions of full size matrices in every movements of shift points, the Jennings method is more efficient to solve large scale eigenvalue problems than the Inverse Power method, for the former needs only one decomposition of full size matrix and several decompositions of small scale matrices on subspace.

PARALLEL PROCESSING ORIENTED MATRIX OPERATIONS

Our vector processor adopts a pipeline system and uses a compiler system in which a FORTRAN source program is translated into a set of instructions specially for the vector processor with recource to the option active in compilation. This means that the object program generated by the compiler depends on the skillfulness of coding.

To discuss more specifically, this section presents the results of our test. In this test, we measured CPU time per single term for the length of a DO loop (string length) by carrying out three inner product type operations and one store type operation, in order to determine the basic operation in matrix decomposition. The results are shown in figure 15. As for the inner product type operations, two cases were further considered: the case where the vector processor was applied and the case where it was not applied. The examples of coding used in our test are:

(1) Complete inner product type

```
      REAL*8  A(1000), B(1000), X(ITER), SS
      DO 10 I = 1,ITER
      SS = 0.0D0
      DO 20 J = 1,LL                          ┐
   20 SS = SS + A(J)*B(J)                     ┘   Inner product loop
      X(I) = X(I) - SS
   10 CONTINUE
```

(2) Index explicit type

```
      REAL*8  X(1), SS
      DO 10 I = 1,ITER
      SS = 0.0D0
      DO 20 J = 1,LL                          ┐
   20 SS = SS + X(IA+J-1)*X(IB+J-1)           ┘   Explicit index type
      X(IC+I-1) = X(IC+I-1) - SS                  inner product loop
   10 CONTINUE
```

(3) Subroutine inner product type

```
      REAL*8  X(1), SS
      DO 10 I = 1,ITER
      SS = 0.0D0
      CALL DOTP (X(IA), X(IB), SS, LL)
      X(IC+I-1) = X(IC+I-1) - SS
   10 CONTINUE
```

```
      SUBROUTINE DOTP (A, B, SS, LL)
      REAL*8  A(LL), B(LL),  SS, S
      S = 0.0D0
      DO 10 I = 1,LL
      S = S + A(I)*B(I)
   10 CONTINUE
      SS = S
      RETURN
      END
```

(4) Store type (the three terms operation)

```
      REAL*8  X(1) ,SS
      DO 10 I = 1,ITER
      DO 20 J = 1,LL
      X(IC+J-1) = X(IC+J-1) + X(IA+J-1)*X(IB+J-1)
   20 CONTINUE
   10 CONTINUE
```

The above examples of coding are only for our test and there is no meaning in operation itself. The index ITER is the number of iteration loops and ITER = 2000 in our test. Though the store type (the three terms) operation is applicable to the vector processor by changing its indices, at that time we left it as it was, and then it was not applicable to the vector processor.

A close observation of figure 15 first exhibits that, as for the complete inner product type operation, use of the vector processor brings about an improvement in speed as much as 5.5 times that obtained by the same type of operation without the vector processor. Unfortunately, however, NASTRAN is not oriented to the way of coding for the complete inner product type, since it uses an open core as a working area. Thus, two possible ways for coding are explicit index and subroutine inner product types. Without a vector processor, the subroutine inner product type is more advantageous than the explicit index type. On the other hand, with the processor, the former is less advantageous than the latter.

Further, figure 15 reveals that, with the vector processor, the explicit index type almost keeps in step with the complete inner product type in respect to CPU time. However, without the processor, the former has consumed CPU time as much as 2.2 times that the latter has consumed. For a longer inner product loop, the subroutine inner product type

without the vector processor is more advantageous than the explicit index type without it. This is attributable to that there is a difference in optimization level between the operation with the vector processor and that without it. The subroutine inner product type is advantageous if the subroutine's overhead time can be overridden due to a long DO loop; otherwise, it is less advantageous than other types in speed. The result of the store type operation is also exhibited in this figure; this type does not enjoy the maximum benefits of optimization.

An observation of figure 15 also shows that the extent of optimization in various types of operations depends largely on program coding. Of course, it is ideal that the maximum optimization is always possible for any type of operation; however, the extent of optimization varies depending upon type of FORTRAN. Accordingly, in coding the algorithm for the Skyline method of matrix decomposition, we adopted the explicit index type if use of a vector processor was possible; otherwise, we used the subroutine inner product type. In the future we intend to use the explicit index type as long as the optimization feature of FORTRAN is satisfactorily refined.

So far, the inner product type has been more advantageous than the store type in respect to speed thanks to use of registers. However, the advent of a vector or array processor is changing this situation. Actually, in case of HITAC M-200H, the latter has displayed almost the same performance as the former. Further improvements of the parallel processing systems may reverse the superiority of the inner product type to the store type.

As already described, CPU time needed for the store type, inner product type operations accompanied with or without data transmission depends largely on how to make a program. Use of a higher speed computer and parallel processing system is greatly expected to change a current software environment to a large extent. Technological breakthroughs of software and of hardware would interact more closely in improving sparse matrix operations.

CONCLUDING REMARKS

In this paper we discussed about improvements in sparse matrix operations of NASTRAN. Recent advance of parallel processing systems has been changing surroundings in software. Especially, a vector processor attached to a general-purpose computer is favorable to a long DO loop operation. For example, the Skyline method which we have developed this time in the field of matrix triangular decomposition conforms to the pipeline control feature observed in the vector processor. On the contrary, the conventional band matrix method or the wavefront method which adopt store type operations don't adapt themselves to the pipeline control system, for they need complicated indices operations and are difficult to deal with a set of arithmetic data as vectors.

What is more, the way of packing/unpacking and the method of forward elimination and backward substitution were conformed themselves to the Skyline method, so that the CPU time for solving a problem was reduced by half. Further, in real eigenvalue extraction we have improved the CPU performance of the Inverse Power method and added the Jennings method to NASTRAN. The Jennings method is more effective in many cases than the new Inverse Power method.

REFERENCES

1. McCormick, C. W.: Sparse Matrix Operations in NASTRAN. Proc. of 1973 Tokyo Seminar on FEA, 1973, pp. 611-631.

2. McCormick, C. W.: The NASTRAN Program for Structural Analysis. Proc. 2nd U.S.-Japan Seminar, 1972, pp. 551-571.

3. McCormick, C. W.: Application of Partially Banded Matrix Method to Structural Analysis, Sparse Matrix Proceedings, IBM T. Watson Research Center, 1968, pp. 155-158.

4. E. L. Wilson and H. H. Dovey: Solution or Reduction of Equilibrium Equations for Large Complex Structural Systems. Lecture Notes Part 1, SAP Conference, Tokyo, 1978, pp. 13-24.

5. E. L. Wilson: An eigensolution strategy for the Dynamic Analysis of Large Structural Systems. Lecture Notes Part 2, SAP Conference, Tokyo, 1978, pp. 1-19.

6. NASTRAN User's Manual/ Theoretical Manual.

7. Melosh, R. J. and Bamford, R. M.: Efficient Solution of Load-Deflection Equations. Proc. American Society of Civil Engineers, 95, ST4, 1969, pp. 661-676.

8. Irons, B. M.: A Frontal Solution Program for Finite Element Analysis Int. J. Num. Meth. Engng., vol 2, 1970, pp. 5-32.

9. K. J. Bathe and E. L. Wilson: Numerical Method in Finite Element Analysis. Prentice-Hall. Inc., 1976.

10. K. J. Bathe: Solution for Eigenvalue Problems in Structural Mechanics. Int. J. Num. Meth. Engng., vol 6, 1973, pp 213-226.

11. Jennings, A.: A Direct Method for the Solution of Large Sparse Symmetric Simultaneous Equations. Large Sparse Set of Linear Equations, REID, Academic Press, 1971, pp. 97-104.

12. H. Rutishauser: Computational Aspects of F. L. Brauer's Simultaneous Iteration Method. 1969

13. Jennings, A. and D. R. L. Orr: Application of the Simultaneous Iteration Method to Undamped Vibration Problems. Int. J. Num. Meth. Engng., vol 3, 1971, pp. 13-24.

14. Gupta, K. K.: Recent Advances in Numerical Analysis of Structural Eigenvalue problems. Proc. of 1973 Tokyo Seminar on FEA, Univ. of Tokyo Press, 1973, pp. 249-271.

15. Y. Yamamoto and H. Ohtsubo: Subspace Iteration Accelerated by the Use of Chebyshev Polynomials for Eigenvalue Problems with symmetric Matrices. Int. J. Num. Meth. Engng., vol 10, 1976, pp. 935-944.

16. James L. Rogers, Jr.: The Impact of Fourth Generation Computers on NASTRAN. NASTRAN User's Experiences, NASA TM X-3428, 1976, pp. 431-447.

17. Control Data Corporation: Study of the Modification Needed for Efficient Operation of NASTRAN on the Control Data Corporation STAR-100 Computer. NASA CR-132644, 1975

18. Universal Analytics, Inc.: Feasibility Study for the Implementation of NASTRAN on the ILLIAC IV Parallel Processor. NASA CR-132702 1975.

TABLE I.- GROUP CONTROL INFORMATION

Group	K(1)	K(2)	K(3)	K(4)	M(J)					
Group 1	1	1	1	6	1	3	6	8	12	15
Group 2	2	1	7	9	4	7	14			
Group 3	3	2	10	11	4	8				
Group 4	4	1	12	12	11					

K(1) : Current group number

K(2) : Minimum group number needed by calculation of group K(1)

K(3) : Minimum column number in group K(1)

K(4) : Maximum column number in group K(1)

M(J) : Pointer array of the diagonal term in the J-th column

TABLE II. MATRIX CHARACTERISTICS OF EXAMPLE PROBLEMS

Data name	Nodes	Elements	Total degree of freedom	Band matrix method			Skyline method					Memory size(kB) used
				B	C	R	Divisions of group	Group operation counts	Group read counts	Total columns read	Average length of column	
DATA1	394	352	2082	84	253	83	9	38	36	8328	176	1024
DATA2	489	477	2127	151	47	150	7	13	7	2127	128	1024
DATA3	611	2435	2769	93	288	92	13	66	63	13419	201	1024
DATA4	799	892	4474	156	0	155	18	35	18	4474	147	1024

TABLE III. CPU TIME: OLD, NEW INVERSE POWER METHODS AND JENNINGS METHOD

Data name	Total degree of freedom	Extraction method	Vector processor	Number of eigenvalues to be extracted	CPU time (sec)	
					Total	Eigenvalue extraction
Data 1	2082	Old Inverse Power method	Can not be applied	10	1864	1813
Data 1	2082	New Inverse Power method	Yes	10	344	293
Data 1	2082	Jennings method	Yes	10	241	191
Data 2	2127	Old Inverse Power method	Can not be applied	10	1105	1062
Data 2	2127	New Inverse Power method	No	10	398	345
Data 2	2127	New Inverse Power method	Yes	10	332	278
Data 2	2127	Jennings method	No	10	276	232
Data 2	2127	Jennings method	Yes	10	266	222

FIGURE 1. MATRIX PACKING FORMAT

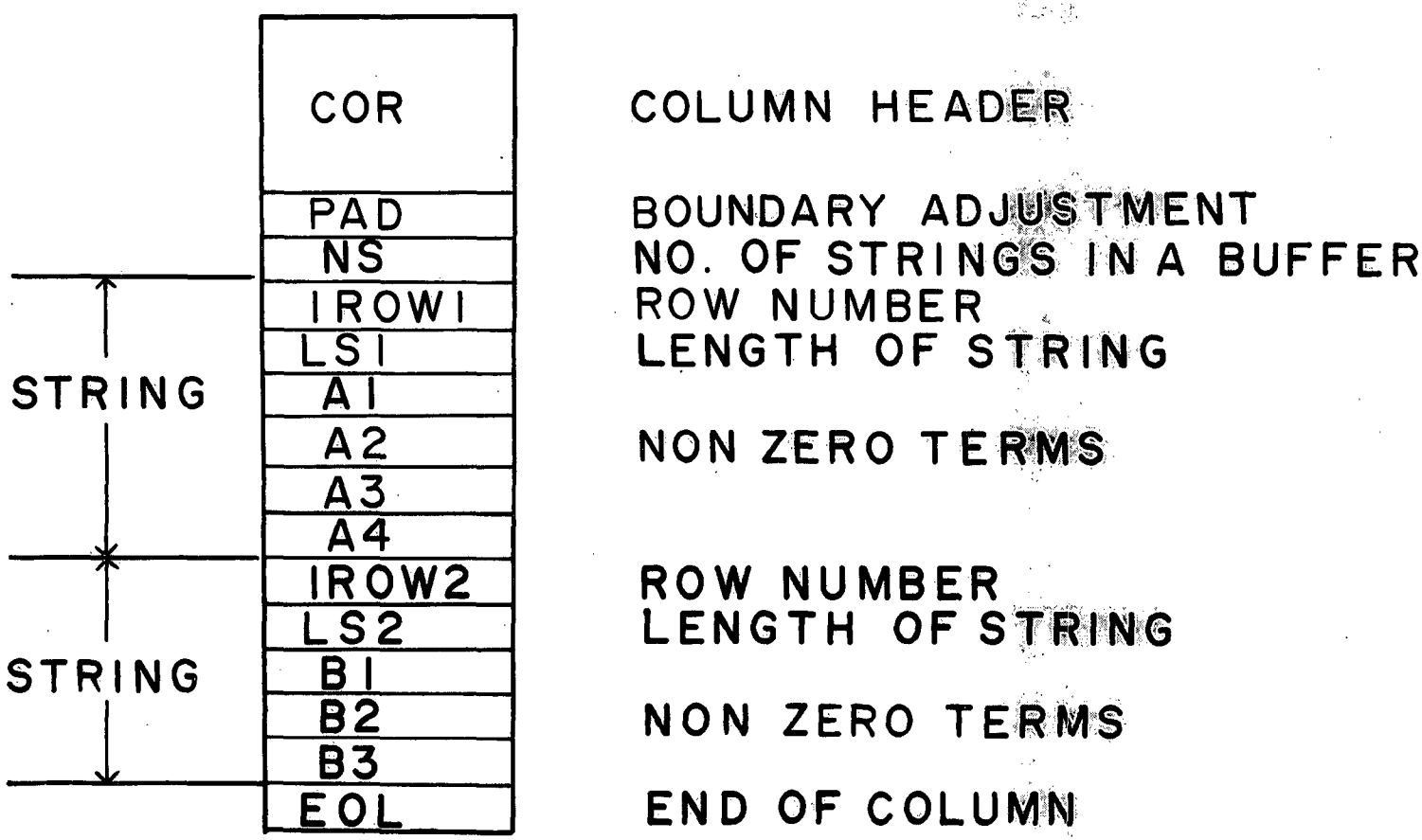


FIGURE 2. CONVENTIONAL BACKSPACING

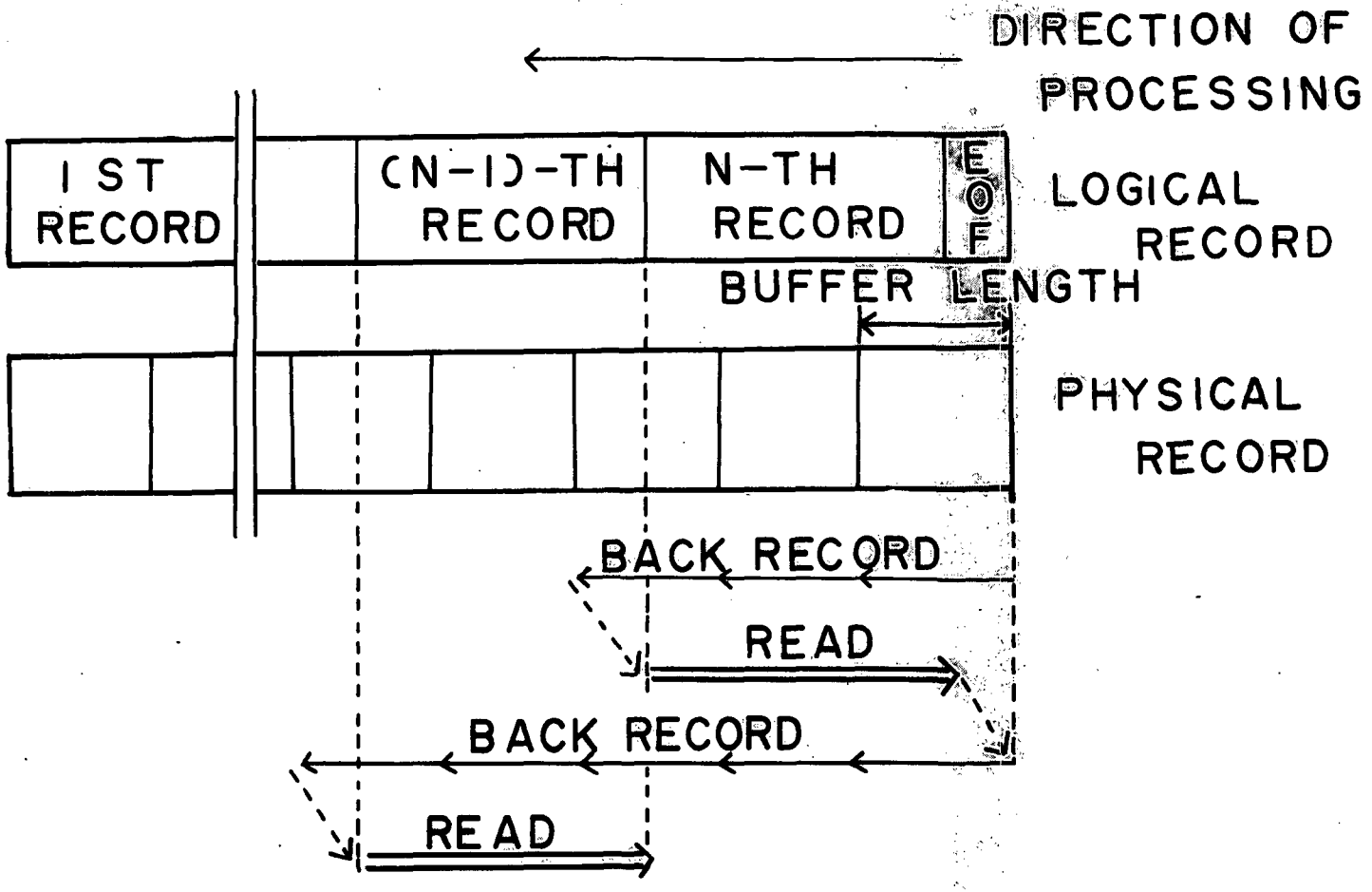


FIGURE 3. BUFFER-BY-BUFFER BACKSPACING

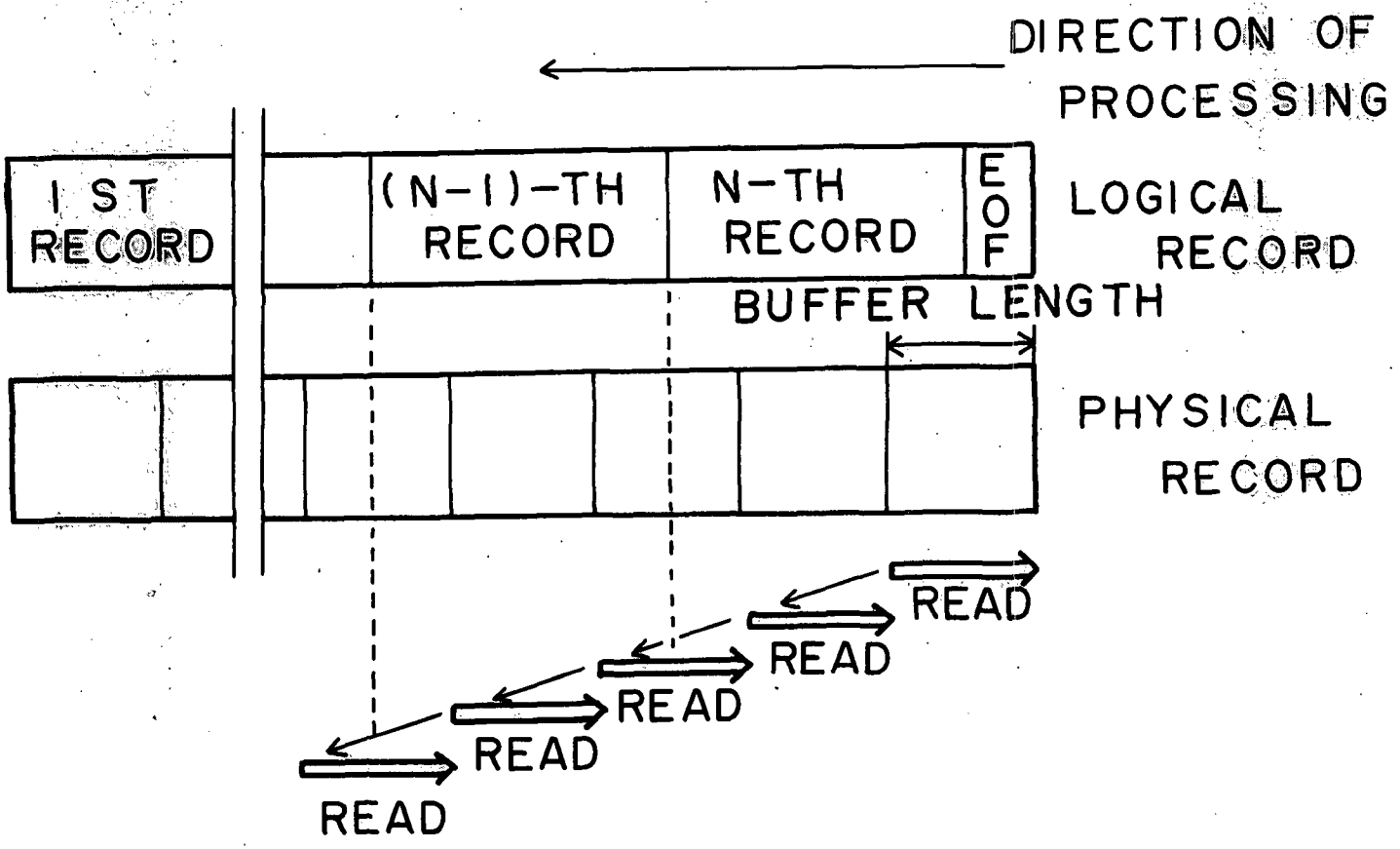


FIGURE 4. CPU TIME FOR PACKING/UNPACKING (2000-DIMENSION VECTOR : M-200H)

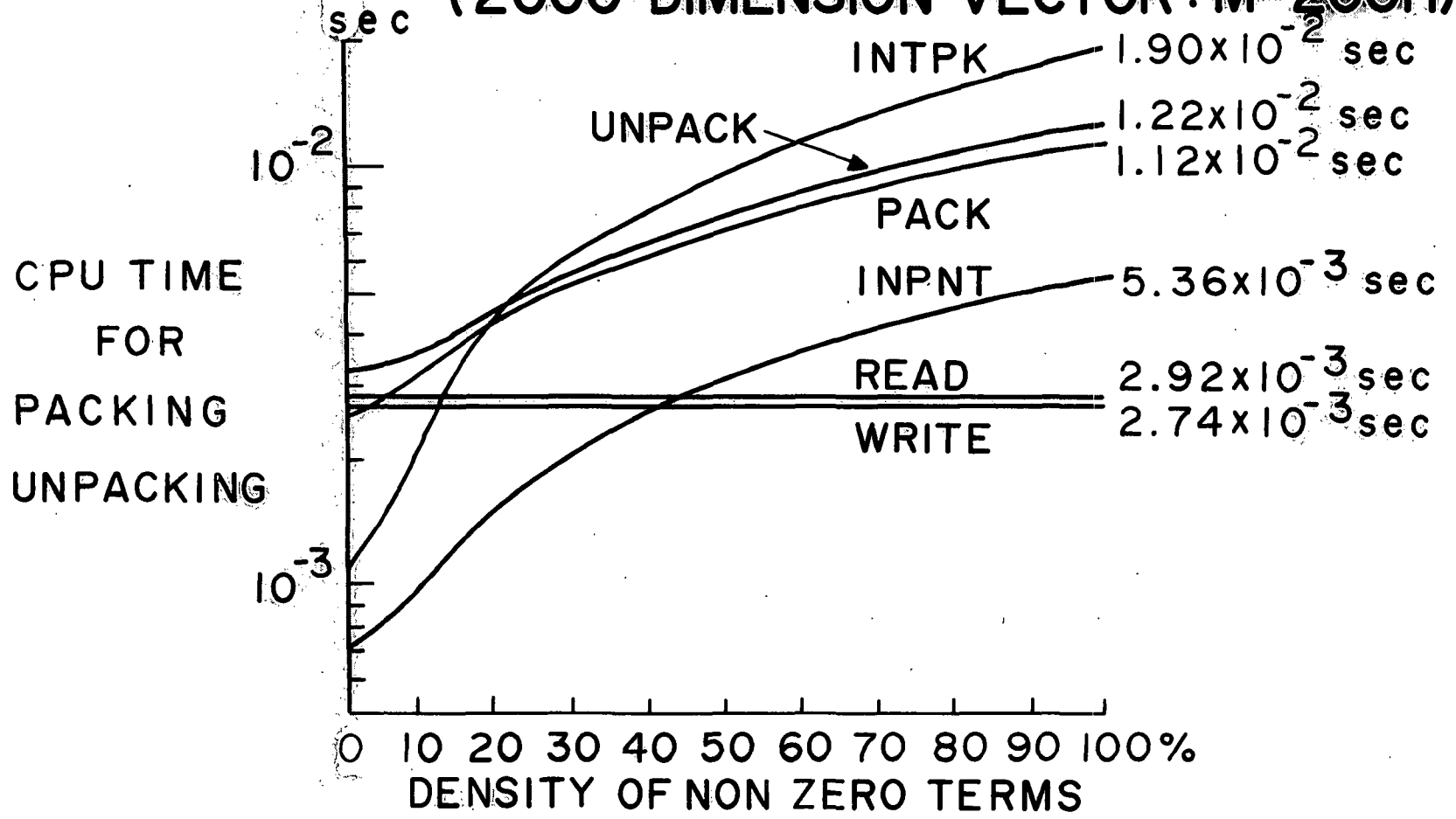


FIGURE 5. MATRIX DECOMPOSITION

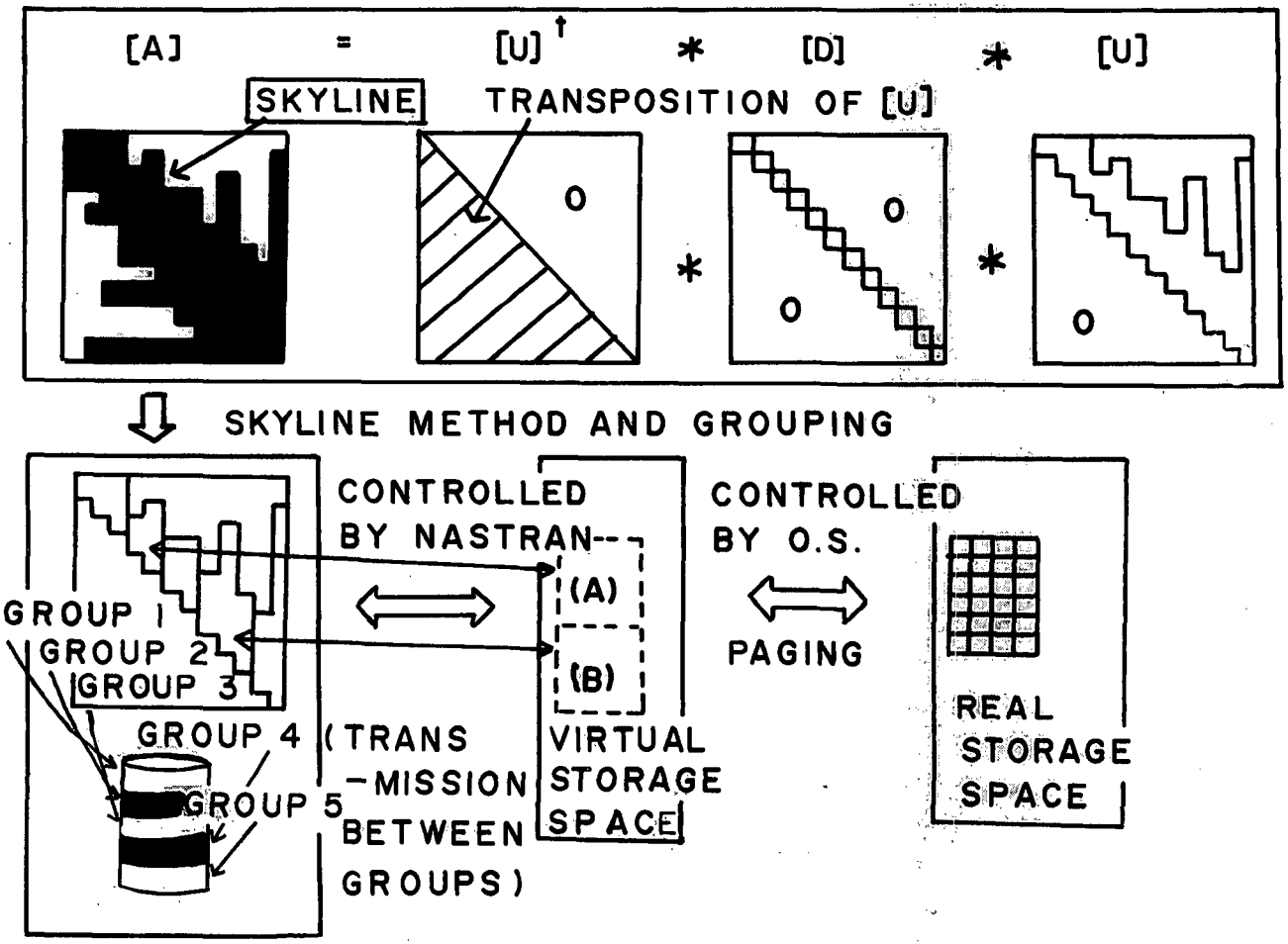


FIGURE 6. ALGORITHM FOR THE SKYLINE METHOD (INCORE TYPE)

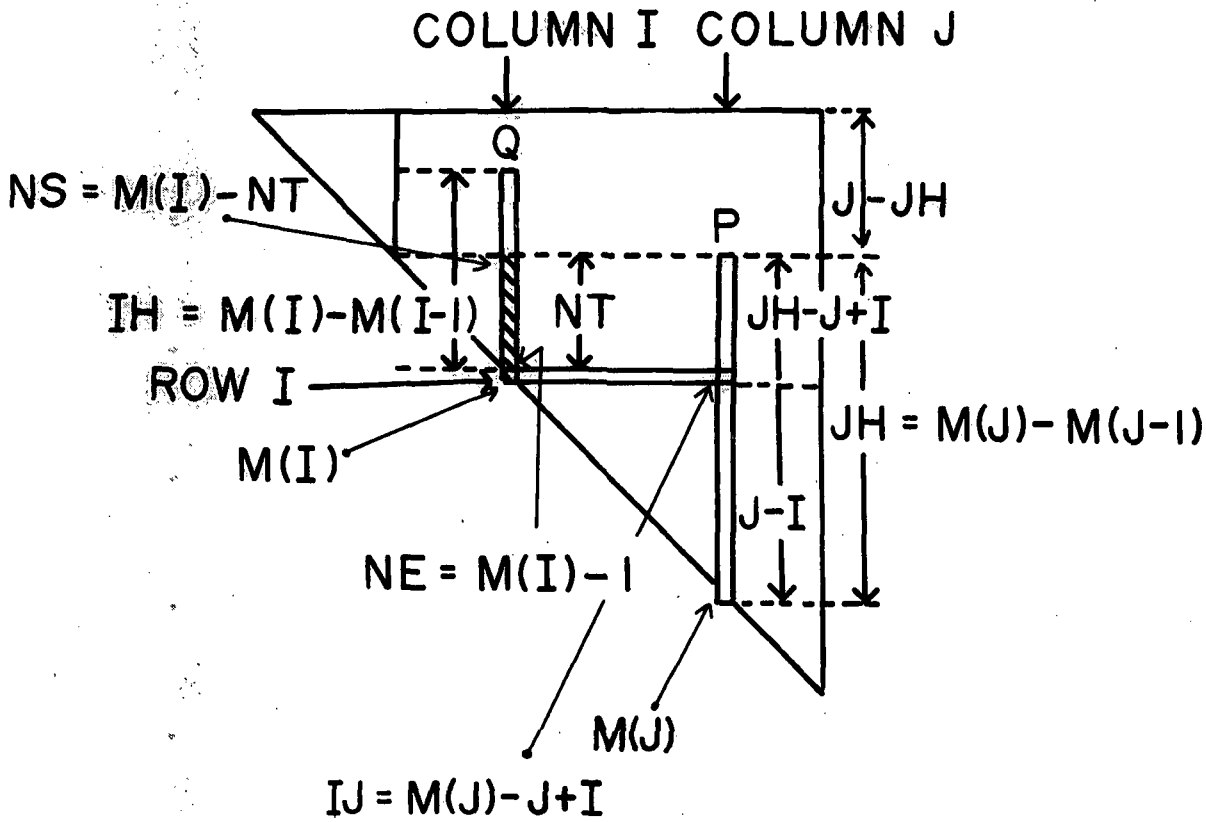


FIGURE 7. MATRIX GROUPING

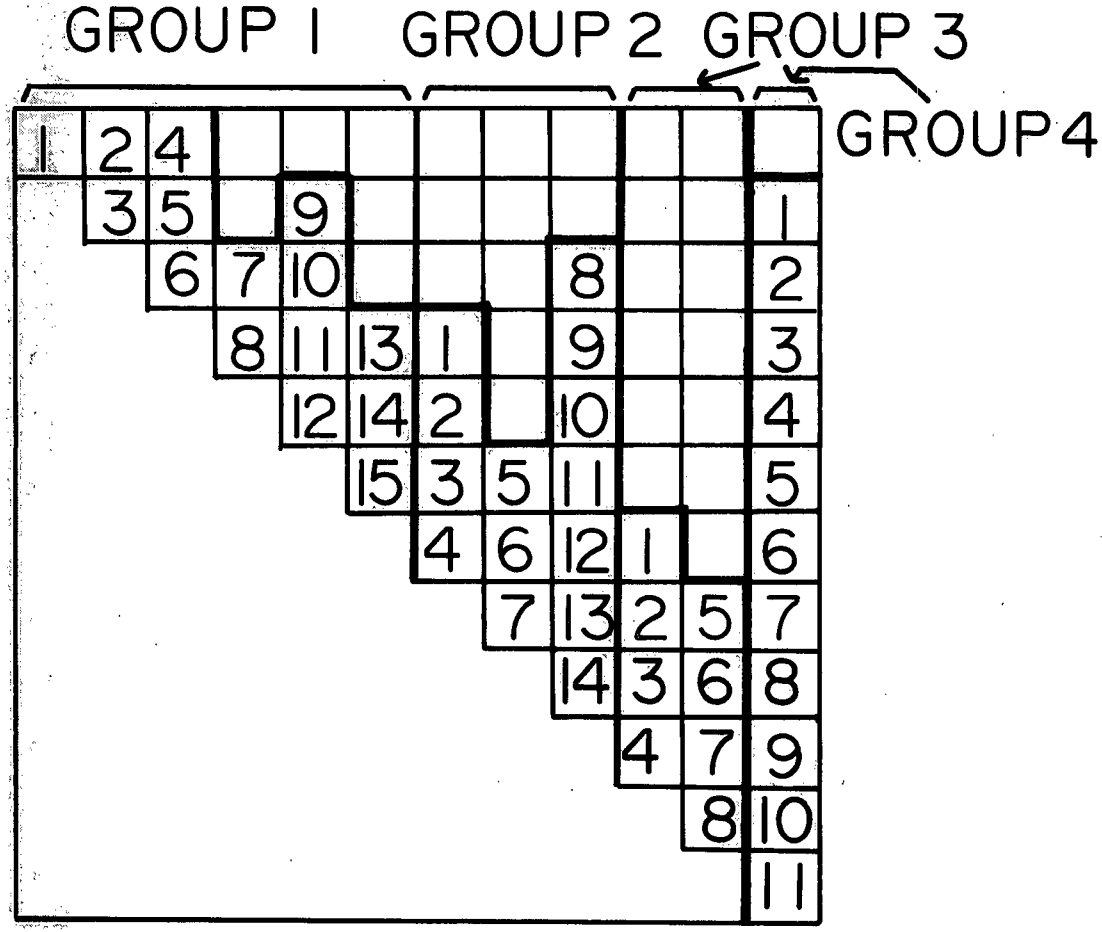


FIGURE 8. SKYLINE METHOD WITH GROUPING

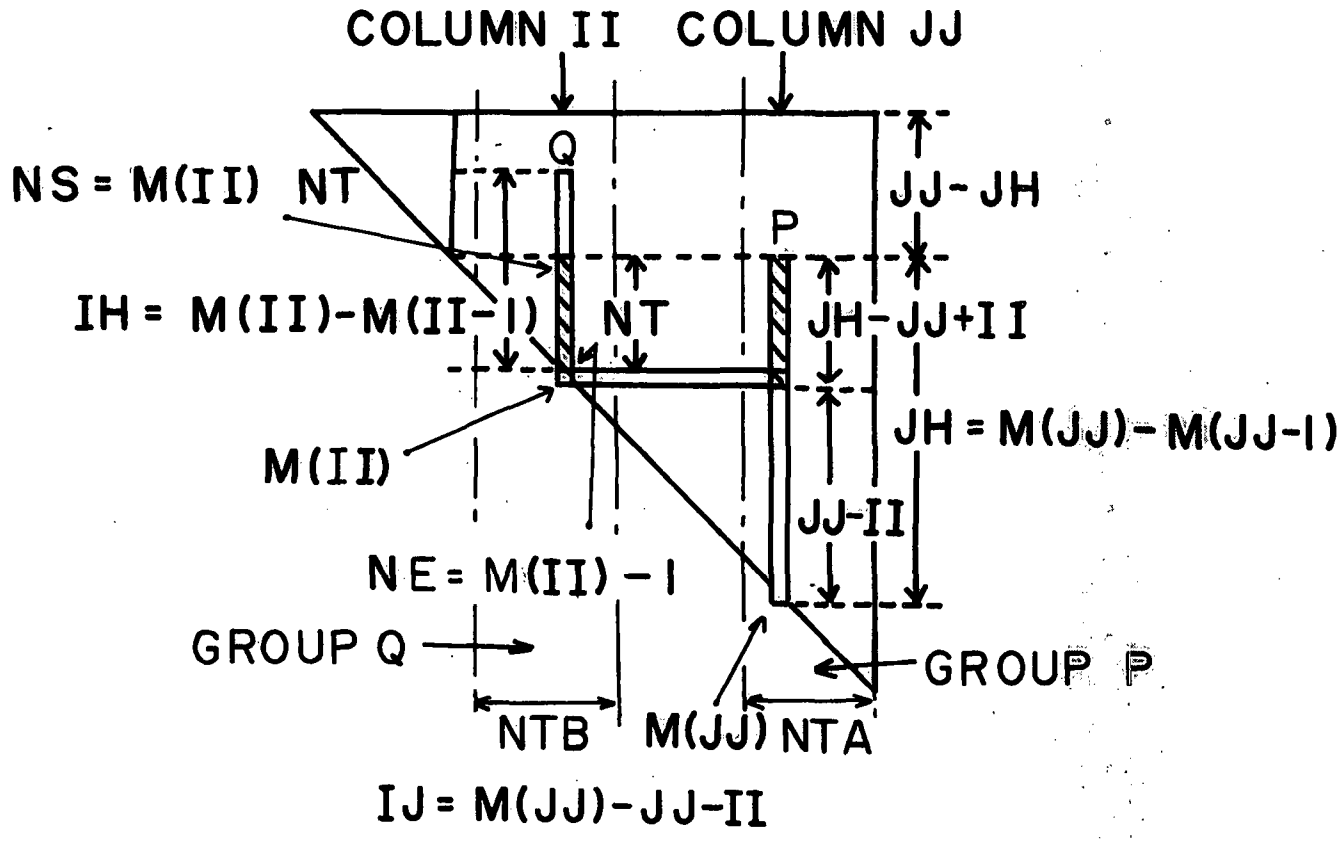


FIGURE 9. OPEN CORE LAYOUT IN SKYLINE METHOD

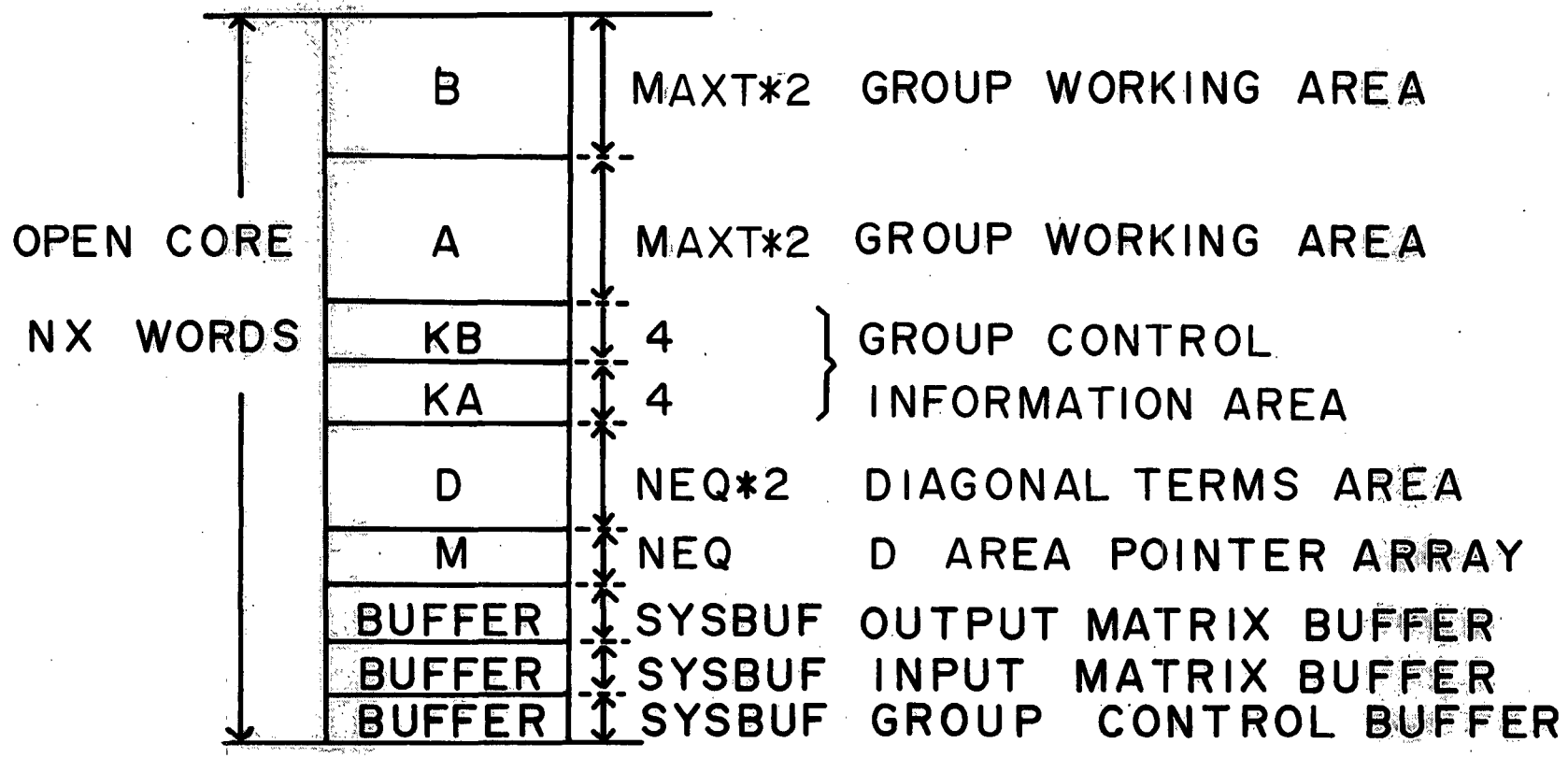


FIGURE 10. CPU TIME: SKYLINE METHOD AND BAND MATRIX METHOD

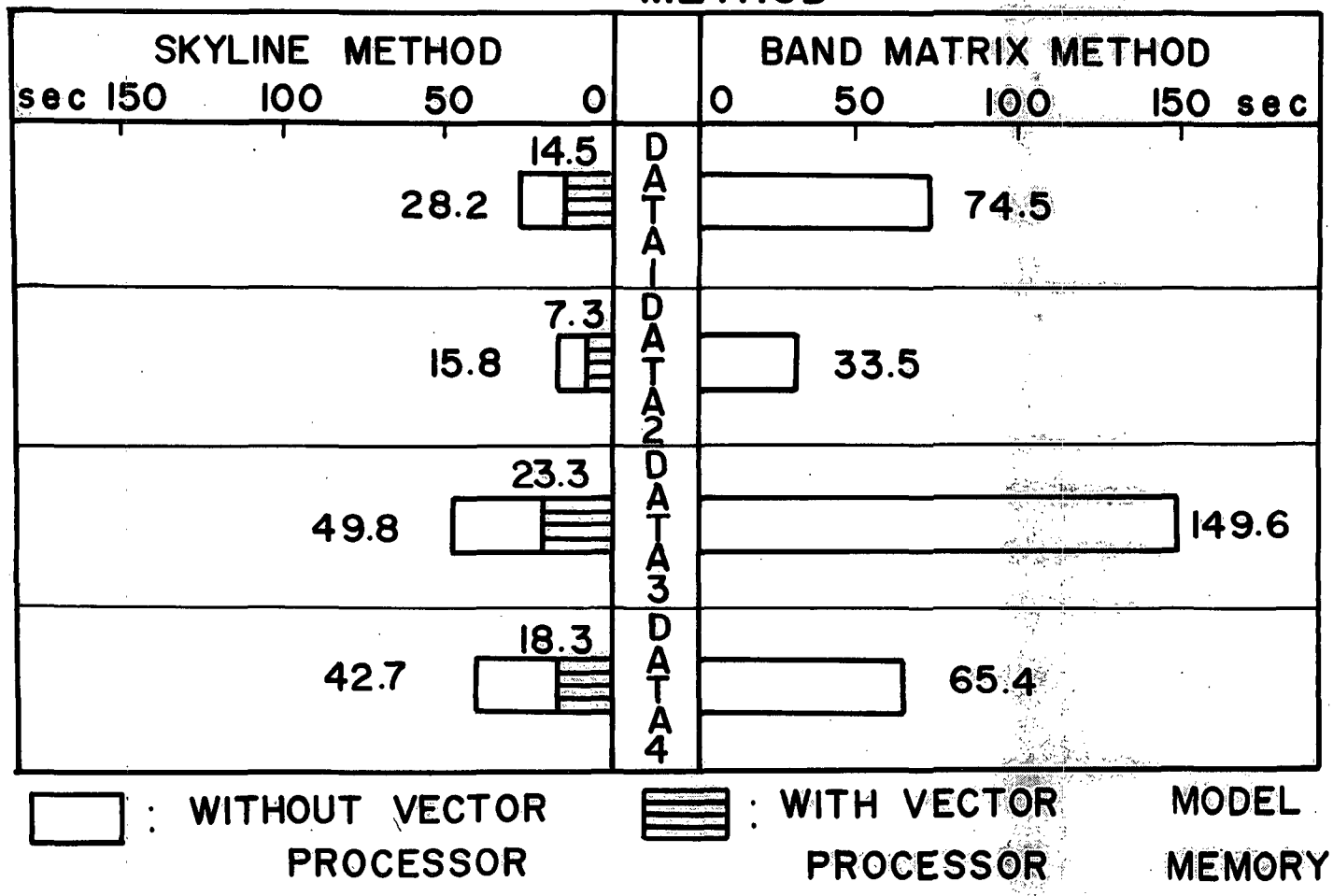


FIGURE 11. FORWARD ELIMINATION PROCESS

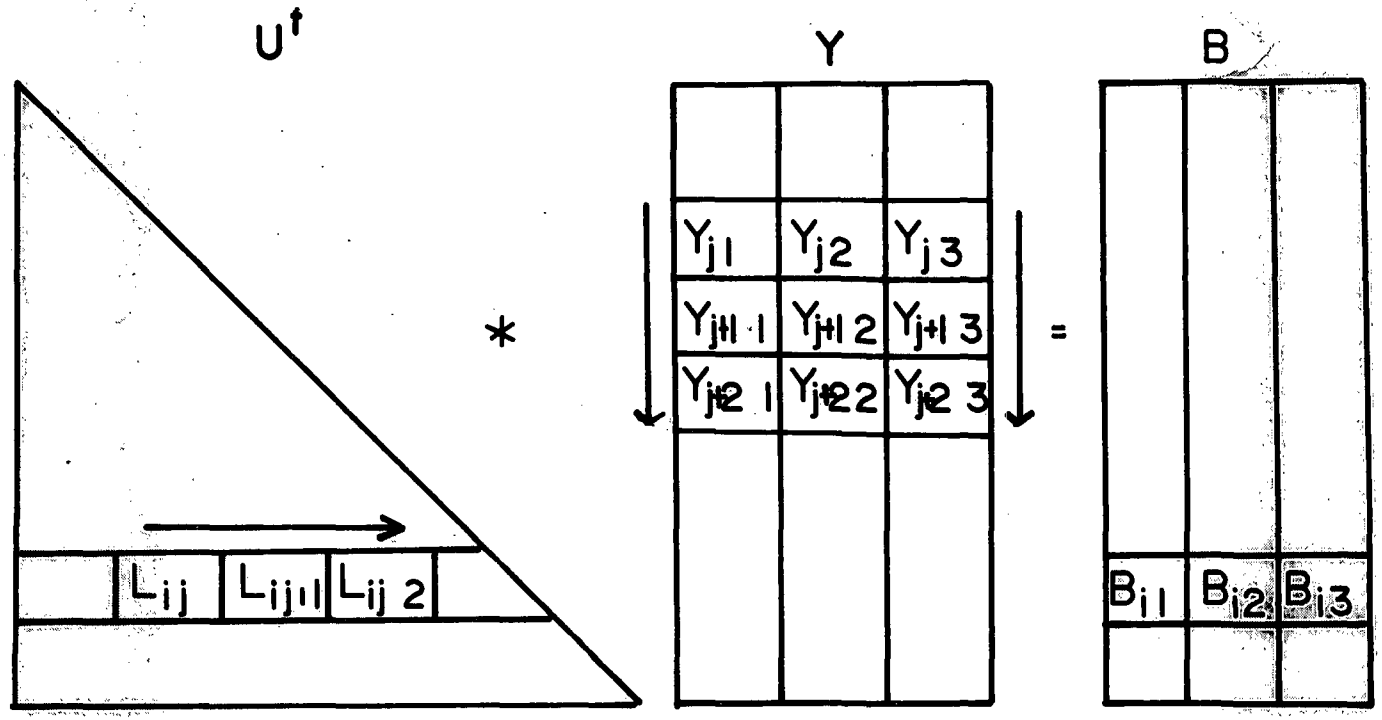
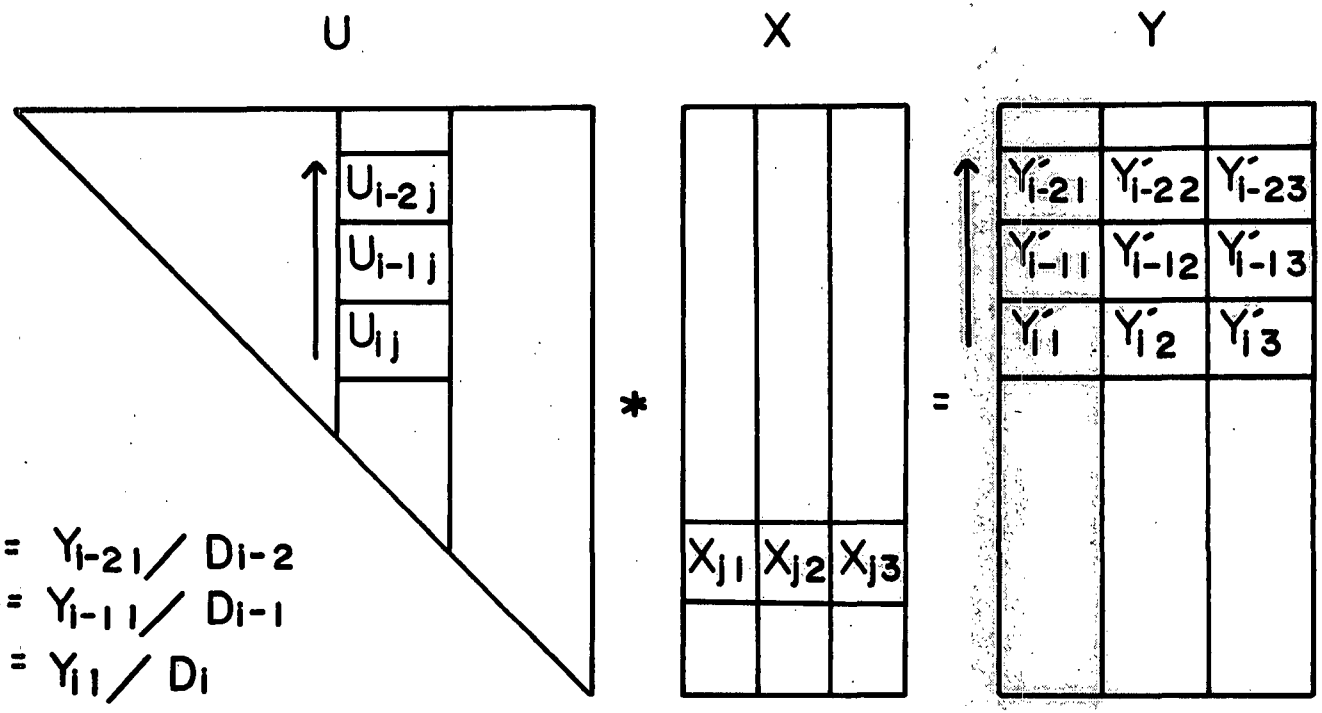


FIGURE 12. BACKWARD SUBSTITUTION PROCESS



$Y'_{i-2\,1} = Y_{i-2\,1} / D_{i-2}$

$Y'_{i-1\,1} = Y_{i-1\,1} / D_{i-1}$

$Y'_{1\,1} = Y_{1\,1} / D_{1}$

FIGURE 13. CPU TIME: NEW STRING-BY-STRING METHOD AND CONVENTIONAL FBS

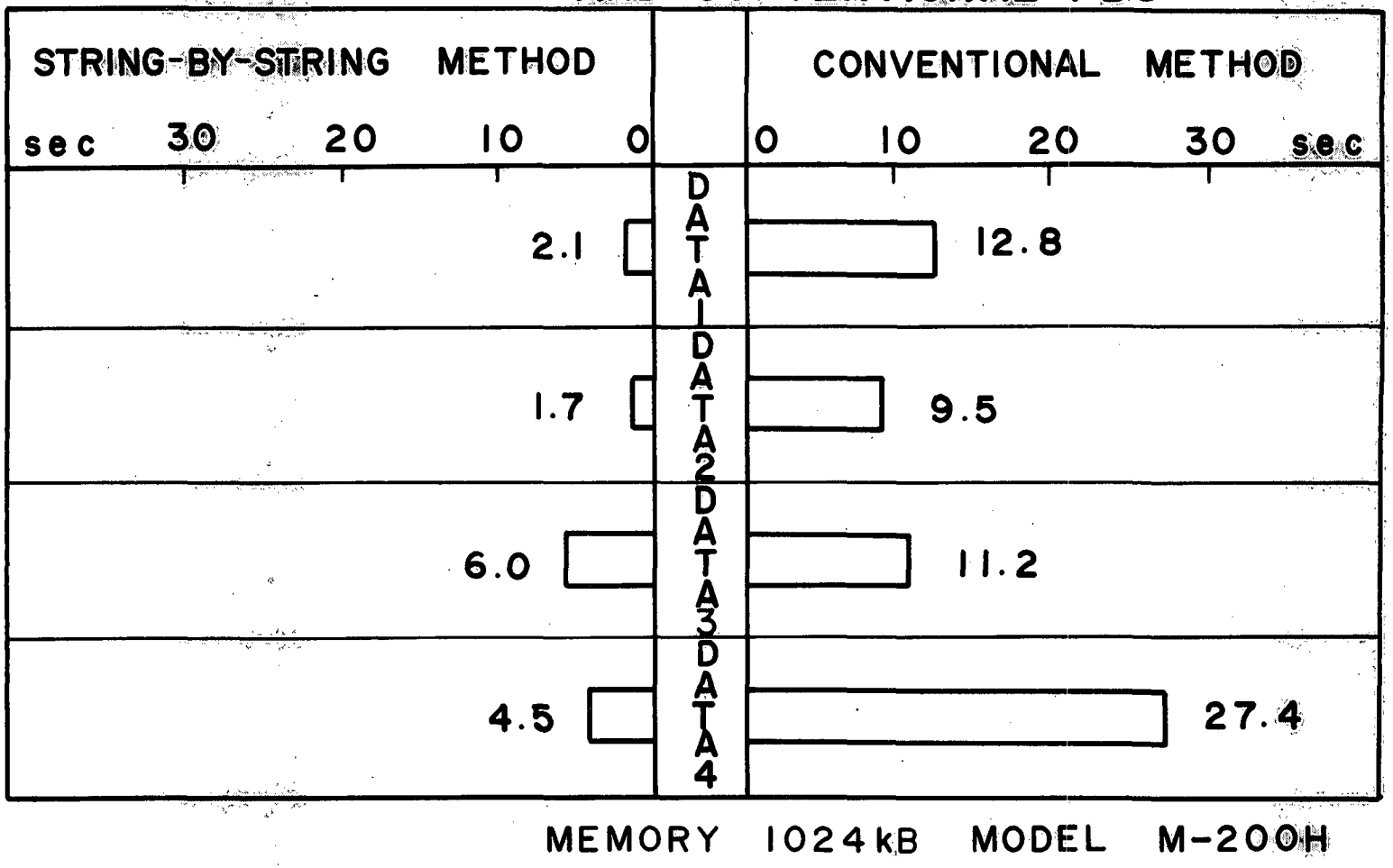


FIGURE 14. ALGORITHM FOR JENNINGS METHOD

(a) DETERMINATE THE DIMENSION OF SUBSPACE $m = \min\{n, 2q, q+8\}$

(b) CHOOSE INITIAL VECTORS $[G_0]\,(n \times m)$ AND $[G] = [G_0]$

(c) $[K][Z] = [G]$ SOLVE $[Z]\,(n \times m)$

(d) $[\bar{K}] = [Z]^{\dagger}[G]$ CALCULATE $[\bar{K}]$, $[\bar{M}]$

(e) $[\bar{M}] = [Z]^{\dagger}[M][Z]$

(f) $[\bar{K}][Q] = [R][\bar{M}][Q]$ SOLVE EIGENVALUE PROBLEM ON SUBSPACE

(g) SORT EIGENVALUES IN ASCENDING ORDER AND GO TO (m) IF q EIGENVALUES FROM THE LOWEST ARE FOUND

(h) LET $[P]\,(m \times m)$ BE A MATRIX COMPOSED OF EIGENVECTORS CORRESPONDING TO SORTED EIGENVALUES

(i) $[G'] = [M][Z][P]$

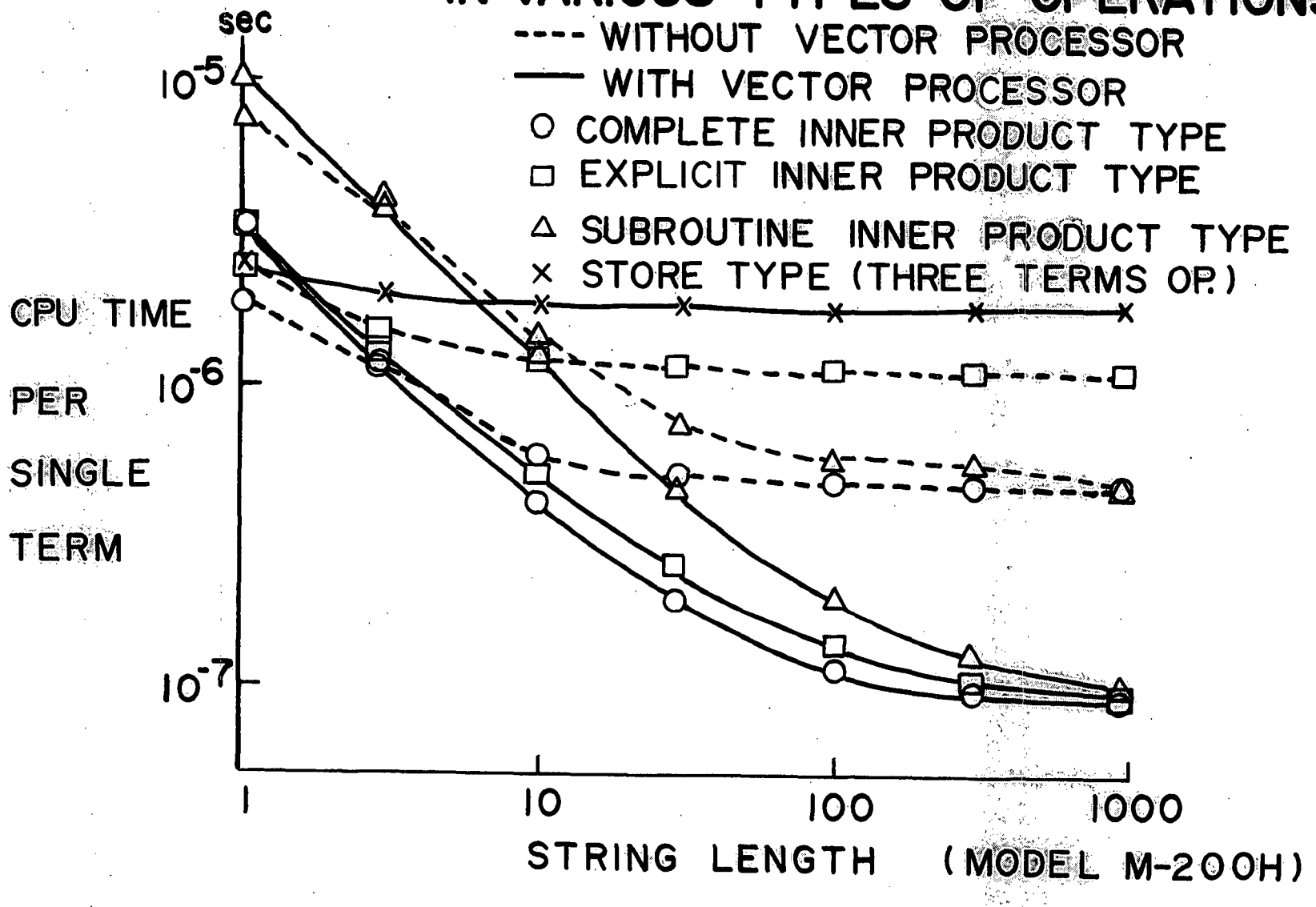
(j) $[S] = [G']^{\dagger}[G']$

(k) $[S] = [U]^{\dagger}[U]$ $U^{\dagger}U$-DECOMPOSITION

(l) $[G] = [G'][U]^{-1}$

(m) WRITE OUT EIGENVALUES AND VECTORS TO OUTPUT FILE

FIGURE 15. CPU TIME VERSUS STRING LENGTH IN VARIOUS TYPES OF OPERATIONS



SOLUTION SENSITIVITY AND ACCURACY STUDY OF NASTRAN FOR LARGE DYNAMIC PROBLEMS INVOLVING STRUCTURAL DAMPING

A. J. KALINOWSKI

NAVAL UNDERWATER SYSTEMS CENTER

SUMMARY

This paper is concerned with both the solution sensitivity and solution accuracy of large dynamic problems involving NASTRAN SOLUTION 8 (i.e., the steady state dynamic response option wherein all response quantities vary as $e^{i\omega t}$, where ω is the driving frequency and t is time). Using a submerged steel plate with a viscoelastic layer as the bench mark sample, the solution sensitivity and solution accuracy is checked. The solution sensitivity is examined by running the same finite element model on different computers, different versions of NASTRAN, and different precision levels. The solution accuracy is evaluated for these same runs by comparing the NASTRAN results with the exact solution of the same problem.

SYMBOLS

$[B]$	Damping Matrix
c_1	Dilational Wave Speed in Fluid
$\{F\}$	Applied Force Vector
$[K]$	Stiffness Matrix
$[\bar{K}]$	Modified Complex Stiffness Matrix
$[M]$	Mass Matrix
k	Wave Number (ω/c_1)
P_i	Incident Fluid Pressure
P_o	Plane Wave Amplitude
P_B	Back Side Fluid Pressure
P_S	Front Side Fluid Pressure (Scattered Component)
t	Time
$\{U\}$	Solution Displacement Vector

SYMBOLS (Cont'd)

x	Spatial Coordinate
ω	Driving Frequency
$\{\Delta\}$	Residual Solution Vector
λ^r, μ^r	Real Elastic Lame' Constants
λ^i, μ^i	Corresponding Viscoelastic Constants
ρ	Material Mass Density

INTRODUCTION

This paper is concerned with the solution accuracy of 1, 2, or 3-dimensional steady state (time harmonic) structural and/or continuum problems whose response quantities all vary in time in proportion to $e^{i\omega t}$. The linear equations of motion for such problems usually reduce to an expression of the form

$$\underbrace{\left[-\omega^2[M] + i\omega[B] + [K]\right]}_{[\bar{K}]}\{U\} = \{F\} \qquad (1)$$

where [M], [B], and [K] denote the mass, damping and stiffness matrices (MDD, BDD and KDD using usual NASTRAN DMAP notation), ω is the driving frequency and {F} are the applied forces. The results presented in this paper focus on continuum type (e.g., figure 1) applications with structural damping, however, once the form of equation (1) has been constructed, the solution becomes a matter of solving large banded symmetrical systems of complex linear algebraic simultaneous equations. Clearly, such equations can also be the end point resulting from many other NASTRAN steady state formulations, either from direct structural formulations or from related fields through analogies. Thus comparisons of solution accuracy, run time, etc. can be viewed and interpreted in a more general vein than simply applying only to problems of the type depicted in figure 1.

The motivation for this comparative study resulted as a consequence of obtaining some unexpected results on some solution 8 (steady state time harmonic rigid format) problems similar to the one shown in figure 2, except for the fact that the initial model had inclusions throughout the rubber thus making analytical solutions to the problem unwieldy.

PARAMETRIC STUDY MODEL

In order to better understand the accuracy limitations of the results of the initially more complicated inclusion filled model, a simpler homogeneous layered model (figure 2) was constructed and physically corresponds to a totally submerged 2.0" steel plate with a 3.05" viscoelastic rubber layer glued to the steel surface. The input corresponds to an incident pressure wave

$$P_i = P_o\, e^{i(kx + \omega t)},\quad k = \omega/c_1 \qquad (2)$$

where x is the horizontal coordinate along the line of propagation, c_1 is the dilatational wave speed in the fluid, and P_o is the plane wave amplitude. The exact analytical solution to this problem is known (ref. 1), consequently an accuracy check on the finite element solutions is available. Clearly, the figure 1 model is a spatially one-dimensional problem, consequently the corresponding finite element model need only be one element wide as was done, for example in ref. 1. However, the finite model was made up to eight elements wide for the following reasons: (1) the model simulates the more complex model except for the fact that the inclusions are removed by filling their space are with uniform elements having the same material property as the surrounding rubber material; (2) the problem is artificially made mathematically larger so that more meaningful comparisons of CPU run times could be made; (3) larger problem sizes tend to draw out any potential problems with equation solvers. It is not our intent to discuss or explain the setup of wave propagation problems of the type represented by the figures 1 - 2 example model; the reader is referred to refs. 1 and 2 for supplementary details. In fact, the demonstration problem used here is very similar to the one used in the ref. 1 sample problem except that the plate and viscoelastic thicknesses are different, the damping coefficient in the viscoelastic layer is different and that the steel plate is represented here approximately with CBAR elements rather than with solid elements as in ref. 1. Specifically, the material constants employed are listed below

DEMONSTRATION PROBLEM PHYSICAL CONSTANTS

MATERIAL	λ^r psi	μ^r psi	λ^i psi	μ^i psi	ρ lb-sec^2/in^4
Water	345,600.	0.0	0.0	0.0	.000096
Steel	17,307,000.	11,538,000.	0.0	0.0	.000735
Viscoelastic Material	86,703.	115.9	8670.3	11.59	.0003599

where the meaning of the elastic and viscoelastic constants are defined in detail in ref. 1.

Since the topic of interest here is related to the class of problem treated by ref. 1, it appears appropriate to print an errata to the ref. 1 paper

- in equation (15) of ref. 1, replace G22 = λ^r with G22 = $\lambda^r + 2\mu^r$
- in equation (16) of ref. 1, replace G22 = ε with G22 = $\varepsilon(1 + 2\mu^i/\lambda^i)$
- in equation (2) of ref. 1, replace ω^2 with $-\omega^2$
- in equation (17) of ref. 1, replace $+\eta_{d_2}$ with $+i\eta_{d_2}$ in the k_2 definition

PARAMETER VARIATIONS

The basic finite element model, figure 2, was exercised for a frequency sweep of 7 different incident frequencies (3.0 kHz, 4.0 kHz, 6.0 kHz, 8.0 kHz, 17.5 kHz, 22.5 kHz, 35.0 kHz). Running the figure 1 model on NASTRAN for the above frequency sweep is designated as a typical run and correspondingly assign it a "run number", which runs from the number 2 through 9. Run number 1 is the exact solution and therefore is the only non NASTRAN designation (it is called a run since even the analytical solution involves a computer evaluation).

Next, the same frequency sweep input data was rerun while varying the following parameters:

- solution precision (S.P. or D.P. on the same computer)
- type of computer (UNIVAC 1108; DEC-VAX; CDC Cyber 175)
- level of NASTRAN (both NASA and MSC versions are considered)
- date (i.e., the same input is resubmitted on the same computer, using the same version of NASTRAN but on different days)

The last parameter (i.e., the date) seems a waste of computer time, however as is shown later, some unexpected results are encountered.

DMAP INSTRUCTIONS FOR PRINTING SOLUTION ERROR RESIDUAL

It is of interest to know the accuracy of the solution solving capability of the equation solver used by the particular version of NASTRAN employed by the user. Specifically, if the solution {U} is found by NASTRAN, how well does it satisfy the linear simultaneous equations (1)? Consider

substituting the solution {U} into equation (1) and then transposing the applied force vector to the left hand side of equation (1) to obtain

$$[\overline{K}]\{U\} - \{F\} = \{\Delta\} \qquad (3)$$

If the equations have been solved exactly, then the residual vector {Δ} will be identically zero. The appearance of large nonzero entries in {Δ} would imply potential inaccuracies in the solution vector {U}. The question of "how large is large?" should be viewed by comparing the size of a particular entry in the {Δ} vector to the size of the applied loads (for this reason, the load vector {F} is also printed). For example, a residual of .2 would be a big residual if the applied forces are on the order of 1.0 lbs.; however, if applied forces are say 100,000 lbs., the .2 residual is acceptable.

In order to print out the residual vector {Δ} for the 3,000 Hz driving frequency case, the following DMAP instructions were used (note, the {Δ} vector is printed with the heading DELSØL).

- For UNIVAC 1108, SOL 8, LEVEL 17.0 (Runs 3a, 4a)

```
ALTER 159
ADD5 KDD,BDD,MDD,,/KBARX/C,Y,ALPHA=(1.0,0.0)/C,Y,BETA=(0.0,18849.5592)/
                        C,Y,GAMA=(-355305758.44,0.0) $
MPYAD KBARX,UDVF,PDF/DELSØL/C,N,0/C,N,1/C,N,-1/ $
MATPRN DELSØL,PDF,,,// $
ENDALTER
CEND
```

where frequency dependent input constants BETA and GAMA are to be input by the user and are simply defined as:

$$\text{BETA} = 0.0 + i\omega$$

$$\text{GAMA} = -\omega^2 + i\ 0.0$$

where ω = the driving frequency in radians/sec

- For UNIVAC 1108, SOL 8, NASA LEVEL 15.5 (Run 2) replace ALTER 159 with ALTER 139
- For VAX, SOL 8, NASA LEVEL 17.5 (Run 5b) same as 1108, NASA LEVEL 17.0
- For CDC CYBER, SOL 8, MSC LEVEL 48B (Run 6) replace ALTER 159 with ALTER 139
- For VAX, SOL 8, MSC LEVEL 52 (Run 7) replace ALTER 159 with ALTER 139
- For VAX, SOL 8, MSC LEVEL 60 (Run 8) replace ALTER 159 with ALTER 139
- For VAX, SOL 26, MSC LEVEL 60 (Runs 9b, 9c, 9d)
  replace ALTER 159 with ALTER 409
  replace UDVF with UHV (3rd line)
  replace PDF with PD (3rd and 4th line)

When a large residual is encountered, it is desirable to know the node and component number where a large residual appears (i.e., knowing the run number of the questionable residual, what node-component number does this correspond to?). By inserting a DIAG 22 card, the desired correspondence between run number of $\{\Delta\}$ and the node-component number can be made.

It is important to note that the simple DMAP sequence as presented will apply to only a single frequency; thus, if a frequency sweep is employed, only the n^{th} column of the DELSOL vector (i.e., $\{\Delta\}$) will be correct; the remaining columns of DELSOL should be ignored, where n = the n^{th} value in the frequency sweep appearing on the NASTRAN FREQ card. ard.

DISCUSSION OF RESULTS

The primary variables of interest to us in this study are: (1) the transmitted pressure in the fluid on the back side of the steel plate, P_B, (e.g., in element number 100352 as shown in figure 2) and (2) the scattered pressure in the fluid on the front side of the plate, P_S (e.g., in element number 100378 as shown in figure 2). The transmitted pressure is read directly from the NASTRAN printout, whereas the scattered pressure is obtained indirectly from the NASTRAN printout by simply subtracting the incident pressure (equation (2)) from the total pressure printed by NASTRAN. The scattered pressure is of prime importance with regard to establishing the energy absorbing properties of the viscoelastic configuration. As discussed in ref. 2, it has been our experience that for steady state wave propagation problems of the type considered here, at least 10 elements per wave length are needed to adequately compute the pressure response for elements of the type employed in this study. In order to demonstrate this accuracy limitation, the model has been purposely exercised in a driving frequency range that is too high for the mesh to properly produce sufficiently accurate results (i.e., the mesh is too coarse for some of the higher frequencies). For a rubber wave speed of C_2 = 15540. in/sec, and the coarsest element in the rubber mesh (.1" x .1" elements), the following frequency vs. element/(wave length) chart is constructed:

freq. (kHz)	elements/wave length (rubber in figure 2)
3.0	51.8
4.0	38.8
6.0	25.9
8.0	19.4
17.5	8.8
22.5	6.9
35.0	4.4

Based on the above chart, it is expected that the accuracy of the finite element solution in relation to the exact solution should start to drift at frequencies of 17.5 kHz and higher.

A total of 14 computer runs were made (designated as runs number 2, 3a, 3b, ... 9d) and are tabulated in Table 1 (for the transmitted back side pressure P_B) and in Table 2 (for the scattered front side pressure, P_S). The pressure results are normalized by the magnitude of the incident pressure wave, P_0. In addition to the pressure result, Table 1 has two additional pieces of information, namely the magnitude of the largest complex residual, $|\Delta|$, appearing in the residual vector $\{\Delta\}$ as computed by equation (3), (the $|\Delta|$ value is obtained by scanning the DMAP printed DELSØL printout and seeking out the largest absolute value of all the rows of the $\{\Delta\}$ vector corresponding to the 3,000 Hz frequency case. Note that the residual for only the 3,000 Hz case is reported. Also listed is the total CPU time required to execute the full frequency sweep solution for the run in question.

Word Length (Precision) Sensitivity

The earlier Level 15.5 version of NASA NASTRAN for a UNIVAC 1108 computer, does not efficiently solve complex systems of equations of the type given by equation (1), (i.e., steady state time harmonic rigid format 8) when the double precision option is used. Experience on large problems (e.g., the size of the one in ref. 2, pg 435) has demonstrated that in the solving of the equation (1), NASTRAN has spent literally hours in the decomposition operation. In order to obtain reasonable run times, a PARAM DECOMOPT4 card is added to the bulk data in order to force the decomposition to work in single precision. In comparing the NASTRAN Run 2 (Table 1) (1108-S.P.) results to the exact solution over the frequency range

(3. - 8. kHz) where the mesh is sufficiently fine,* it is noted that at 6.0 kHz, a 154.0% error in the transmitted pressure is experienced. The corresponding error on the scattered pressure (Run 2, Table 2) is not as severe, namely 16.5%. It is noted that the 154% error is the situation that motivated this entire comparative study. The percent errors at 3. kHz, 4.0 kHz and 8 kHz are also much larger than should be expected for the mesh size employed. The largest residual, $|\Delta|$, in the residual vector $\{\Delta\}$, is .272 at 3.0 kHz; this is in comparison to a load vector component of the size (.20) (the largest residual was not at a loaded node however). This is a further indication that the solution resulting from decomposition in single precision is not accurate enough. Upon running the same problem on a Level 17.0 version of NASA-NASTRAN on an 1108 computer (single precision must be invoked with DMAP) in single precision, Run 3a still results in a similar bad solution with a similar worse residual $|\Delta|$. The fact that Level 17.0 uses a different decomposition algorithm did not improve the bad results. However, again running the same problem on Level 17.0 on the UNIVAC 1108 in double precision (the default situation) resulted in excellent results in comparison to the exact solution. For example, at the 6.0 kHz, the percent error reduced from 154.0% down to 0.4% error. Similarly good results were obtained in the entire (3.0 - 8.0 kHz range) for both the transmitted and scattered pressure. The double precision gave a very small worse residual at 3,000 Hz, $|\Delta| = 2.099 \times 10^{-8}$, which suggests that equation (1) has been solved accurately. In comparing CPU times between Run 3a and Run 4a in Table 1, it appears there is little penalty in CPU time between single and double precision runs for Level 17.0. Consequently, there is really no incentive (from a time saving point of view) to make Sol. 8 type runs in single precision, as there was for Level 15.5 NASTRAN.

Again running the same problem on Level 48B, MSC version of NASTRAN on a CDC CYBER 175 computer (Run 6), very good results were obtained in relation to the exact solution over the (3.0 - 8.0 kHz range); further, excellent consistency with the UNIVAC 1108 double precision runs is demonstrated at all frequencies by comparing Run 4a with Run 6. The worst residual, $|\Delta|$, on the CDC computer is not as good as the 1108 double precision run, but this is expected since the CDC single precision word length is slightly smaller than the 1108 double precision word length; however, it should be noted that differences in the decomposition algorithms could also account for differences in the worse residual, even if the word lengths were the same.

The Level 52, MSC version of NASTRAN on a DEC-VAX computer (Run 7) gave comparible results to the Level 17.0 NASA NASTRAN double precision 1108, and to Level 48B, MSC NASTRAN on a CDC-CYBER 175. The word length in double precision on the VAX is slightly more than the single precision CDC and slightly less than the double precision 1108. It is noted the Run 7 gave the smallest worse residual.

---

* It should be emphasized that in comparing any NASTRAN result to the exact solution, for the purpose of measuring the formulation quality, it should be done only in the frequency range of 3.0 - 8.0 kHz where the mesh satisfies the 10 element/wave length criterion.

Computer Type/Level of NASTRAN Sensitivity

It is inconvenient to run the same exact version of NASTRAN on different computers due to leasing restrictions, consequently this combination was not done. Thus running NASTRAN on different computers always involved running a different NASTRAN version as well. In scanning the results of Table 1 and Table 2, all the runs performed with decomposition precision using word lengths between 60 - 72 bits (i.e., Runs 3a, 4a, 6, 7) gave both accurate results in comparison to the exact solution (3.0 - 8.0 kHz range) and consistent results from machine-to-machine and version-to-version. We have purposely not commented on the accuracy of Runs 5, 8 and 9 involving the DEC-VAX computer due to some reservations we have regarding the operating conditions of the particular VAX on which these runs were made, and will be discussed next.

Solution Repeatability

During the process of preparing this collection of comparative runs, an unexplained phenomenom (which is still unexplained as of this writing) occurred, namely the fact that Level 60 MSC-NASTRAN run on a DEC-VAX computer gave different results to the same problem upon rerunning the same data. The Run 9 series of runs were made on rigid format 26 which is comparable (there are differences in the decomposition routine) to NASA-NASTRAN rigid format 8. For example the input producing Run 9a was resubmitted over again (producing Run 9b) so that the residual vector $\{\Delta\}$ could be printed (employing the DMAP instructions given earlier). In comparing the solutions, the results were slightly different (e.g., 5.2% at 3.0 kHz). The same input data was again rerun on successive days producing Runs 9c and 9d. Run 9c is the closest to the more stable results made on the 1108 and CDC computers.

Using the same DEC-VAX computer facility, the base case input was resubmitted again employing the Level 17.5 NASA/GODDARD NASTRAN and thus producing Runs 5a and 5b. Again the nonrepeatability of the solution on the VAX was experienced, this time with an entirely different version of NASTRAN.

An MSC Level 60 DEC-VAX computer run was made similar to Run 9a through Run 9d, except that the older rigid format 8 instead of the newer MSC rigid format 26 was employed. The results were poor in comparison to the exact solution, further, the worse residual of $|\Delta| = .2127$ was unacceptably high. No reruns of the same input were made on this version and level of NASTRAN.

In order to demonstrate that repeatable results are possible (a notion we usually assume is true on most modern computers), the base case data was rerun on the UNIVAC 1108 computer; single precision Runs 3a and 3b were totally repeatable as well as double precision Runs 4a and 4b.

The facts that (1) the VAX computer resulted in nonrepeatable results employing two separate versions of NASTRAN and (2) the Level 60 MSC NASTRAN for

the VAX computer, Sol. 8 (Run 8) gave poor results, strongly suggests a problem with the particular VAX computer on which the runs were made. The following list provides some possible reasons for nonrepeatability:

- computer central processing drops a bit in the main memory or operating register
- main memory itself drops a bit between storing and retrieving data
- floating point accelerator drops a bit during calculations
- disk subsystem (drive or interface drops a bit)
- computer temperature rises due to air conditioning not keeping up with thermal load during the summer months when the runs were made; this could result in dropping a bit by one of the four above mentioned possibilities.

Finally it is noted that as of this writing, non NASTRAN users of the same VAX computer that made the runs reported here, did not report any repeatability problems with computer program results totally unrelated to NASTRAN.

CONCLUSIONS

The paper is concerned with both the accuracy of equation solvers and with the accuracy of the problem formulation for large dynamic steady state problems (e.g., rigid format 8). Based on a series of computer runs on different versions of NASTRAN on different computers, the following set of conclusions are drawn:

- NASTRAN solution decomposition algorithms employing less than a 60 bit word could lead to serious errors in the results (e.g., employing 36 bit single precision words on an 1108 computer gave up to 154% error in the solution with both Level 15.5 and Level 17.0 versions of NASA NASTRAN.
- Correlation between results run on the 1108 double precision Level 17.0 NASA NASTRAN; CDC-CYBER 175 single precision Level 48B MSC-NASTRAN; and DEC-VAX double precision Level 52 MSC-NASTRAN were excellent.
- Unexplained unrepeatability of results were experienced on the DEC-VAX computer for both MSC-Level 60 NASTRAN and NASA-GODDARD Level 17.5 NASTRAN; Level 17.0 of NASA NASTRAN had no repeatability problems on an 1108 computer.
- A minimum of 10 elements per wave length should be used to model traveling wave propagation problems of the type treated in this paper; coarser meshes lead to increasingly bad results when comparing NASTRAN results to the exact continuum solution to the same problem.

As a final comment regarding repeatability, it is noted that it is not being suggested that this problem is one to be found in all VAX computers employing NASTRAN. The spirit of the NASTRAN Colloquium is to share USER's experiences, thus it was felt that our problem should be brought to the attention of the NASTRAN USER's community in the event that similar problems are encountered by others.

REFERENCES

1. Kalinowski, A. J., "Modeling Structural Damping for Solids Having Distinct Shear and Dilatational Loss Factors", Seventh NASTRAN USER's COLLOQUIUM, Oct. 1978 (NASA CP-2062).

2. Kalinowski, A. J., "Fluid/Structure Interaction", Shock and Vibration Computer Programs - Review and Summaries, edited by W. Pilkey and B. Pilkey, The Shock and Vibration Information Center, 1975, U.S. Dept. of Defense.

FIGURE 1 - TEST PROBLEM

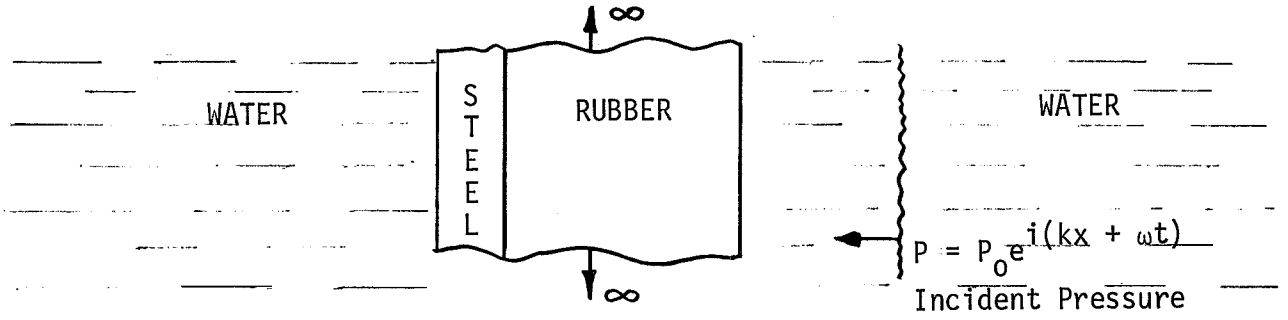
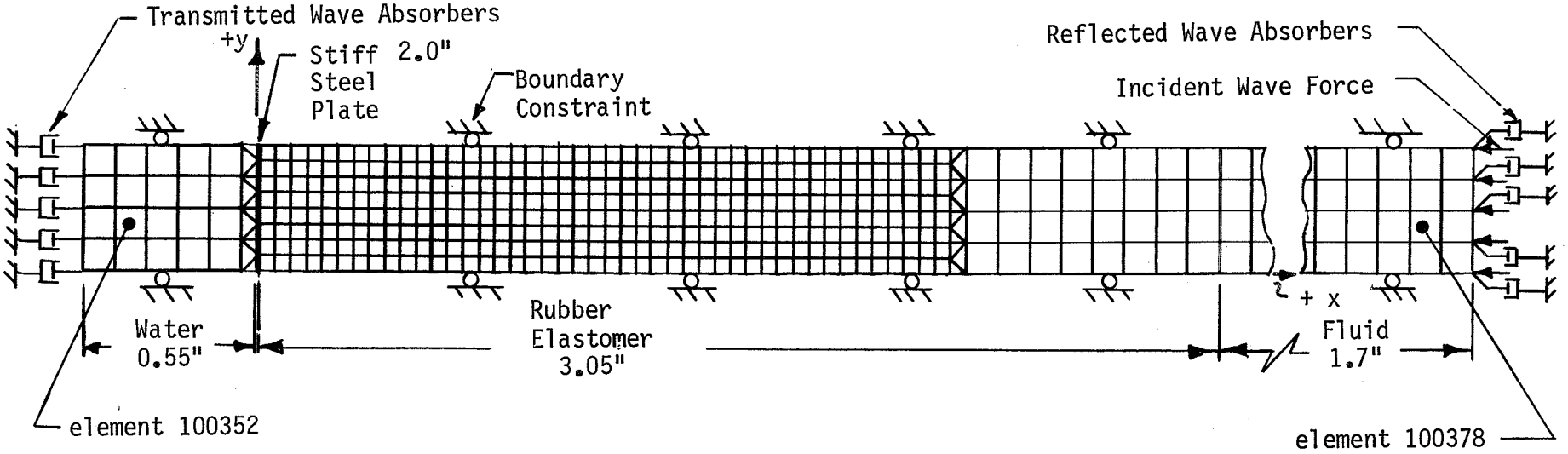


FIGURE 2 - FINITE ELEMENT MODEL



- 856 Elements (CQDMEM, (TRMEM, CBAR)
- 588 Grid Points

S.P. (Single Precision)
D.P. (Double Precision)

TABLE 1 - NASTRAN COMPARATIVE SOLUTIONS
(BACK SIDE PRESSURE, element 100352)

FREQUENCY / SOLUTION SOURCE	BACK SIDE PRESSURE, P_B/P_0 (amplitude) 35. kHz	22.5 kHz	17.5 kHz	8. kHz	6. kHz	4. kHz	3. kHz	Largest Residual $\|\Delta\|$ at 3. kHz	TOTAL CPU (min)	RUN NUMBER
EXACT ANALYTICAL (Ref. 1)	.00809	.01776	.02796	.09550	.13893	.22863	.31324	not apply	not apply	1
Level 15.5 NASA-NASTRAN, UNIVAC 1108 Computer, 36 Bit S.P. Decomposition (Sol.8)	.00284	.01343	.02573	.16978	.35292	.34048	.27799	.272	21.12	2
Level 17.0 NASA-NASTRAN, UNIVAC 1108 Computer, 36 Bit S.P. Decomposition (Sol.8)	.00282 .00282	.01318 .01318	.02454 .02454	.13781 .13781	.30044 .30044	.38303 .38303	.33906 .33906	.247 not avail.	19.15 19.07	3a 3b
Level 17.0 NASA-NASTRAN, UNIVAC 1108 Computer, 72 Bit D.P. Decomposition (Sol.8)	.00277 .00277	.01268 .01268	.02307 .02307	.09247 .09247	.13638 .13638	.22677 .22677	.31191 .31191	2.099×10^{-8} not avail.	20.01 19.29	4a 4b
Level 17.5 NASA/ GODDARD NASTRAN, DEC-VAX Computer, 64 Bit D.P. Decomposition (Sol.8)	.00285 .00279	.01289 .01139	.02278 .02248	.08913 .08145	.13305 .12512	.21937 .22887	.30407 .27257	not avail. .07426	52.18 59.45	5a 5b
Level 48B MSC-NASTRAN CDC CYBER 175 Computer, 60 Bit S.P. Decomposition (Sol.8)	.00277	.01268	.02307	.09247	.13638	.22678	.31192	1.24×10^{-5}	3.03	6
Level 52 MSC-NASTRAN, DEC-VAX Computer, 64 Bit D.P. Decomposition (Sol.8)	.00277	.01268	.02307	.09247	.13638	.22677	.31191	7.422×10^{-9}	22.33	7
Level 60 MSC-NASTRAN, DEC-VAX Computer, 64 Bit D.P. Decomposition (Sol.8)	.00259	.01268	.02304	.09266	.13659	.22602	.45052	.2127	54.28	8
Level 60 MSC-NASTRAN, DEC-VAX Computer 64 Bit D.P. Decomposition (Sol.26) (4 runs on same data)	.00272 .00277 .00276 .00277	.01263 .01265 .01268 .01268	.02261 .02314 .02325 .02159	.08125 .09278 .09360 .09246	.13343 .13627 .13633 .13912	.22745 .21877 .22709 .22467	.27247 .28690 .31192 .31191	not avail. 5.65×10^{-5} 20.6×10^{-5} 9.82×10^{-5}	not avail. 51.56 51.81 23.44	9a 9b 9c 9d

S.P. (Single Precision)
D.P. (Double Precision)

TABLE 2 - NASTRAN COMPARATIVE SOLUTIONS
(FRONT SIDE PRESSURE, element 100378)

SOLUTION SOURCE \ FREQUENCY	FRONT SIDE SCATTERED PRESSURE, P_S/P_0 (amplitude)							RUN NUMBER
	35.kHz	22.5 kHz	17.5 kHz	8. kHz	6. kHz	4. kHz	3. kHz	
EXACT ANALYTICAL (Ref. 1)	.04253	.08580	.13310	.34059	.44266	.59075	.59907	1
Level 15.5 NASA-NASTRAN, UNIVAC 1008 Computer, 36 Bit S.P. Decomposition (Sol.8)	.15093	.05609	.11772	.32372	.36949	.59695	.62678	2
Level 17.0 NASA-NASTRAN, UNIVAC 1108 Computer, 36 Bit S.P. Decomposition (Sol.8)	.15093	.05603	.11759	.32875	.39233	.58218	.60932	3a
	.15093	.05603	.11759	.32875	.39233	.58218	.60932	3b
Level 17.0 NASA-NASTRAN, UNIVAC 1108 Computer, 72 Bit D.P. Decomposition (Sol.8)	.15093	.05592	.11744	.33419	.44176	.59134	.59943	4a
	.15093	.05592	.11744	.33419	.44176	.59134	.59943	4b
Level 17.5 NASA/GODDARD NASTRAN, DEC-VAX Computer, 64 Bit D.P. Decomposition (Sol.8)	.14389	.06016	.12378	.29474	.40715	.58416	.59813	5a
	.14385	.06717	.11805	.24716	.32194	.50502	.56985	5b
Level 48B MSC-NASTRAN CDC CYBER 175 Computer, 60 Bit S.P. Decomposition (Sol.8)	.15093	.05591	.11744	.33419	.44176	.59134	.59942	6
Level 52 MSC-NASTRAN, DEC-VAX Computer, 64 Bit D.P. Decomposition (Sol.8)	.15093	.05592	.11744	.33419	.44176	.59134	.59942	7
Level 60 MSC-NASTRAN, DEC-VAX Computer, 64 Bit D.P. Decomposition (Sol.8)	.16173	.05591	.11745	.33203	.43973	.59726	.59972	8
Level 60 MSC-NASTRAN, DEC-VAX Computer, 64 Bit D.P. Decomposition (Sol.26)	.15304	.05422	.11353	.28611	.44209	.55800	.50161	9a
(4 runs on same data)	.15100	.05838	.11587	.33329	.44178	.59291	.60261	9b
	.15095	.05592	.11822	.33563	.44131	.59211	.59944	9c
	.15093	.05592	.12031	.33385	.45400	.59269	.59942	9d

RING ELEMENT DYNAMIC STRESSES

NANCY LAMBERT
A. O. SMITH ENGINEERING SYSTEMS

MICHAEL TUCCHIO
NAVAL UNDERWATER SYSTEMS CENTER

ABSTRACT

The stresses in the CTRAPRG and CTRIARG ring elements are not calculated for any of the dynamic solutions in the current COSMIC version of NASTRAN. This paper presents a DMAP alter sequence for Solution 8 and post-processing program, NASTPOST, to calculate these stresses. Test cases are presented which describe the method. The stiffness and the consistent versus concentrated mass problems which have been ascribed to this element are reviewed.

The DMAP alter sequence introduces Solution 8 displacements to a Solution 1 module to calculate Real and Imaginary stress components during the execution of Solution 8. The post-processor, NASTPOST, calculates the magnitude/phase stress results.

The DMAP sequence has been written specifically for Level 52 MSC/NASTRAN, but can certainly be used for any COSMIC version with slight modification.

INTRODUCTION

None of the currently documented versions of NASTRAN calculate the dynamic stresses in the CTRAPRG and CTRIARG solid of revolution elements. The stresses for these elements are calculated in NASTRAN for static solutions (e.g., Solution 1) but not in the dynamic solutions (e.g., Solution 8). Comments have been made by others which express the reasons for not including the stress calculations are related to the formulation of the mass matrix for the element.

Sample problems are given to show that the difference between the consistent and concentrated mass approach is greater than one might expect from arguments solely between the merits of consistent or concentrated mass.

This paper describes a DMAP alter sequence for Solution 8 and a post-processing program, NASTPOST, to calculate these dynamic stresses. The DMAP alter sequence introduces the displacements computed in Solution 8 to a Solution 1 module to calculate the complex stresses in the form of real and imaginary components. The post-processor, NASTPOST, calculates the stresses in the form of magnitude/phase.

DISCUSSION

It is not spelled out in the NASTRAN Users Manual that stresses for the solid of revolution elements are not calculated for dynamic solutions. Therefore, if one asks for stresses in a Solution 8 case control, the run is not aborted, but no stresses are obtained.

In order to perform noise path studies of an axisymmetric structure it became necessary to obtain these stresses. At first, the displacements for the entire structure, obtained from a Solution 8 forced vibration analysis were written into an output file; then these displacements, less one, were written into SPC format as enforced displacements for a Solution static analysis (this was done for the real and imaginary components separately). This technique was later modified, utilizing the DMAP alter sequence AOS8$CS and a post-processor, NASTPOST.

The DMAP alter sequence is given in Figure 1. The major points are:

- The user can specify output requests as usual for SPCFORCES and DISPLACEMENTS.
- The user should specify STRESS (PUNCH) = ALL or a particular set ID if he wishes to subsequently use NASTPOST to calculate the magnitude/phase. This punched file will be sent to the users system space. (FOR 013.DAT for the MSC/NASTRAN VAX 11/780 VERSION).
- AOS8$CS should be placed on the user's RFALTER library and executed then by calling RFAI = AOS8$CS.

The program NASTPOST is given in the appendix and is used to calculate magnitude/phase stress components from real/imaginary stress components. The major points are:

- The components from FOR013.DAT above, are used as input to calculate the magnitude/phase stress components.
- This program can be run immediately after the execution of MSC/NASTRAN or at some later time.

The test problem for AOS8$CS and NASTPOST is a circular plate fixed at the edges and driven by a single force, 100 dynes, at the center, normal to the plane of the plate. The finite element control model is the CQUAD2 and CTRIAG2 bending element model shown in Figure 2. The CTRAPRG model, shown in Figure 3, is formulated as a concentrated or consistent mass for each of the runs. The NASTRAN default value is the consistent mass matrix. The concentrated mass matrix is entered as CONM2 data. The three cases are compared in Table 1 for static, 2000 Hz and 8000 Hz at a position near the concentrated load and at the fixed edge.

The concentrated mass formulation gives good results, as compared to the control model. The consistent mass, or default formulation, gives results which do not agree with the control model at either the low, 2 kHz, or high, 8 kHz, forcing frequencies.

The static solution agrees very well with the control model which indicates that the stiffness of the model is represented correctly by solid of revolution elements. The error therefore is associated with the mass matrix formulation. The degree of error is obviously greater than one would expect from the normal arguments of consistent versus concentrated mass differences.[1]

It can be argued that the use of cyclic symmetry with 3D elements rather than solid of revolution elements would have been a possible solution. This is certainly an avenue that deserves added investigation for comparison of cost and accuracy of solution compared to the solid of revolution elements with concentrated mass matrix.

CONCLUDING REMARKS

A DMAP alter sequence for Solution 8 and a post-processing program NASTPOST has been presented to calculate the dynamic stresses in CTRAPRG and CTRIARG solid of revolution ring finite elements. Users of this technique are cautioned to use the concentrated or lumped mass matrix rather than the consistent mass (default value) matrix.

The DMAP sequence has been written specifically for Level 52 MSC/NASTRAN, but can certainly be used for any COSMIC version with slight modification.

REFERENCES

1. Cook, R. D., "Concepts and Applications of Finite Element Analysis", John Wiley & Sons, Inc.

TABLE 1

COMPARISON OF STRESSES, 3/8 cm from CONCENTRATED LOAD

FREQUENCY	0[1]	2 kHz	8 kHz
QUAD2	134.4	75.5	66.4
TRAPRG (CONS.)	132.3	17.2	63.1
TRARG (CONC.)	132.3	96.	60.5

TABLE 2

COMPARISON OF STRESSES, 3/8 cm from FIXED EDGE

FREQUENCY	0[1]	2 kHz	8 kHz
QUAD2	44.4	34.2	38.2
TRAPRG (CONS.)	45.6	27.0	10.0
TRAPRG (CONC.)	45.6	33.0	36.0

[1] OBTAINED FROM SOLUTION 1

FIGURE 1 - ALTER AOS8$CS

```
$ BEGINNING OF ALTER AOS8$CS
$
$ THIS ALTER PACKAGE IS USED TO CALCULATE
$
$       *DISPLACEMENTS   (REAL/IMAGINARY) OR
$                        (MAGNITUDE/PHASE)
$
$       *SPCFORCES       (REAL/IMAGINARY) OR
$                        (MAGNITUDE/PHASE)
$
$       *STRESSES        (REAL/IMAGINARY)
$
$   FOR THE CTRAPRG AND CTRIARG RING ELEMENTS
$
$
$ CASE CONTROL INPUT
```

FIGURE 1 - (Cont'd)

```
$
$       THE USER SHOULD SELECT THE DESIRED
$       OUTPUT AS USUAL FOR DISPLACEMENTS
$       AND SPCFORCES.
$
$       THE USER SHOULD SELECT THE PUNCH
$       OPTION FOR STRESS IF IT IS DESIRED TO
$       SUBSEQUENTLY CALCULATE (MAGNITUDE/
$       PHASE) USING A POST-PROCESSING PROGRAM
$
$
ALTER 166
OFP OPPC1,OQPC1,OUPVC1,,,//V,N,CARDNO $
ALTER 185,186
PARAM //STSR/13/-64 $
GP3 GEOM3,EQEXIN,GEOM2/,ETT/0/V,N,NOGRAV/0 $
```

FIGURE 1 - (Cont'd)

```
PARAML UPVC//C,N,TRAILER/2/V,N,ROWS $
MATGEN ,/UNIT/1/ROWS $
MODTRL UPVC////3 $
MPYAD UNIT,UPVC,/ASQR/ $
DIAGONAL ASQR/ATRM// $
ADD UPVC,/BSQR/(0.0,-1.0) $
DIAGONAL BSQR/BTRM// $
SDR2 CASECC,CSTM,MPT,DIT,EQEXIN,SIL,ETT,EDT,BGPDT,,,ATRM,EST,
  XYCDB/,,,OESCR,,/STATICS/S,N,NOSORT2 $
SDR2 CASECC,CSTM,MPT,DIT,EQEXIN,SIL,ETT,EDT,BGPDT,,,BTRM,EST,
  XYCDB/,,,OESCI,,/STATICS/S,N,NOSORT2 $
OFP  ,,,OESCR,,//S,N,CARDNO $
OFP  ,,,OESCI,,//S,N,CARDNO $
PARAM //STSR/7/-64 $
ENDALTER $
$
```

FIGURE 2 - CQUAD2, CTRIAG FINITE ELEMENT MODEL OF 10.00 CM DIA., 1 CM THK PLATE

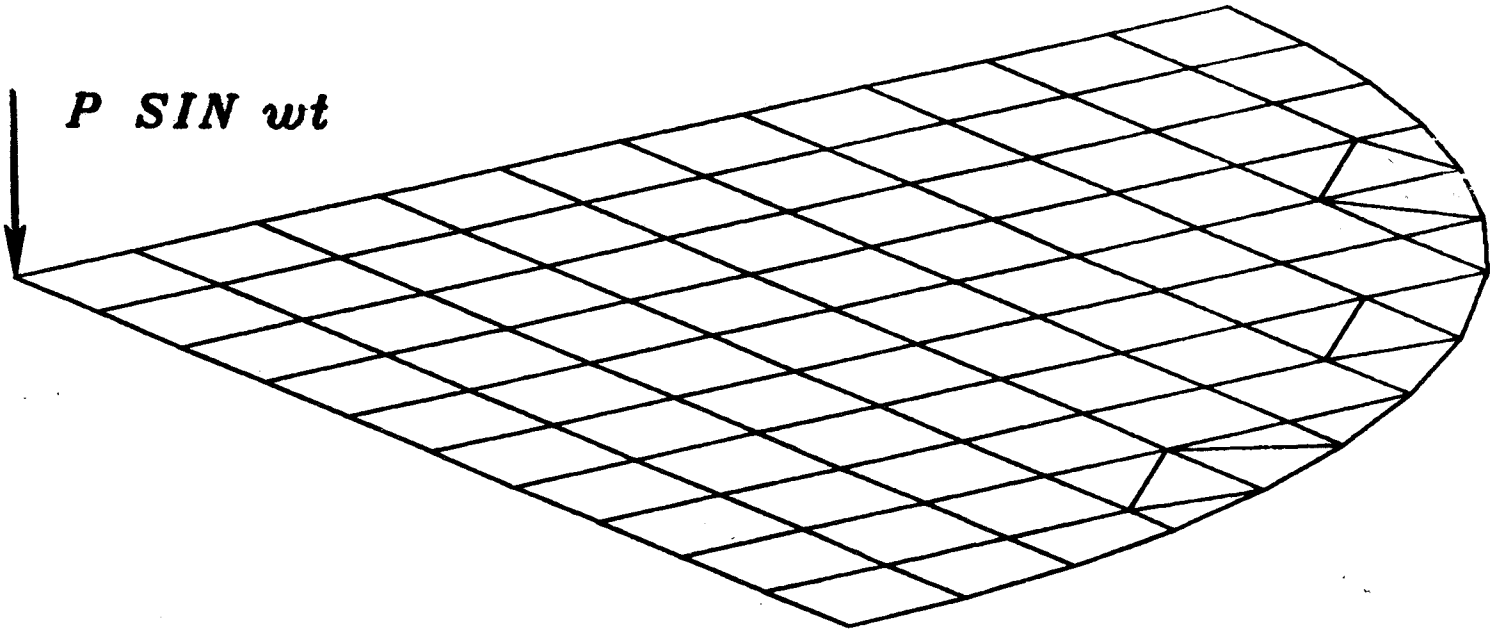
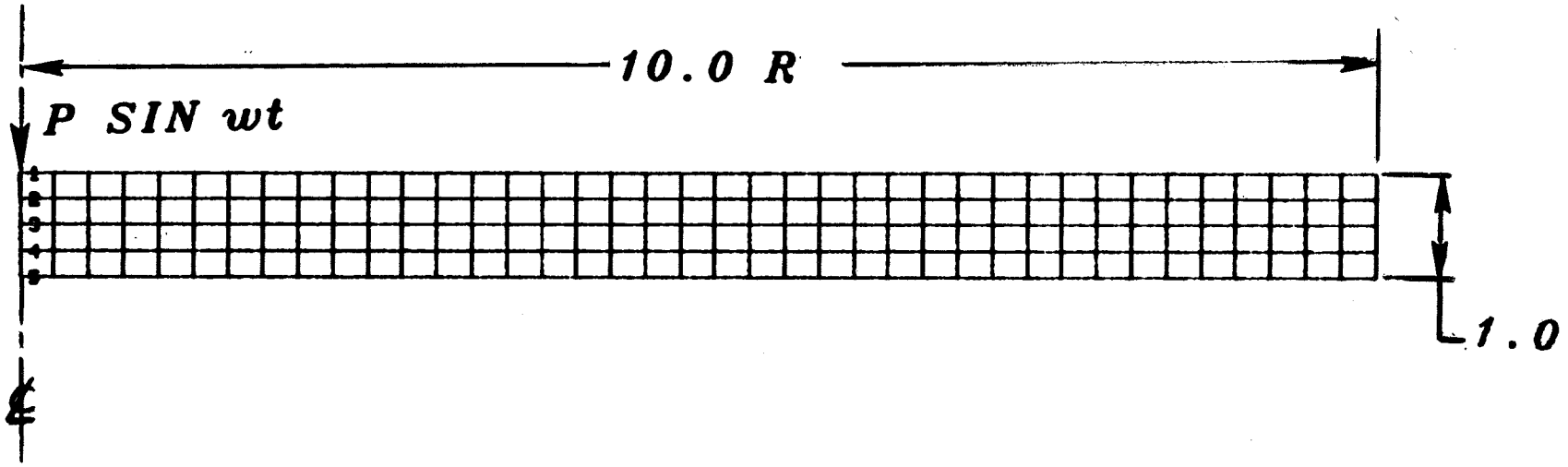


FIGURE 3 - CTRAPRG SOLID OF REVOLUTION FINITE ELEMENT MODEL



APPENDIX A

THE NASTPOST PROGRAM

```
C          DATA SET NASTPOST   AT LEVEL 017 AS OF 11/05/79                  00001
      COMMON /HDRCOM/TITLE(16),SUBT(16),LABEL(16)                           00002
      DATA DTIT/'$TIT'/,CASE/'CASE'/,DSUB/'$SUB'/,                          00003
     *     DELE/'$ELE'/,BSTR/' STR'/,DLAB/'$LAB'/                           00004
      DATA I036,I037/2*0/                                                   00005
    1 CONTINUE                                                              00006
      REWIND 7                                                              00007
C - GET TITLE CARD                                                          00008
    5 CONTINUE                                                              00009
      READ(7,900,END=999) TEMP,TITLE                                        00010
      IF(TEMP.EQ.DTIT) GO TO 6                                              00011
      GO TO 5                                                               00012
C - GET SUBTITLE CARD                                                       00013
    6 CONTINUE                                                              00014
      READ(7,900,END=999) TEMP,SUBT                                         00015
      IF(TEMP.EQ.DSUB) GO TO 7                                              00016
      GO TO 6                                                               00017
C - GET LABEL CARD                                                          00018
    7 CONTINUE                                                              00019
      READ(7,900,END=999) TEMP,LABEL                                        00020
      IF(TEMP.EQ.DLAB) GO TO 10                                             00021
      GO TO 7                                                               00022
C - GET STRESS CARD                                                         00023
   10 CONTINUE                                                              00024
      READ(7,910,END=999) TEMP                                              00025
      IF(TEMP.EQ.BSTR) GO TO 20                                             00026
      GO TO 10                                                              00027
C - GET SUBCASE IDENTIFICATION                                              00028
   20 CONTINUE                                                              00029
      READ(7,920,END=999) TEMP,ISID                                         00030
      IF(TEMP.EQ.CASE) GO TO 30                                             00031
      GO TO 20                                                              00032
C - GET ELEMENT TYPE                                                        00033
   30 CONTINUE                                                              00034
      READ(7,930,END=999) TEMP,IELTYP                                       00035
      IF(TEMP.NE.DELE) GO TO 5                                              00036
C - CHECK ELEMENT TYPES                                                     00037
      IF(IELTYP.EQ.36) GO TO 360                                            00038
      IF(IELTYP.EQ.37) GO TO 370                                            00039
      GO TO 5                                                               00040
C - ELEMENT TYPE = 36                                                       00041
  360 CONTINUE                                                              00042
      IF(I036 .EQ. 0) CALL RW36(ISID,IELTYP,IEOF)                           00043
      IF(I036 .EQ. 1) CALL RC36(ISID,IELTYP,IEOF)                           00044
      IF(I036 .EQ. 1 .AND. IEOF .EQ. 1) GO TO 999                           00045
      I036 = MOD(I036+1,2)                                                  00046
      GO TO 6                                                               00047
C - ELEMENT TYPE = 37                                                       00048
  370 CONTINUE                                                              00049
      IF(I037 .EQ. 0) CALL RW37(ISID,IELTYP,IEOF)                           00050
      IF(I037 .EQ. 1) CALL RC37(ISID,IELTYP,IEOF)                           00051
      IF(I037 .EQ. 1 .AND. IEOF .EQ. 1) GO TO 999                           00052
      I037 = MOD(I037+1,2)                                                  00053
      GO TO 6                                                               00054
  999 STOP                                                                  00055
  900 FORMAT(A4,6X,15A4,A2)                                                 00056
  910 FORMAT(8X,A4)                                                         00057
  920 FORMAT(4X,A4,8X,I9)
  930
```

```
FORMAT(A4,12X,I11)                                                      58
      END                                                               00059
C           DATA SET NASTRU36   AT LEVEL 004 AS OF 11/02/79
      SUBROUTINE RU36(ISID,IELTYP,IEOF)                                 00001
      DIMENSION TEMP(2),DATA(4)                                         00002
      DATA TITLE/'ST  '/,CONT/'-CON'/,BLANK/'    '/                     00003
      DATA INN,IOUT/7,9/                                                00004
      REWIND IOUT                                                       00005
        PRINT 10
10      FORMAT('SUBROUTINE RU36')
      READ(INN,900,END=999) IELNO,DATA(1),DATA(2),DATA(3)               00007
001     CONTINUE
      READ(INN,910,END=990) CARDN,DATA(4)                               00008
      IF(CARDN .NE. CONT) GO TO 990                                     00009
      WRITE(IOUT) ISID,IELTYP,IELNO,DATA                                00010
C     READ(INN,920,END=999) TEMP                                        00011
C     BACKSPACE INN                                                     00012
        CALL BACKSP(TEMP,INN,&999)
        IF(TEMP(1).EQ.BLANK)
     $  READ(10,900,END=999)IELNO,DATA(1),DATA(2),DATA(3)
      IF(TEMP(1) .EQ. BLANK) GO TO 001                                  00013
      IF(TEMP(1) .NE. TITLE) GO TO 990                                  00014
  800 CONTINUE                                                          00015
      ENDFILE IOUT                                                      00016
      REWIND IOUT                                                       00017
      RETURN                                                            00018
  990 CONTINUE                                                          00019
      STOP 3600                                                         00020
  999 IEOF = 1                                                          00021
      GO TO 800                                                         00022
  900 FORMAT(I10,8X,3E18.6)                                             00023
  910 FORMAT(A4,14X,3E18.6)                                             00024
  920 FORMAT(2A2)                                                       00025
      END                                                               00026
C           DATA SET NASTRU37   AT LEVEL 004 AS OF 11/02/79
      SUBROUTINE RU37(ISID,IELTYP,IEOF)                                 00001
10      FORMAT('SUBROUTINE RU37')
      DIMENSION TEMP(2),DATA(20),KKREAD(33)
      DATA TITLE/'ST  '/,CONT/'-CON'/,BLANK/'    '/                     00003
      DATA INN,IOUT/7,8/                                                00004
      REWIND IOUT                                                       00005
      PRINT 10
      READ(INN,900,END=999) IELNO,DATA(1),DATA(2),DATA(3)               00007
001   CONTINUE
      READ(INN,910,END=990) CARDN,DATA(4),DATA(5),DATA(6)               00008
      IF(CARDN .NE. CONT) GO TO 990                                     00009
      READ(INN,910,END=990) CARDN,DATA(7),DATA(8),DATA(9)               00010
      IF(CARDN .NE. CONT) GO TO 990                                     00011
      READ(INN,910,END=990) CARDN,DATA(10),DATA(11),DATA(12)            00012
      IF(CARDN .NE. CONT) GO TO 990                                     00013
      READ(INN,910,END=990) CARDN,DATA(13),DATA(14),DATA(15)            00014
      IF(CARDN .NE. CONT) GO TO 990                                     00015
      READ(INN,910,END=990) CARDN,DATA(16),DATA(17),DATA(18)            00016
      IF(CARDN .NE. CONT) GO TO 990                                     00017
      READ(INN,910,END=990) CARDN,DATA(19),DATA(20)                     00018
      IF(CARDN .NE. CONT) GO TO 990                                     00019
      WRITE(IOUT) ISID,IELTYP,IELNO,DATA                                00020
C     READ(INN,920,END=999) TEMP                                        00021
C     BACKSPACE INN                                                     00022
      READ(INN,930,END=999)KKREAD
       REWIND 10
      WRITE(10,930)KKREAD
      REWIND
```

```
10
      READ(10,920)TEMP
      REWIND 10
      IF(TEMP(1) .EQ. BLANK)
     $ READ(10,900,END=999)IELNO,DATA(1),DATA(2),DATA(3)
      IF(TEMP(1).EQ.BLANK) GOTO 001                                    00024
      IF(TEMP(1) .NE. TITLE) GO TO 990                                 00025
  800 CONTINUE                                                         00026
      ENDFILE IOUT                                                     00027
      REWIND IOUT                                                      00028
      RETURN                                                           00029
  990 CONTINUE                                                         00030
      STOP 3700                                                        00031
  999 IEOF = 1                                                         00032
      GO TO 800                                                        00033
  900 FORMAT(I10,8X,3E18.6)                                            00034
  910 FORMAT(A4,14X,3E18.6)                                            00035
  920 FORMAT(2A2)
  930 FORMAT(33A4)                                                     00036
      END
C          DATA SET NASTRC36   AT LEVEL 025 AS OF 11/05/79            00001
      SUBROUTINE RC36(ISID,IELTYP,IEOF)
   10    FORMAT('SUBROUTINE RC36')                                     00002
      DIMENSION TEMP(2),DATAI(4),DATAR(4),RMAG(4),PHASE(4)             00003
      DATA TITLE/'$T   '/,CONT/'-CON'/,BLANK/'      '/                 00004
      DATA IPRT,INN,IOUT/6,7,9/
       PRINT 10                                                        00005
      IELCNT = 99                                                      00006
      RADDEG = 57.29578                                                00008
      READ(INN,900,END=999) IELNO,DATAI(1),DATAI(2),DATAI(3)
  001   CONTINUE                                                       00009
      READ(INN,910,END=990) CARDN,DATAI(4)                             00010
      IF(CARDN .NE. CONT) GO TO 990                                    00011
      READ(IOUT) ISIDR,IELTPR,IELNOR,DATAR                             00012
      IF(ISIDR .NE. ISID) GO TO 990                                    00013
      IF(IELTPR .NE. IELTYP) GO TO 990                                 00014
      IF(IELNOR .NE. IELNO) GO TO 990                                  00015
      DO 699 I = 1,4                                                   00016
        RMAG(I) = SQRT(DATAR(I)*DATAR(I) + DATAI(I)*DATAI(I))          00017
        IF(DATAR(I) .NE. 0.0) GO TO 690                                00018
        IF(DATAI(I) .EQ. 0.0) PHASE(I) = 0.0                           00019
        IF(DATAI(I) .GT. 0.0) PHASE(I) = 90.0                          00020
        IF(DATAI(I) .LT. 0.0) PHASE(I) = 270.0                         00021
        GO TO 699                                                      00022
  690   CONTINUE                                                       00023
        RATIO = ABS(DATAI(I)/DATAR(I))                                 00024
        PHASE(I) = ATAN(RATIO)*RADDEG                                  00025
        IF(DATAI(I).GE.0.0 .AND. DATAR(I).LT.0.0)                      00026
     X    PHASE(I) = PHASE(I) + 90.0                                  00027
        IF(DATAI(I).LT.0.0 .AND. DATAR(I).LT.0.0)                      00028
     X    PHASE(I) = PHASE(I) + 180.0                                 00029
        IF(DATAI(I).LT.0.0 .AND. DATAR(I).GT.0.0)                      00030
     X    PHASE(I) = PHASE(I) + 270.0                                 00031
  699 CONTINUE                                                         00032
C     WRITE(IPRT,930) ISID,IELTYP,IELNO,DATAR,DATAI                    00033
      IF(IELCNT .LT. 50) GO TO 700                                     00034
      CALL HD36(ISID)                                                  00035
      IELCNT = 0                                                       00036
  700 CONTINUE
      IELCNT = IELCNT + 1
```

```
                                    00037                                   00038
      WRITE(IPRT,940) IELNO,((RMAG(I),PHASE(I)),I=1,4)                      00039
C      READ(INN,920,END=999) TEMP                                           00040
C     BACKSPACE INN
        CALL BACKSP(TEMP,INN,&999)
        IF(TEMP(1).EQ.BLANK)
     $  READ(10,900,END=999)IELNO,DATAI(1),DATAI(2),DATAI(3)                00041
      IF(TEMP(1) .EQ. BLANK) GO TO 001                                      00042
      IF(TEMP(1) .NE. TITLE) GO TO 990                                      00043
      RETURN                                                                00044
  990 CONTINUE                                                              00045
      STOP 3601                                                             00046
  999 IEOF = 1                                                              00047
      RETURN                                                                00048
  900 FORMAT(I10,8X,3E18.6)                                                 00049
  910 FORMAT(A4,14X,3E18.6)                                                 00050
  920 FORMAT(2A2)                                                           00051
  930 FORMAT(1X,3I10,2(/,4(5X,1PE12.5)))                                    00052
  940 FORMAT(1X,I5,8X,4(1PE12.5,' /',0PF10.5,5X))                           00053
      END
C           DATA SET NASTRC37   AT LEVEL 022 AS OF 11/05/79                 00001
      SUBROUTINE RC37(ISID,IELTYP,IEOF)
 10     FORMAT('SUBROUTINE RC37')                                          00002
      DIMENSION TEMP(2),DATAI(20),DATAR(20),RMAG(20),PHASE(20)              00003
      DATA TITLE/'ST  '/,CONT/'-CON'/,BLANK/'    '/                         00004
      DATA IPRT,INN,IOUT/6,7,8/
       PRINT 10                                                             00005
      IELCNT = 10                                                           00006
      RADDEG = 57.29578                                                     00008
      READ(INN,900,END=999) IELNO,DATAI(1),DATAI(2),DATAI(3)
  001   CONTINUE                                                            00009
      READ(INN,910,END=990) CARDN,DATAI(4),DATAI(5),DATAI(6)                00010
      IF(CARDN .NE. CONT) GO TO 990                                         00011
      READ(INN,910,END=990) CARDN,DATAI(7),DATAI(8),DATAI(9)                00012
      IF(CARDN .NE. CONT) GO TO 990                                         00013
      READ(INN,910,END=990) CARDN,DATAI(10),DATAI(11),DATAI(12)             00014
      IF(CARDN .NE. CONT) GO TO 990                                         00015
      READ(INN,910,END=990) CARDN,DATAI(13),DATAI(14),DATAI(15)             00016
      IF(CARDN .NE. CONT) GO TO 990                                         00017
      READ(INN,910,END=990) CARDN,DATAI(16),DATAI(17),DATAI(18)             00018
      IF(CARDN .NE. CONT) GO TO 990                                         00019
      READ(INN,910,END=990) CARDN,DATAI(19),DATAI(20)                       00020
      IF(CARDN .NE. CONT) GO TO 990                                         00021
      READ(IOUT) ISIDR,IELTPR,IELNOR,DATAR                                  00022
      IF(ISID .NE. ISIDR) GO TO 990                                         00023
      IF(IELTYP .NE. IELTPR) GO TO 990                                      00024
      IF(IELNOR .NE. IELNO) GO TO 990                                       00025
      DO 699 I = 1,20                                                       00026
        RMAG(I) = SQRT(DATAR(I)*DATAR(I) + DATAI(I)*DATAI(I))               00027
        IF(DATAR(I) .NE. 0.0) GO TO 690                                     00028
        IF(DATAI(I) .EQ. 0.0) PHASE(I) = 0.0                                00029
        IF(DATAI(I) .GT. 0.0) PHASE(I) = 90.0                               00030
        IF(DATAI(I) .LT. 0.0) PHASE(I) = 270.0                              00031
        GO TO 699                                                           00032
  690   CONTINUE                                                            00033
        RATIO = ABS(DATAI(I)/DATAR(I))                                      00034
        PHASE(I) = ATAN(RATIO)*RADDEG                                       00035
        IF(DATAI(I).GE.0.0 .AND. DATAR(I).LT.0.0)                           00036
     X    PHASE(I) = PHASE(I) + 90.0                                       00037
        IF(DATAI(I).LT.0.0 .AND. DATAR(I).LT.0.0)
```

```
     X    PHASE(I) = PHASE(I) + 180.0                                  00038
          IF(DATAI(I).LT.0.0 .AND. DATAR(I).GT.0.0)                    00039
     X    PHASE(I) = PHASE(I) + 270.0                                  00040
  699 CONTINUE                                                         00041
C     WRITE(IPRT,930) ISID,IELTYP,IELNO,DATAR,DATAI                    00042
      IF(IELCNT .LE. 7) GO TO 700                                      00043
      CALL HD37(ISID)                                                  00044
      IELCNT = 0                                                       00045
  700 CONTINUE                                                         00046
      IELCNT = IELCNT + 1                                              00047
      DO 710 I = 1,5                                                   00048
        J = 4*(I-1) + 1                                                00049
        K = J + 3                                                      00050
        IF(I .EQ. 1) WRITE(IPRT,940) IELNO,I,                          00051
     X     ((RMAG(IX1),PHASE(IX1)),IX1=J,K)                          00052
        IF(I .NE. 1) WRITE(IPRT,950) I,                                00053
     X     ((RMAG(IX1),PHASE(IX1)),IX1=J,K)                          00054
  710 CONTINUE                                                         00055
      WRITE(IPRT,960)                                                  00056
C     READ(INN,920,END=999) TEMP                                       00057
C     BACKSPACE INN                                                    00058
        CALL BACKSP(TEMP,INN,&999)
        IF(TEMP(1).EQ.BLANK)
     $  READ(10,900,END=999)IELNO,DATAI(1),DATAI(2),DATAI(3)
      IF(TEMP(1) .EQ. BLANK) GO TO 001                                 00059
      IF(TEMP(1) .NE. TITLE) GO TO 990                                 00060
      RETURN                                                           00061
  990 CONTINUE                                                         00062
      STOP 3701                                                        00063
  999 IEOF = 1                                                         00064
      RETURN                                                           00065
  900 FORMAT(I10,8X,3E18.6)                                            00066
  910 FORMAT(A4,14X,3E18.6)                                            00067
  920 FORMAT(2A2)                                                      00068
  930 FORMAT(1X,3I10,10(/,4(5X,1PE13.6)))                              00069
  940 FORMAT(1X,I5,1X,I3,4X,4(1PE12.5,' /',0PF10.5,5X))                00070
  950 FORMAT(7X,I3,4X,4(1PE12.5,' /',0PF10.5,5X))                      00071
  960 FORMAT(' ')                                                      00072
      END                                                              00073
C          DATA SET NASTHD36   AT LEVEL 007 AS OF 10/24/79
      SUBROUTINE HD36(ISID)                                            00001
10      FORMAT('SUBROUTINE HD36')
      COMMON /HDRCOM/TITLE(16),SUBT(16),LABEL(16)                      00002
       PRINT 10
      IPRT = 6                                                         00003
      WRITE(IPRT,100) TITLE                                            00004
      WRITE(IPRT,110) SUBT                                             00005
      WRITE(IPRT,120) LABEL,ISID                                       00006
      WRITE(IPRT,140)                                                  00007
      WRITE(IPRT,150)                                                  00008
      WRITE(IPRT,160)                                                  00009
      WRITE(IPRT,170)                                                  00010
      RETURN                                                           00011
  100 FORMAT('1',3X,15A4,A2)                                           00012
  110 FORMAT(' ',3X,15A4,A2)                                           00013
  120 FORMAT('0',3X,15A4,A2,50X,'SUBCASE',I3)                          00014
  130 FORMAT(' ')                                                      00015
  140 FORMAT(27X,'S T R E S S E S   F O R   T H E   T R I A N ',       00016
```

```
     X          'G U L A R   R I N G S   ( C T R I A R G )')          00017
  150 FORMAT(61X,'(MAGNITUDE/PHASE)')                                 00018
  160 FORMAT(4X,'EL',17X,'RADIAL',19X,'CIRCUMFERENTIAL',              00019
     X 19X,'AXIAL',24X,'SHEAR')                                       00020
  170 FORMAT(4X,'ID',19X,'(X)',24X,'(THETA)',24X,'(Z)',               00021
     X 26X,'(ZX)')                                                    00022
      END                                                             00023
C          DATA SET NASTHD37   AT LEVEL 006 AS OF 10/24/79
      SUBROUTINE HD37(ISID)                                           00001
10      FORMAT('SUBROUTINE HD37')
      COMMON /HDRCOM/TITLE(16),SUBT(16),LABEL(16)                     00002
       PRINT 10
      IPRT = 6                                                        00003
      WRITE(IPRT,100) TITLE                                           00004
      WRITE(IPRT,110) SUBT                                            00005
      WRITE(IPRT,120) LABEL,ISID                                      00006
      WRITE(IPRT,140)                                                 00007
      WRITE(IPRT,150)                                                 00008
      WRITE(IPRT,160)                                                 00009
      WRITE(IPRT,170)                                                 00010
      RETURN                                                          00011
  100 FORMAT('1',3X,15A4,A2)                                          00012
  110 FORMAT(' ',3X,15A4,A2)                                          00013
  120 FORMAT('0',3X,15A4,A2,50X,'SUBCASE',I3)                         00014
  130 FORMAT(' ')                                                     00015
  140 FORMAT(27X,'S T R E S S E S   F O R   T H E   T R A P E ',      00016
     X          'Z O I D A L   R I N G S   ( C T R A P R G )')       00017
  150 FORMAT(61X,'(MAGNITUDE/PHASE)')                                 00018
  160 FORMAT(4X,'EL',2X,'ST',13X,'RADIAL',19X,'CIRCUMFERENTIAL',      00019
     X 19X,'AXIAL',24X,'SHEAR')                                       00020
  170 FORMAT(4X,'ID',2X,'PT',15X,'(X)',24X,'(THETA)',24X,'(Z)',       00021
     X 26X,'(ZX)')                                                    00022
      END                                                             00023
        SUBROUTINE BACKSP(TEMP,INN,*)
        DIMENSION KKREAD(33),TEMP(2)
        READ(INN,930,END=999)KKREAD
        REWIND 10
        WRITE(10,930)KKREAD
        REWIND 10
        READ(10,920)TEMP
        REWIND 10
  930   FORMAT(33A4)
  920   FORMAT(2A2)
        RETURN
  999   RETURN 1
        END
$
```

AN ENHANCEMENT OF NASTRAN FOR THE SEISMIC ANALYSIS OF STRUCTURES

John W. Burroughs
Civil Design Department, Ontario Hydro

SUMMARY

New modules, bulk data cards and DMAP sequence have been added to NASTRAN to aid in the seismic analysis of structures. These allow input consisting of acceleration time histories and result in the generation of acceleration floor response spectra. The resulting system contains numerous user convenience features, as well as being reasonably efficient.

INTRODUCTION

At ONTARIO HYDRO, the primary analysis tool is NASTRAN. This use began with the purchase of level 15.1 and continued with 15.5 and SPERRY/NASTRAN. Currently MSC/NASTRAN is being implemented as levels 16 and above are not available in CANADA. To perform a seismic analysis of nuclear power plant structures, the NASTRAN normal modes analysis has been utilized in conjunction with two post processors written at ONTARIO HYDRO. The one performs a time history method analysis to generate the desired floor response spectra, the other performs a response spectrum method analysis by utilizing these floor response spectra. While the response spectrum processor has stood the test of time, the time history method processor has not.

The time history analysis post processor was initially conceived to be used with simple stick type lumped mass models. However, as the complexity of the analyses increased, this processor was unable to satisfy all requirements. In addition, the cost of analysis for the more complex structures became excessive. About this time, the need to treat problems which were subjected to multiple-support excitations became a requirement.

Considering the shortcomings of the existing post processor, as well as future requirements, it was decided to develop an entirely new capability and to include it within NASTRAN. This project was then divided into two main development stages. The first, which is described in this paper, is a replacement for the existing post processor and provides the ability to handle any size problem, efficiency, improved output and user convenience features. The second stage, development of which is underway, extends the first stage to allow for the consideration of multiple-support excitation problems.

SYMBOLS

$a(t)$	-	acceleration time history at the ground (g).
b_j	-	structural damping ratio associated with a particular mode j.
$F_i(t)$	-	generated force time history acting at a specific degree of freedom.
g_e	-	structural damping ratio associated with the 1th element.
k_j	-	stiffness matrix associated with a specific mode j.
k_l	-	stiffness matrix associated with a particular element 1.
m_j	-	modal mass matrix associated with the jth mode.
m_i	-	total structural mass associated with grid point i.
$P_i(t)$	-	absolute acceleration time history used as a load for the response spectra generation.
β	-	equipment damping ratio for which spectra are required.
ξ	-	displacement components in modal co-ordinates.
ϕ_j	-	mode shape associated with the jth mode.
ω_o	-	input and modal frequencies at which the response spectra will be computed.

THEORY

Since the input for seismic activity is usually available as acceleration time histories, and force time histories are required, a conversion must be performed. Several techniques are available, the one chosen here replaces the acceleration time history by a set of force time histories according to the equation

$$F_i(t) = m_i\, a(t) \qquad (1)$$

This results in a force time history at each free degree of freedom corresponding in direction to the input acceleration.

Once the force time histories have been created, a modal transient analysis is performed. The resulting relative acceleration time histories

are then converted to absolute accelerations prior to the generation of the floor response spectra. This is done by adding the input acceleration to each of the computed acceleration time histories in the corresponding direction.

Floor response spectra are computed by performing a transient analysis for each of a set of one degree of freedom oscillators. The transient analysis performed utilizes the solution of separate second order differential equations of the following form.

$$\ddot{\xi}i + 2\beta\omega_o \dot{\xi}i + \omega_o^2 \xi i = \frac{1}{mi} Pi(t) \tag{2}$$

$$\omega_o \beta = \frac{bi}{2mi} \tag{3}$$

$$\omega_o^2 = \frac{Ki}{mi} \tag{4}$$

The frequencies ω_o utilized are a combination of user input frequencies and the frequencies determined for the structure. As results are required only in modal coordinates, the value of m_i is arbitrarily set to 1.0. This requires then only the solution of the equation

$$\ddot{\xi}i + 2\beta\omega_o \dot{\xi}i + \omega_o^2 \xi i = Pi(t) \tag{5}$$

The equation of motion, therefore, corresponds to that of a single degree of freedom system having the prescribed damping and frequency properties and subjected to the prescribed degree of freedom acceleration. The relative acceleration is obtained at each time step as follows:

$$\ddot{\xi}i,n+1 = \frac{Pi,n+1}{mi} - 2\beta\omega_o \dot{\xi}i,n+1 - \omega_o^2 \xi i, n+1 \tag{6}$$

To obtain the desired absolute accelerations, the computed acceleration is added to the above relative acceleration. The maximum acceleration is then retained over all time steps. This procedure is repeated for each of the designated frequencies and equipment damping input. The resulting table of maximum acceleration versus frequency is the desired floor response spectra.

Damping must be included in the analysis. Here, the user may specify modal damping, uniform structural damping or element structural damping. The preferred technique is to use element structural damping. In this case, the composite modal damping values will be computed. These values are based upon the fraction of the strain energy sustained by each element in the model. The modal damping for the jth mode is computed as follows:

$$b_j = \frac{\sum_{i=1}^{N} \{\phi_j\}^T K_l \{\phi_j\} g_l}{K_j} \qquad (7)$$

APPROACH

This facility has been implemented in NASTRAN by means of a DMAP program. This program is a modification of that used for Modal Transient Analysis (Rigid Format 12). The general problem flow is as shown in Figure 1.

To implement this facility, four new modules were written. Two of which precede the transient analysis module (TRD) and two follow. In addition, extensive use is made of existing NASTRAN modules in the solution.

The standard NASTRAN approach is followed for the matrix generation and eigenvalue extraction phase. Following this, the equivalent force time histories are created. The input acceleration time histories may come either from TABLED1 cards, or from a user file where the tables have been prestored. A modal transient analysis is then performed. The output from this stage consists of relative acceleration time histories. An added module will convert these into absolute accelerations. This matrix is subsequently transposed and from it the floor response spectra are generated. Finally, XY plots of the spectra may be produced. All normal NASTRAN output is available, in addition to the output produced by these new modules.

NASTRAN IMPLEMENTATION

To implement this capability in NASTRAN, two bulk data cards, four modules and a complete DMAP sequence have been developed. The new bulk data cards are SDATA and SET1.

The SDATA card, used to define the input loading and optionally to select the data required to generate the floor response spectra, is illustrated in the appendix. The SDATA card is selected by the DLOAD case control card. If the acceleration is to be combined with other acceleration or force time histories, then the DLOAD bulk data card may be used to combine them. Each SDATA card may select acceleration time histories for up to six degrees of freedom at any one grid point. In addition, data may be provided for the generation of floor response spectra. This data includes the equipment damping set and the set of grid points at which spectra are desired. Miscellaneous data for the control of the analysis may also be provided.

The SET1 card is used to define the grid points at which spectra are computed. It is selected by the SDATA card and is required only if floor response spectra are to be generated. The card format is shown in the appendix.

The acceleration time histories required may be supplied either as TABLED1 cards or from prestored tables on a user file. The latter technique is preferred when a standard set of time histories is available.

Four modules were created for this enhancement. They are SCNTL, SAPF, STHGNMX, and SFRG.

SCNTL - This module verifies all data input on SDATA cards and ensures that the required sets and data tables are available before proceeding with the analysis.

SAPF - This module accepts the input acceleration time history and generates the required force time histories. This is done by creating new forms of the DLT and DIT tables.

STHGNMX - This module accepts the relative time histories output from TRD and creates the absolute acceleration time histories. In addition, this module can, at user request, reduce the size of the output matrices prior to subsequent output requests.

SFRG - This module accepts the transposed absolute acceleration matrix and creates the required floor response spectra. This includes the generation of data for XY plots as well as printed output. This information is then passed to the XY PLOT and OFP modules.

In addition to these bulk data cards and modules, the DMAP sequence contains a number of parameters which may be used by the engineer to select optional processing paths. In general, the engineer need not use any of these parameters as the default values will select the most appropriate options.

All the normal output from a Modal Transient analysis may be requested. This includes both relative and absolute accelerations. The primary output, and all which is normally required, is the floor response spectra which may be printed or plotted as desired.

SAMPLE PROBLEM

To illustrate the ease with which this enhancement may be used, the structure shown in Figure 3 is analyzed and floor response spectra generated at grid point 6. As a point of comparison, the floor response spectra generated by the previous post processor is shown in Figure 4. That portion of the NASTRAN input data required for this enhancement is shown in Figures 5, 6 and 7 for the Executive control, Case control and Bulk data decks respectively. The plot of the resulting floor response spectra is shown in Figure 8 and a portion of the printed output in Figure 9. A review of this data and the resulting output demonstrates the ease with which floor response spectra may be generated.

EXTENSIONS

Several extensions to this enhancement are either planned or in progress. One will be to implement this facility in the other NASTRAN versions available at ONTARIO HYDRO. Another is the extension to solve multiple-support excitation problems. Upon completion of this last item, the original reason for developing these features will be satisfied. All future work will then add more user convenience features or efficiency improvements.

CONCLUSIONS

As a result of this enhancement of NASTRAN, an easy to use capability for the generation of floor response spectra has been made available. This is an extension of the existing Modal Transient analysis which retains all the original capabilities. In addition, the approach used is relatively efficient in terms of computer resources and user interaction required. It is a vast improvement over the original post processor and removes all of the restrictions inherent in it.

REFERENCES

1. The NASTRAN User's Manual NASA SP-222 (01), May 1973

2. The NASTRAN Theoretical Manual NASA SP-221 (01), December 1972

APPENDIX

Input data card SDATA

Description: Defines an input acceleration time dependent loading and various parameters for the generation of floor response spectra.

Format and example

1	2	3	4	5	6	7	8	9	10
SDATA	SID	RUN	XQT	DTM	SAVE	SOURCE	THN1	DIR1	+abc
SDATA	101	-1	3	2			BRTEQH	2	+SD1

+abc	IDFSP	IDEQ	NSKO	IDUMP	THN2	DIR2	THN3	DIR3	+def
+SD1	10	20							

+def		THN4	DIR4	THN5	DIR5	THN6	DIR6		

Field	Contents
SID	Set identification number (integer > 0)
RUN	Run type control parameter < 0 - simple structure > 0 - multi excitation
XQT	Erection phase control = 1 generate time histories only = 2 generate floor response spectra only = 3 both 1 and 2
DTM	Identification number of the grid point at which the acceleration time history is applied (integer > 0)

SAVE	A flag indicating that the output time histories are to be saved for subsequent use (integer > 0 or blank)
SOURCE	Identifier of the file on which the input time history is stored (integer ≥ 0, default = 0)
THNi	Name of the acceleration time history if on a file, or the id of a TABLED1 card.
DIRi	Direction associated with this time history (1 ≤ integer ≤ 6)
IDFSP	Set id of a SET1 card on which the points at which floor spectra as desired are listed (integer > 0 or blank)
IDEQ	Set id of FREQ, FREQ1 or FREQ2 cards on which the equipment dampings are defined (integer > 0 or blank)
NSKO	Alternate skip factor to reduce the output time histories by (integer > 0 or blank)
IDUMP	Intermediate output flag (integer > 0 or blank) = 1 print input tables = 2 print generated tables

Remarks:

1. The SDATA card must be selected by the DLOAD Case Control card.

2. This loading may be combined with other loadings by means of the DLOAD bulk data card.

3. The items SAVE and SOURCE refer to the NASTRAN GINO file INPT, INP1 thru INP9. INPT is denoted by zero, INP1 - INP9 by the integers 1 to 9.

4. If SAVE is blank, the output time histories will not be saved.

5. If any DIRi or THNi is left blank, then the scan of these items is terminated.

6. Up to 6 acceleration time histories at a single grid point may be defined on one logical card.

Input data card SET1

Description: Defines a set of grid points at which output is desired.

Format and example

1	2	3	4	5	6	7	8	9	10
SET1	SID	G1	G2	G3	G4	G5	G6	G7	+abc
SET1	10	6							

+abc	G8	-etc-							

Field	Contents
SID	Set identification number (integer > 0)
G1, G2 etc	List of structural grid points (integer > 0 or "THRU")

Remarks:

1. These cards are referenced by the SDATA card.
2. If "THRU" is used if must appear in field 4. Fields 6 to 10 will then be blank.
3. All points referenced within a THRU list must exist.

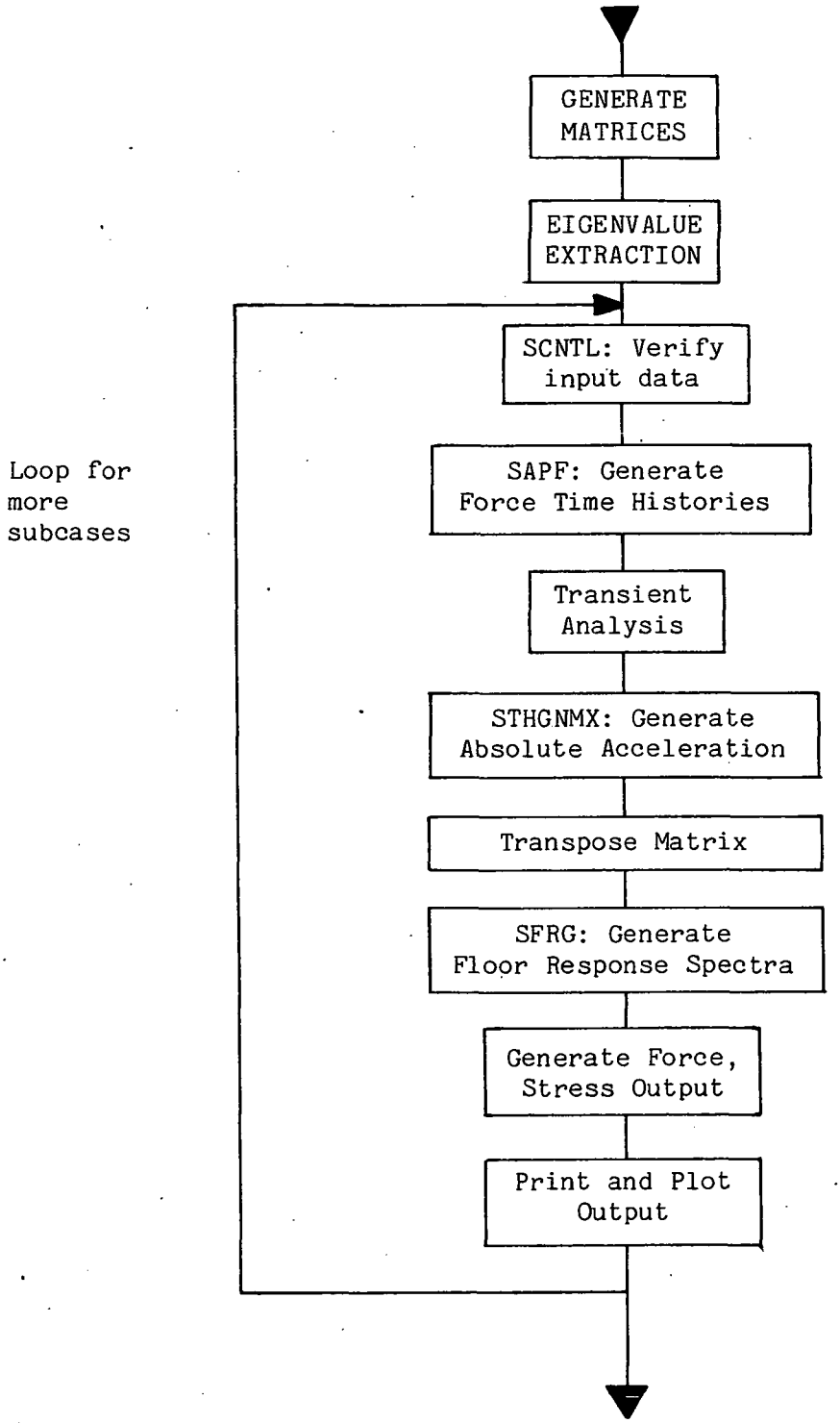


Figure 1: Generation of Floor Spectra Problem Flow

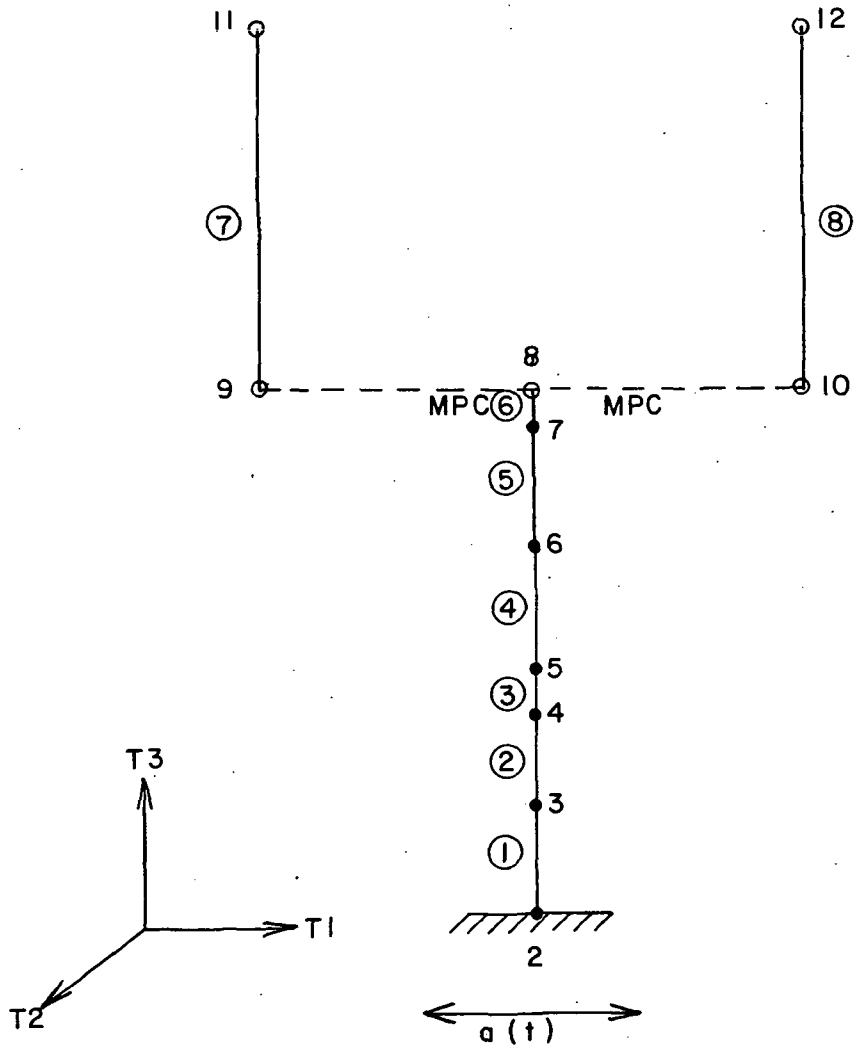


Figure 3: Sample Problem Geometry

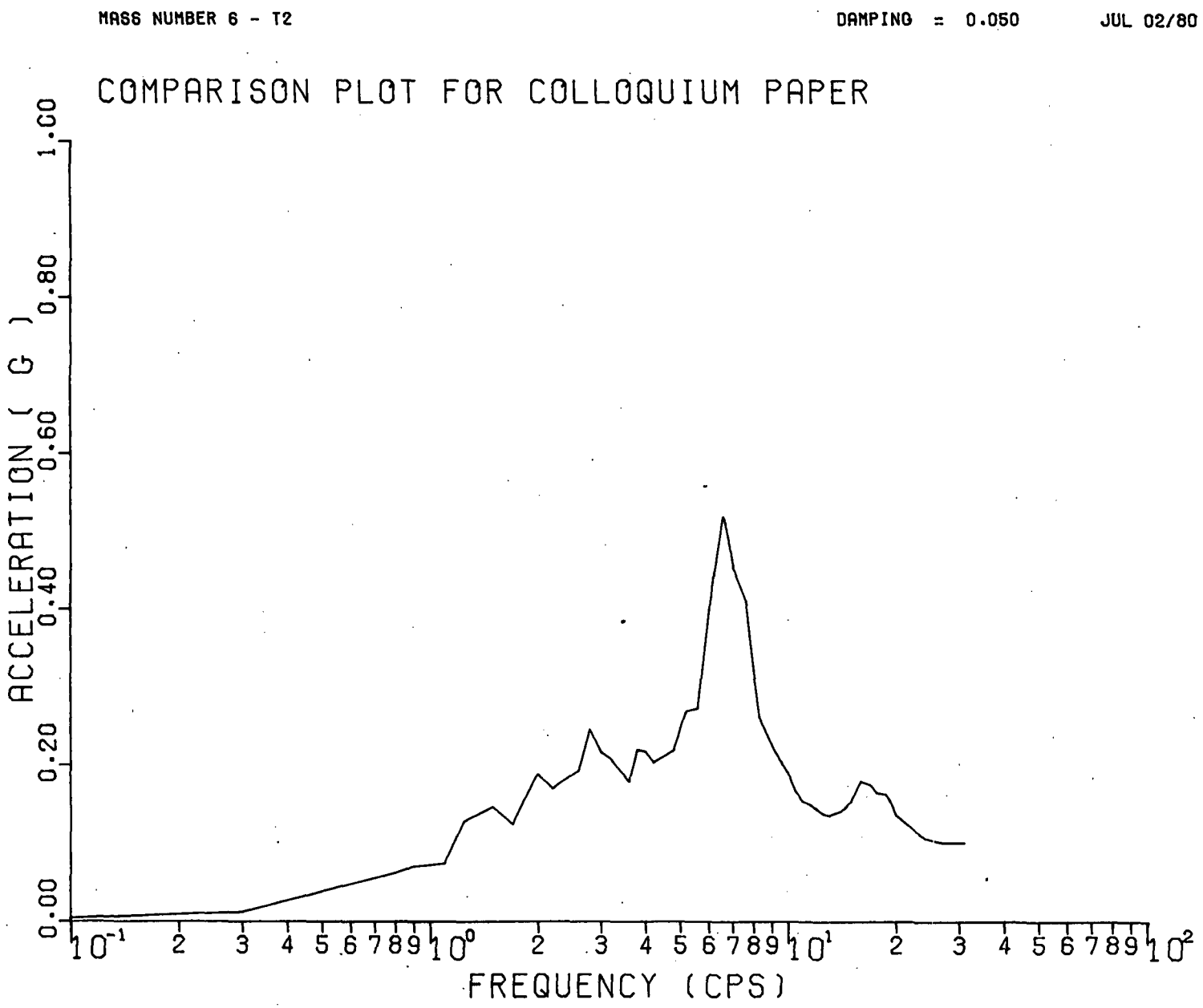


Figure 4: Comparison Floor Response Spectra

```
ID      A,B

APP     DMAP

TIME    10

BEGIN   $
$INSERT DMAP SEQUENCE HERE

END     $

CEND
```

Figure 5: Sample Executive Control Deck

```
TITLE      =    GENERATE FLOOR RESPONSE SPECTRA
TSTEP      =    50
METHOD     =    10
SPC        =    100
DLOAD      =    200
FREQ       =    201
OUTPUT (XYPLOT)
PLOTTER    =    NASTPLT MODEL D,O
```

Figure 6: Sample Case Control Deck

```
BEGIN BULK
   1        2         3         4         5         6         7         8         9        10
+--------+---------+---------+---------+---------+---------+---------+---------+---------+--------+
|        |         |         |         |         |         |         |         |         |        |
|FREQ    |    2    |     .01 |    .05  |         |         |         |         |         |        |
|FREQ    |  201    |     .1  |    .5   |    .9   |   1.2   |   1.7   |   2.0   |   2.5   |  +F1   |
|+F1     |    3.0  |    3.5  |   4.0   |   4.5   |   5.0   |   6.0   |   7.0   |   8.0   |  +F2   |
|+F2     |   10.0  |   12.0  |  14.0   |  20.0   |  28.0   |  30.0   |  35.0   |         |        |
|SDATA   |  200    |    -1   |    3    |    2    |   -1    |         |  BRTEH  |    2    |  +SD1  |
|+SD1    |    1    |    2    |         |         |         |         |         |         |        |
|SET1    |    1    |    6    |         |         |         |         |         |         |        |
|TSTEP   |   50    |  3000   |    .005 |    2    |         |         |         |         |        |
|        |         |         |         |         |         |         |         |         |        |
|$       |         |         |         |         |         |         |         |         |        |
|$       | ALL OTHER GEOMETRIC DATA    |         |         |         |         |         |        |
|$       |         |         |         |         |         |         |         |         |        |
|        |         |         |         |         |         |         |         |         |        |
|ENDDATA |         |         |         |         |         |         |         |         |        |
+--------+---------+---------+---------+---------+---------+---------+---------+---------+--------+
```

Figure 7: Sample Bulk Data Deck

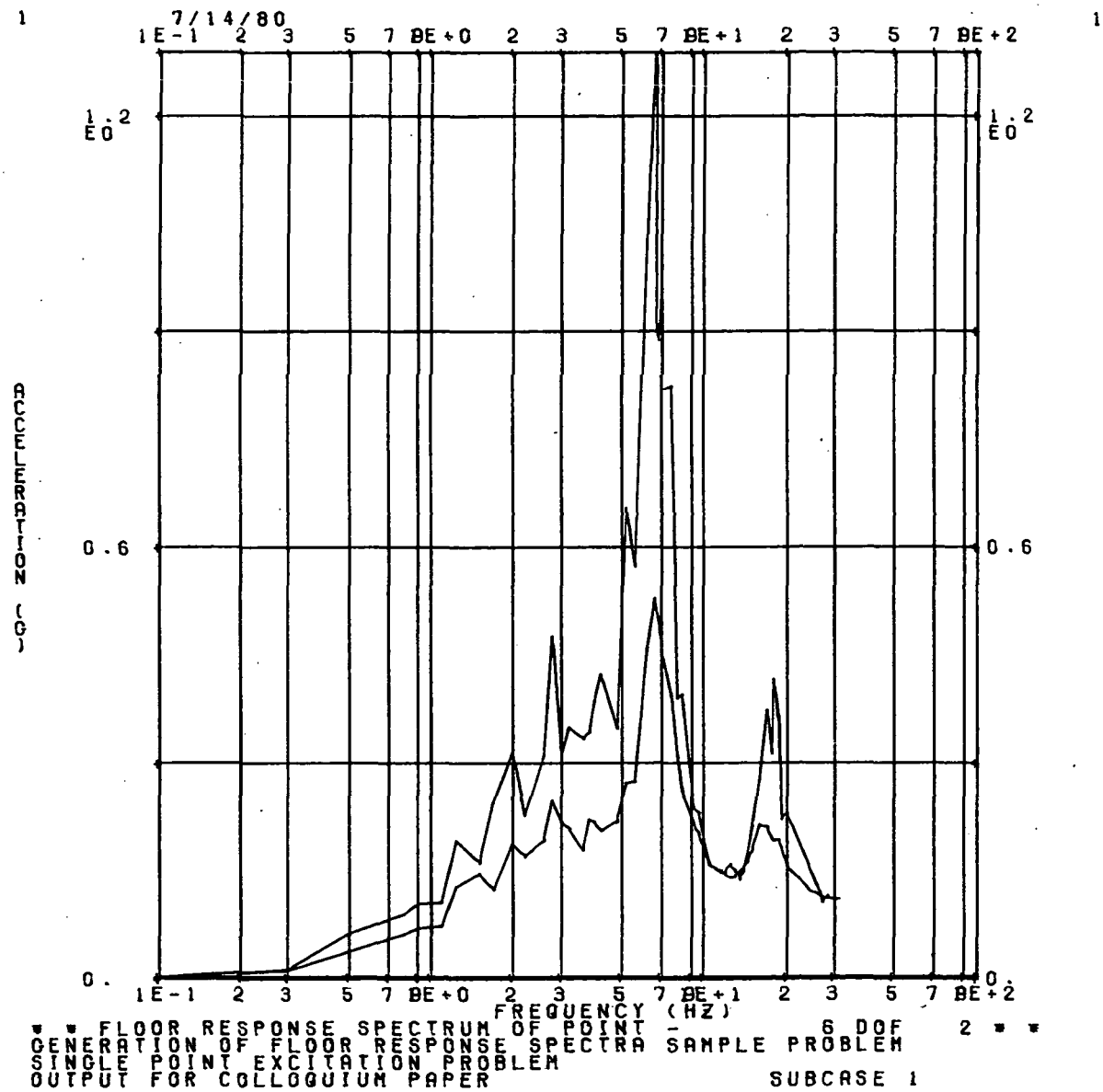


Figure 8: Generated Floor Response Spectra Plot

OUTPUT FOR COLLOQUIUM PAPER SUBCASE 1
POINT-ID = 6 DIRECTION = 2

FLOOR RESPONSE SPECTRA

FREQUENCY	TYPE	EQUIPMENT DAMPINGS .010	.050				
3.200000+00	G	3.499337-01	2.088033-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
3.600000+00	G	3.340834-01	1.794977-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
3.800000+00	G	3.436410-01	2.215128-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
4.000000+00	G	3.850916-01	2.188926-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
4.200000+00	G	4.236383-01	2.059928-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
4.800000+00	G	3.487423-01	2.203638-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
5.200000+00	G	6.550089-01	2.713436-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
5.600000+00	G	5.732045-01	2.750139-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
6.200000+00	G	1.040017+00	4.595057-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
6.600000+00	G	1.287466+00	5.287637-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
6.680144+00	G	9.027770-01	5.140159-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
6.800000+00	G	8.883788-01	4.994127-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
6.971636+00	G	9.545243-01	4.633139-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
7.000000+00	G	8.189829-01	4.544786-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0
7.600000+00	G	8.235375-01	3.937230-01	.0	.0	.0	.0
		.0	.0	.0	.0	.0	.0

Figure 9: Generated Floor Response Spectra - Printed

SOLUTION OF ENFORCED BOUNDARY MOTION IN DIRECT TRANSIENT AND HARMONIC PROBLEMS

Prepared By

Gary L. Fox

Director

Engineering Mechanics Division

NKF ENGINEERING ASSOCIATES, INC.
8150 Leesburg Pike
Vienna, Virginia 22180
(703) 442-8900

INTRODUCTION

The current versions of NASTRAN, i.e., NASA, MSC, and MAC support non-zero boundary displacements only in the static analysis. Forcing functions in the dynamic analysis formats allow only forces and pressures to exercise the mathematical model. This limitation can be circumvented by the application of a DMAP alter sequence. For the direct harmonic problem, a simple change to module FRRD can be easily incorporated to effect a more efficient use of the code.

Let the equation of motion be written with the dynamic set of coordinates in partition form with subscript b as the boundary set and subscript c as the complimentary boundary set, i.e.,

$$\begin{bmatrix} m_{cc} & m_{cb} \\ m_{bc} & m_{bb} \end{bmatrix} \begin{Bmatrix} \ddot{X}_c \\ \ddot{X}_b \end{Bmatrix} + \begin{bmatrix} d_{cc} & d_{cb} \\ d_{bc} & d_{bb} \end{bmatrix} \begin{Bmatrix} \dot{X}_c \\ \dot{X}_b \end{Bmatrix} + \begin{bmatrix} k_{cc} & k_{cb} \\ k_{bc} & k_{bb} \end{bmatrix} \begin{Bmatrix} X_c \\ X_b \end{Bmatrix} \qquad (1)$$

$$= \begin{Bmatrix} \bar{P}_c \\ P_b \end{Bmatrix} + \begin{Bmatrix} P_{nl} \\ 0 \end{Bmatrix}$$

where

m, d, k = mass, damping, and stiffness matrix coefficients

P, P_{nl} = linear and non-linear load vectors

Equation (1) is not solved by the direct transient or frequency formats when p_c, X_b, and therefore $\ddot{X}_b$ and $\dot{X}_b$, are known and P_b, X_c, and therefore $\ddot{X}_c$ and $\dot{X}_c$ are unknown. However, equation (1) can be rewritten in the form needed for solution by the standard NASTRAN modules. The first of these are:

$$[m_{cc}] \ [\ddot{X}] + [d_{cc}] \ [\dot{X}] + [k_{cc}] \ [X_c] = [P_c] + [P_{nl}] \qquad (2)$$

where

$$[P_c] = [\bar{P}_c] + [m_{cb}] \ [\ddot{X}_b] + [d_{cb}] \ [\dot{X}] + [k_{cb}] \ [X_b]$$

By the use of the partitioning modules, the submatrices in Equations (1) or (2) are easily formed. By letting the boundary displacement vector be input through the FORCE or DLOAD cards, the force vector is actually identified as $[P_b] = [X_b]$ (or the first or second derivatives).

The formation of the load vector is different for the transient and harmonic cases. These issues will be discussed below. Somewhat independent of the problem is the requirement that the solution vector to be processed by the remaining modules must be of the dimensions of the "d" set. By using once more partitioning vectors and the MERGE module, the solution vector $[X_c]$, and in the transient case $[\ddot{X}_c]$ and $[\dot{X}_c]$, is merged with the boundary vector $[X_b]$ to form the dynamic vector $[X_d]$. With the "d" set solution vectors formed, the remaining DMAP sequence can be executed without NASTRAN knowing the difference.

In the case of harmonic analysis the non-linear force is zero and equation (2) becomes

$$(w^2 \ [m_{cc}] + iw \ [d_{cc}] + [k_{cc}]) \ [X_c] = [P_c] \qquad (3)$$

where

w = circular frequency, $2 \pi f$.

HARMONIC ANALYSIS

The DMAP alter that was written to partition the matrix equation (1) into the form of equation (2) and then solve the lower order equation (3) is shown in Figure A-1. The following paragraphs discuss the steps involved.

1. FRRD calculates the load vector PDF and exits the module. The parameter ISKP is changed from -1 to a positive number to be transferred to FRRD the second time the module is executed. If the value of ISKP was set to zero, the default value, the module would have been executed normally. A normal

execution would give a solution to equation (1). The FORTRAN listing of module FRRD is shown in Figure A-2. The added code is underlined. Only the subroutines FRRD1A and FRRD1B are executed in this step.

2. The parameter ISKIP is saved for later use.

3. The partition vector DPAR is used to partition the stiffness matrix KDD. The submatrix identification is related to equation (2) by the following:

Figure A-1. DMAP Alter for Harmonic Response

```
ALTER  159.159
FRRD   CASEXX,USETD,DLT,FRL,GMD,GOD,KDD,BDD,MDD,DIT/UDVF,PSF,PDF,PPF/      (1)
         C,N,DISP/C,N,DIRECT/V,N,LUSETD/V,N,MPCF1/V,N,SINGLE/V,N,OMIT/
         V,N,NONCUP/V,N,FRQSET/V,N,ISKIP=-1/  $
SAVE   ISKIP  $                                                            (2)
PARTN   KDD,DPAR,/KD11,KD21,KD12,KD22/  $                                  (3)
PARTN   MDD,DPAR,/MD11,MD21,MD12,MD22/  $                                  (4)
PARTN   PDF,,DPAR/PD11,PD21,PD12,PD22/C,N,1  $                             (5)
MPYAD   KD11,PD21,PD11/P1DF/C,N,0/C,N,-1/  $                               (6)
FRRD   CASEXX,USETD,DLT,FRL,GMD,GOD,GOD,KD11,,MD11,DIT/UIDVF,PSF,P1DF,     (7)
       PPF/C,N,DISP/C,N,DIRECT/V,N,LUSETD/V,N,MPCF1/V,N,MPCF1/V,N,SINGLE/
       V,N,OMIT/V,N,NONCUP/V,N,FROSET/V,N,ISKIP/  $
MERGE   KD11,KD21,KD12,KD22,DPAR,/KDD/  $                                  (8)
MERGE   MD11,MD21,MD12,MD22,DPAR,/MOD/  $                                  (9)
MERGE   U1DVF,PD21,PD22,,OPAR/UDVF/C,N,1  $                               (10)

ENDALTER
CEND
```

Figure A-2. Listing of Module FRRD

LEVEL 2.2.1 (DFC 77)

```
ISN 0002        SUBROUTINE FRRD                                  00000010
                C                                                00000020
                C    FREQUENCY AND RANDOM RESPONSE MODULE        00000030
                C                                                00000040
                C    INPUTS CASECC,USETD,ULT,FRL,GMD,GOD,KDD,
                     BCD,MDD,PHIDH,DIT                           00000050
                C                                                00000060
                C    OUTPUTS UDV,PS,PD,MP                        00000070
                C                                                00000080
                C    8 SCRATCHES                                 00000090
                C                                                00000100
ISN 0003        INTEGER SINGLE,ONIT,CASECC,USETD,DLT,FRL,
                GMD,GOD,BDD,PHIDH,DIT, 1 SCR1,SCR2,SCR3,
                SCR5,SCR6,UDV,PS,PD,FP,PDD,OPTION                00000120
ISN 0004        INTEGER SCP7,SCRB,NAME&2<,MCB&7>                 00000130
```

```
ISN  0006        INTEGER FOL                                              00000140
ISN  0006        COMMON/APP&2<,MODAL&2<,LUSETD,MULTI,SINGLE,
                 OMIT, NONCUP,FRQSET,                                     00000150
               1 ISKIP                                                    00000155
             C                                                            00000160
ISN  0007        COMMON/FRRDST/OVF&150<,ICNT,IFRST,ITL&3<IDIT,
                 IFRD,K4DD                                                00000170
ISN  0008        DATA CASECC,USETD,DLT,FRL,GMD,GOD,KDD,HDD,
                 MOD,PHIDH,DIT/                                           00000180
                 1 101,102,103,104,105,106,107,108,109,110,111/           00000190
ISN  0009        DATA UDV,PS,PD,PP,PDD/201,202,203,204,203/               00000200
ISN  0010        DATA SCR1,SCH2,SCR3,SCR4,SCR5,SCR6 /301,302,
                 303,304,305,306/                                         00000210
ISN  0011        DATA SCR7,SSCR8/307.308 /                                00000220
ISN  0012        DATA MODA /4HMODA/                                       00000230
ISN  0013        DATA POL/205/                                            00000240
ISN  0014        DATA NAME /4HFRRD,4H /                                   00000250
               C                                                          00000260
               C   BUILD LOADS ON P SET ORDER IS  ALL FREQ.
                   FOR LOAD TOGETHER                                      00000270
               C                                                          00000280
ISN  0015        IF ( ISKIP .GE. 0 ) GO TO 5                              00000281
ISN  0017        NLOAD = ISKIP / 2*16                                     00000282
ISN  0018        NFREQ = ISKIP - NLOAD/*2**16                             00000283
ISN  0019        GO TO 15                                                 00000284
ISN  0020      5 CONTINUE                                                 00000285
ISN  0021        CALL FRPDIA&DIT,FRL,CASECC,DIT,PF,LUSETD,
                 NFREQ,NLOAD,FRQSET,FOL,                                  00000290
                 1 NOTRD<                                                 00000300
ISN  0022        1F&MULTI.LT.0.AND.SINGLE,LT.0.AND.OMIT.L.T.0             00000310
                 AND.MODAL
                 1 & 1< .NF. MODA< GO TO 60                               00000320
               C                                                          00000330
               C   REDUCE LOADS TO D OR H SET                             00000340
               C                                                          00000350
ISN  0024        CALL FPRU14$PP.USETD,GMD,MULTI,SINGLE,OMIT,              00000360
                 MODAL&1<,PH1DH,PD,
                 1 PS,SCR5,SCR1,SCR2,SCR3,SCR4<                           00000370
ISN  0025        IF ( ISKIP .LT. 0 ) GO TO 40                             00000375
ISN  0027        15 CONTINUE                                              00000377
ISN  0028        IF ( MULTI .LT. 0 .AND.SINGLE.LT.0 .AND.
                 OMIT .LT. 0                                              00000378
                 . .AND. MODAL(1) .NE. MODA ) POD = PD                    00000379
               C                                                          00000380
               C   SCR5 HAS PH IF MODAL FORMULATION                       00000390
               C                                                          00000400
ISN  0030        IF &MODAL&1< .EQ.MODA< PDD #SCR5                         00000410
               C                                                          00000420
               C   SOLVE PROBLEM FOR EACH FREQUENCY                       00000430
               C                                                          00000440
ISN  0032        IF&NONCUP .LT. 0 .AND. MODAL&1< .EQ. MODA<               00000450
```

```
                GO TO 50
ISN 0034      10 IF&FREQ .EQ. 1 .OR. NLOAD .EQ 1< SCR6 # UDV        00000460
ISN 0036         DO 20 1#1,NFPEQ                                   00000470
ISN 0037         CALL KLOCK&LOCK&ITIME1<                           00000480
                 C                                                 00000490
                 C   FORM AND DECOMPOSE MATRICES                   00000500
                 C                                                 00000510
ISN 0038         CALL FRRD1C&FRL,FROSET,MDD,RDD,KDD.1,SCR1,
                 SCR2,SCR3,SCR4,SCR8,
                 1 SCP7.1GOOD<                                     00000530
                 C                                                 00000540
                 C   ULL IS ON SCR1 -- LLL IS IN SCR2               00000550
                 C                                                 00000560
                 C   SOLVE FOR PD LOADS STACK ON SCR6               00000570
                 C                                                 00000580
                 C                                                 00000590
ISN 0039         CALL FRRD1D&PDD,SCR1,SCR2,SCR3,SCR4,SCR6,
                 NLOAD,1GOOD,NFREQ<                                00000600
ISN 0040         CALL KLOCK&ITIME2<                                00000610
ISN 0041         CALL IMTOGO&ITLEF1<                               00000620
ISN 0042         IF&2*&ITIME2-ITIME1<.GT. ITLEFT .AND. I .NE.      00000630
                 NFREQ< GO TO 70
ISN 0044      20 CONTINUE                                          00000640
ISN 0045         1 # NFREQ                                         00000650
ISN 0046      30 CONTINUE                                          00000660
ISN 0047         IF&NFREQ .EQ. 1 .OR. NLOAD .EQ 1< GO TO 40        00000670
                 C                                                 00000680
                 C   RESORT SOLUTION VECTORS INTO SAME ORDER AS LOADS 00000690
                 C                                                 00000700
ISN 0049         CALL FRRD1E&SCR6,UDV,NLOAD,I<                     00000710
ISN 0050         40 ISKIP = NFREQ +NLOAD*2**16                     00000720
ISN 0051         RETURN                                            00000725
                 C                                                 00000730
                 C   UNCOUPLED MODAL                               00000740
                 C                                                 00000750
ISN 0052         50 CALL FRRD1F&MDD,HDD,KDD,FRL,FRQSET,NLOAD,
                 NFREQ,PDD,UDV<                                    00000760
ISN 0053         GO TO 40                                          00000770
ISN 0054         60 PDD # PP                                       00000780
ISN 0055         GO TO 10                                          00000790
                 C                                                 00000800
                 C   INSUFFICIENT TIME TO COMPLETE ANOTHER LOOP     00000810
ISN 0056         70 CALL MESAGE&.5.NFREQ-I,NAME<                   00000820
ISN 0057         MCA&1< # SCR6                                     00000830
ISN 0058         CALL RDTFL&MCA*1<<                                00000840
ISN 0059         MDONE # MCD&2<                                    00000850
ISN 0060         MCR&1< # PP                                       00000860
ISN 0061         CALL ROTR1&MCH&1<<                                00000870
ISN 0062         MCR&2< NOONF                                      00000880
ISN 0063         CALL WRT1FL&MCB&1                                 00000890
```

```
ISN  0064      IF&SINGLE .LT. 0< GO TO 80                          00000900
ISN  0066      MCA&1< # PS                                         00000910
ISN  0067      CALL PUTRL&MCA&1<<                                  00000920
```

K_{dd} = KD11
K^{cb} = KD12
K_{bc} = KD21
K_{bb} = KD22

4. The partition of the mass matrix, MDD, is similar to the stiffness matrix.

5. Because the load vector is calculated for all frequencies and loading conditions at once, PDF is a load matrix, a load vector in each column. The partition vector DPAR is used again to separate the enforced displacements from the forces. The relationship to equation (2) is

P^c = PD11
P_b = PD21

6. The module MPYAD performs the matrix multiplication and additions required by equation (2). Here

P_c = P1DF

7. Module FRRD is executed again, but this time the parameter ISKIP is positive. A jump to statement 15, underlined in Figure A-2, causes only the subroutines FRRD1C, FRRD1E and FRRD1F to be executed. The solution to equation (3) is obtained in this step. The code uses the following names related to equation (3).

M = MD11
K^{cc} = KD11
P^c = P1DF
X^c_c = U1DF

8. The stiffness matrices are merged to form the system stiffness matrix. This is the inverse of operation 3.

9. Similar to the stiffness matrix, this operation is the inverse of operation 4.

10. Merges the solution vector X_c of equation (6-7) with X_b to form the system solution vector X_d.

The three merges, operations 8, 9, and 10, are made necessary because NASTRAN uses the displacement approach to the problem solution. In order to calculate stress and forces in the members, the solution vector must contain all grid points.

TRANSIENT ANALYSIS

The DMAP Alter required for the Rigid Format 9, Direct Transient Response, is shown in Figure A-3. The discussions below relates to the circled numbers in the DMAP listing.

1. The Stiffness matrix is partitioned in accordance with Equation (2) where

$$KD11 = K_{cc}$$
$$KD12 = K_{cb}$$
$$KD21 = K_{bc}$$
$$KD22 = K_{bb}$$

2. The Mass matrix is partitioned similar to the Stiffness matrix

$$[MDD] = \left[\begin{array}{c|c} MD11 & MD12 \\ \hline MD21 & MD22 \end{array}\right]$$

Figure A-3. DMAP Alter to Rigid Format 9

```
ALTER  163
PARTN  KDD,OPAR,/KD11,KD21,KD12,KD22/  $                                    (1)
PARTN  MDD,OPAR,/MDLL,MD21,MD12,MD22/  $                                    (2)
PARTN  PD,OPAR/PD11,PA21,PD12,PD22/C,N,1  $                                 (3)
MPYAD  PA21,MV1,/PBT21/C,N,0/C,N,1/C,N,0/C,N,2  $                    (4)    (5)
ADD    PBT21,PA21/PB21/C,Y,ALPHA=(0.550E-2.0)/C,Y,BETA=(0.550E-2.0)$        (6)
MPYAD  PB21,MAIT,/PV21/C,N,0/C,N,1/C,N.0/C,N,2  $                           (7)
MPYAD  PV21,MV1,/PCT21/C,N,0/C,N,1/C,N,0/C,N,2  $                           (8)
ADD    PCT21,PV21/PC21/C,Y,ALPHA=(0.550E-2.0.)/C,Y,BETA=(0.550E-2.0)$       (9)
MPYAD  PC21,MAIT,/PU21/C,N,0/C,N,1/C,N,0/C,N,2  $                          (10)
MPYAD  KD12,PU21,PD11/P1D/C,N,0/C,N,1 /$                                   (11)
ALTER  165,165
TRD    CASEXX,TRL,NLFT,DIT,KD11,MD11,PID/UIDVT,P1LD/C,N,DIRECT/
       V,N,NOUE/V,N,NONCUP/V,N,NCOL $                                      (12)
ALTER  166
MERGE  KD11,KD21,KD12,KD22,OPAR/KDD/  $                                    (13)
MERGE  MD11,MD21,MD12,MD22,OPAR,/MDD/  $                                    (14)
MERGE  PD11,PILD,PD12,PD22,,OPAR/PNLD/C,N,1  $                              (15)
PARTN  PA21,PVA,/A21,,PDA12,/C,N,1  $                                       (16)
PARTN  PV21,PVA,/V21,,PDA12,/C,N,1  $                                       (17)
PARTN  PU21,PVA,/U21,,PDA12,/C,N,1  $                                       (18)
MERGE  A21,,V21,,PVVA,/PVA21/C,N,1  $                                       (19)
MERGE  PVA21,,U21,,PVUVA,/PUVA21/C,N,1  $                                   (20)
MERGE  U1DVT,PUVA21,,,,DPAR/UDVT/C,N,1  $                                   (21)
ENDALTER
```

3. The load vector, PD, which is output from module TRLG, is partitioned according to Equation (2), where

$$PD = \{P(t_1)\}, \{P(t_2)\}, \ldots$$

$$PD11 = \{\bar{P}_c(t_1)\}, \{\bar{P}_c(t_2)\}, \ldots$$

$$PA21 = \{P_b(t_1)\}, \{P_b(t_2)\}, \ldots$$

Note that PD is a matrix formed by columns of load vectors, one column for each time step. The matrices PD22 and PD12 are not generated, i.e.

$$PD = \begin{bmatrix} PD11 \\ \hline PA21 \end{bmatrix}$$

4. Direct input matrices, MV1 and MA1T, are used subsequently to calculate the velocity and displacement matrices from the acceleration matrix. The forms of MV1 AND MA1T are

$$MV1 = \begin{bmatrix} 0 & 1 & 0 & 0 & \ldots \\ 0 & 0 & 1 & 0 & \ldots \\ 0 & 0 & 0 & 1 & \ldots \\ \ldots & & & & \\ \ldots & & & & \\ \ldots & & & & \end{bmatrix} \quad \begin{matrix} \uparrow \\ M \\ \downarrow \end{matrix}$$

$$\leftarrow N+2 \rightarrow$$

$$MA1T = \begin{bmatrix} 1 & 1 & 1 & 1 & \ldots \\ 0 & 1 & 1 & 1 & \ldots \\ 0 & 0 & 1 & 1 & \ldots \\ \ldots & & & & \\ \ldots & & & & \\ \ldots & & & & \end{bmatrix} \quad \begin{matrix} \uparrow \\ M \\ \downarrow \end{matrix}$$

$$\leftarrow N+2 \rightarrow$$

The dimensions of both matrices are M X N + 2 where M is the number of coordinates in the b-set and N is the number of time steps.

5. Produces the matrix product

$$[PBT21] = [PA21]*[MV1]$$

$$= [\{P_b(t_1)\}, \{P_b(t_2)\}, \ldots] \begin{bmatrix} 0 & 1 & 0 & 0 & \ldots \\ 0 & 0 & 1 & 0 & \ldots \\ 0 & 0 & 0 & 1 & \ldots \\ \ldots & & & & \\ \ldots & & & & \\ \ldots & & & & \end{bmatrix}$$

$$= [0, \{P_b(t_1)\}, \{P_b(t_2)\}, \ldots]$$

It is seen that this operation moves the columns of the acceleration

vectors from time t_i to t_i+1.

6. Produces the matrix sum

$$[PB21] = \alpha[PBT21] + \beta[PAZ1]$$

The coefficients α and β are set equal to one-half of the integration time step, Δt.

$$[PB21] = \frac{\Delta t}{2}[\{P_1 + P_2\}, \{P_2 + P_3\}, \ldots]$$

$$=[\{\Delta V_1\}, \{\Delta V_2\}, \ldots]$$

where $[P_i] = \{P_c(t_i)\}$; i = 1 to N + 2

The matrix PB21 represents the change in velocity, ΔV_i, between time steps, t_i and t_{i+1}. The calculation is based on the trapesoidal rule for numerical integration.

7. The final step in producing the matrix of velocity vectors, PV21 from the matrix of acceleration vectors, PA21, this module produces the matrix product

$$[PV21] = [PB21]\,[MA1T]$$

$$= [\Delta V_1\}\ , \{\Delta V_2\}, \ldots] \qquad \begin{bmatrix} 1 & 1 & 1 & 1 & . & . \\ 0 & 1 & 1 & 1 & . & . \\ 0 & 0 & 1 & 1 & . & . \\ 0 & 0 & 0 & 1 & . & . \\ . & . & . & . & & \\ . & . & . & . & & \end{bmatrix}$$

$$= [\{\Delta V_1\ \}, \{\Delta V_1 + \Delta V_2\}, \{\Delta_1 + \Delta_2 + \Delta V_3\}, \ldots]$$

8., 9. and 10. A repeat of operations e, f, g. The matrix of displacement PU21, is calculated from the matrix of velocity vectors, PV21.

11. The load vector is calculated in accordance with Equation (2).

KD12 = K_{cb}

PU21 = $\{X_b\}_1, \{X_b\}_2, \ldots$

PD11 = $\{\bar{P}_b\}_1, \{\bar{P}_b\}_2, \ldots$

P1D = $\{P_c\}_1, \{P_c\}_2, \ldots$

12. The module TRD calculates the solution to Equation (2).

KD11 = [Kcc]

MD11 = [Mcc]

P1D = $P_1, P_2, \ldots$

$$U1DVT = \begin{bmatrix} X & X & \\ \vdots & \vdots & \\ X & X & \\ \vdots\vdots & \vdots\vdots & \\ X_{1}, & X_{2}, & \cdots \end{bmatrix}$$

$$[P1LD] = \{P_{\eta\ell}\} .$$

The solution vector, U1DVT, is a matrix of displacements, velocity and acceleration vectors for each grid point; a column for each time step.

13. The system stiffness matrix is formed

$$\left[\begin{array}{cc} KD11 & KD12 \\ \hline KD21 & KD22 \end{array}\right] = [KDD]$$

14. The system mass matrix is formed similar to the operation (13.)

15. The system load vector is formed

$$\left[\begin{array}{c} PD11 \\ \hline P1LD \end{array}\right] = [PNLD]$$

16., 17. and 18. Partition the acceleration, PA21, velocity, PV21, and displacement, PU21, matrices to the correct size to be merged with U1DVT.

19., 20. and 21. These operations merge the solution matrix, UD1VT, with the excitation matrix, PUVA21, to form the final system solution matrix, UDVT.

$$\left[\begin{array}{c} UD1VT \\ \hline PUVA21 \end{array}\right] = [UDVT]$$

From this point on, the solution is calculated according to the Standard Rigid Format 9 procedure.

APPLICATION OF NASTRAN TO THERMAL TRANSIENT ANALYSIS WITH SURFACE ABLATION*

Karl Meyer
Planning Research Corporation
John F. Kennedy Space Center, Florida

SUMMARY

This paper presents a computer modeling technique for solving thermal transient analysis (Solution 9, Approach Heat) with surface ablation problems using the NASTRAN Computer Program.

The mathematical model used in this analysis is one dimensional, which corresponds to the direction of heat flow. All dimensions perpendicular to that of the heat flow direction are assumed to be in thermal equilibrium, i.e., the net heat flow in and out is assumed to be zero. A special modeling technique that would simulate the effects of surface ablation corresponding to the direction of heat flow was developed and incorporated into this mathematical model.

All computer analyses were done using Level 16.0 NASTRAN on a UNIVAC System. This analysis technique was developed to predict the thermal response and the amount of surface ablation that would occur on the deck of the Mobile Launcher Platform (MLP) during a launch of the Space Shuttle at John F. Kennedy Space Center, Florida.

INTRODUCTION

The similarities between heat transfer and structural analysis have been exploited to include heat transfer as a part of NASTRAN analysis capabilities.

In heat transfer, just as in structural mechanics, the analysis of a solid continuum can be modeled using the finite element method. Finite element analysis reduces the problem to the solution of a set of simultaneous equations in which the unknown variables to be calculated are defined at a discrete or finite set of grid points. This set of simultaneous equations may then be expressed in the matrix form:

$$[K]\{u\} + [B]\{\dot{u}\} = \{P\} + \{N\} \text{ (ref. 1)}$$

---

*This work was done under NASA contract No. NAS10-8525.

Table I shows the analogies between heat transfer and structural mechanics. The major difference between heat transfer and structural mechanics analysis is that temperature is a scalar function of position, whereas displacement is a vector with as many as six component degrees of freedom. Thus, in heat transfer analysis, NASTRAN provides one degree of freedom at each grid point (temperature).

The heat conduction matrix [K], and the heat capacity matrix [B], are derived from element and material properties assigned by the programmer. An additional advantage of NASTRAN is that part of the heat conduction matrix may be associated with surface heat convection or radiation. The applied heat flux vectors {P}, are associated with heat flux in or out and are time-dependent loads. The nonlinear heat flux vectors {N} are temperature dependent or rate-of-change-of-temperature dependent heat flux in or out of the system.

Typical output from a NASTRAN thermal transient analysis includes:

a. Temperature vs time or grid points

b. Rate of change of temperature vs time at grid points

c. Thermal flux vs time at grid points

d. Finite element temperature gradient vs time

SYMBOLS

[B]	Matrix of constant heat capacitance coefficients
[K]	Matrix of constant heat conductance coefficients
{N}	Matrix of nonlinear thermal flux which are functions of temperature of the rate of change of temperature
{P}	Matrix of thermal flux are functions of time.
{u}	Matrix of temperature at grid points
$\{\dot{u}\}$	Matrix of the rate of change of temperatures with respect to time at grid points
A	Unit cross-sectional area
F	Radiation factor
S	Arbitrary scale factor

b	Thermal capacity
c	Specific heat
h	Convective film coefficient
k	Thermal conductivity
Δx	Length of element x = .0529 cm (1/48 inch)
ρ	Density
σ	Stefan-Boltzmann constant
f( )	Function of a variable
c	Second component CELASi element or point
i	The i^{th} grid point or element
in	Thermal flux in
out	Thermal flux out
TS	Thermal switch
1,2, etc.	Grid point or finite element number

NASTRAN MODELING PROBLEM

When analyzing a thermal transient heat transfer problem using a finite element program such as NASTRAN in which surface ablation is present, a modeling problem occurs. The problem is that when an element is ablated from the material being analyzed it must be eliminated from the analysis. In the present configuration of NASTRAN, it is not possible to eliminate an element or elements in the middle of the computations and continue on to the completion of the computations for a complete thermal transient analysis.

To solve this problem, a NASTRAN computer model was developed consisting of a one-dimensional isotropic material or materials in which transient transport of thermal energy occurs, with ablation from the front surface. All dimensions perpendicular to that of heat flow direction consist of isotropic material(s) which is assumed to be in thermal equilibrium, i.e., the net heat flow in and out is assumed to be zero.

This NASTRAN model uses a special modeling technique to simulate the removal of elements caused by surface ablation. The modeling technique

involves the use of NASTRAN nonlinear load bulk data cards. These nonlinear loads essentially cancel an element from the analysis when it is ablated and relocate the thermal flux into the model to the next prescribed grid point.

NASTRAN THERMAL MODEL

Refer to figure 1 (NASTRAN Thermal Model) in the following discussion.

To model the thermal properties of the material being analyzed, NASTRAN CROD elements connected in series were used. This series connection of CROD elements describes the one-dimensional thermal properties of the material or materials being analyzed with respect to depth. For most heat conduction problems, 20 elements per centimeter (48 elements per inch) of thickness are needed to represent accurately the component being analyzed (ref. 2). Refering to figure 1, the thermal conductors K_1 to $K_{(i-1)}$ and thermal capacitors B_1 to B_i are modeled using CROD elements 1 to (i - 1). The CROD elements reference PROD property cards in which a unit area (A_j) of one was used. The PROD property card references MAT4 material property card. The MAT4 material property card requires the thermal conductivity (k_j) and the thermal capacity (b_j) of the material(s) being analyzed. NASTRAN calculates the thermal conductance and thermal capacitance for each element using the following relationships:

$$K_j = k_j A_j / \Delta x$$

and $B_j = b_j A_j \Delta x / 2$ (lumped method)

where $b_j = \rho i c_j$

The backface grid point $_j$ has a CHBDY element (convective film coefficient h) which references a temperature at grid point r. The CHBDY element references a PHBDY property card where an area factor of one is specified. The PHBDY card references a MAT4 material where the convective film coefficient is specified.

To model the ablation of elements from the model, a concept was developed which I will call a thermal switch. The thermal switch is an element that has an output response of zero or one, depending upon some predefined ablation

temperature at a reference grid point in the material model. The thermal switch was modeled using grounded CELAS3 elements. There is one thermal switch for each point in material model. These thermal switches are not physically connected to the grid points in the material model. The response temperature of each thermal switch is made a function of the temperature at a reference grid point using NOLIN1 nonlinear loads. The NOLIN1 nonlinear load applies a thermal load N at the ungrounded point of the CELAS3 element, which is a function of a temperature at the reference grid point and a table (TABLEDi) bulk data card. The response temperature at the ungrounded point of the CELAS3 is given by the following relationship:

$$u = \frac{N}{K}$$

It is evident from this equation that if a thermal load N of zero is applied when the temperature at the reference grid point is below the ablation temperature, the temperature response u will be zero. When the temperature of the reference grid point is equal to or greater than the ablation temperature, a thermal load of N = K is applied. The resulting response temperature u at the ungrounded point of the CELAS3 element will be equal to one. Essentially, this modeling technique makes the CELAS3 element an on/off switch which is a function of temperature at a reference grid point in the material being analyzed.

When the response temperature of the CELAS3 element ungrounded point is used in conjunction with NOLIN2 nonlinear loads and other CELASi elements, all effects of ablation can be modeled. The response of the other CELASi ungrounded points are used as the second components, either temperature dependent or rate-of-change-of-temperature dependent, in the NOLIN2 nonlinear load cards. The first component response for NOLIN2 nonlinear load card is the response temperature of the ungrounded point of the respective thermal switch. The general equation for the NOLIN2 nonlinear load card is:

$$N_i = S_i \cdot f(u_{TS_i}) \cdot f(u_{Ci}) \quad \text{(ref. 3)}$$

where: S_i is an arbitrary scale factor for the $_i$th point, $f(u_{TS_i})$ is the response of the $_i$th thermal switch, $f(u_{Ci})$ is the second component response of the $_i$th CELASi element ungrounded point.

The second component response of the unground point of the CELASi element can be used to generate thermal loads such as:

$$f(u_{ci}) = \frac{N_{ci}}{K_{ci}}$$

where, $N_{ci} = \pm F\sigma u^4$ for radiation thermal loads in or out

$N_{ci} = \pm hu$ for convective thermal loads in or out

$N_{ci} = \pm Ku$ for conduction thermal loads in or out

$N_{ci} = \pm B\dot{u}$ for capacitance thermal loads in or out

When the second component response is used in conjunction with the thermal switch response along with the NOLIN2 nonlinear load card, the resulting loads can be turned on or off based on some predefined ablation temperature.

The ablation effects that can be modeled using this modeling technique include:

a. Cancelling of the original thermal flux in and relocating the thermal flux to the next prescribed grid point

b. Cancelling the thermal conductance effects of the ablated element by applying an equal and opposite thermal load at the next grid point of:

$$N_{out_i} = -K_i \cdot (u_{i-1} - u_i) \cdot f(u_{TS_i})$$

c. Cancelling the thermal capacitance effects of the ablated element by applying an equal and opposite thermal load at the next grid point of:

$$N_{out_i} = -B_i \dot{u}_{i-1} \cdot f(u_{TS_{i-1}})$$

NOTE

This effect is negligible and can be assumed to be zero, because the value of $\dot{u}_i$-1 approximately equals zero when the other ablation effects are taken into consideration. Also, this type of nonlinear load cannot be used in NASTRAN's present configuration. (Refer to NASTRAN PROGRAM RECOMMENDATIONS section of this report.)

Again, all these ablation effects can be modeled with thermal switches in combination with NOLIN1 through NOLIN4 nonlinear loads and other grounded CELASi elements. One should note that when CELASi elements are used to provide non-zero temperature responses, a large thermal conductive to ground is required. For the relationship N = Ku, u must be adjusted to the desired temperature by defining the thermal conductance, K, of the CELASi element, which is connected to ground, and a load N, which is applied to the grid or scalar point in question. The numerical value of K should be several orders of magnitude greater than the numerical value of the thermal conductance specified in the rest of the model.

Figure 2 shows a schematic diagram of all thermal load equations applied in a typical ablation program. The equations on the right of figure 2 are used for applying and cancelling thermal flux in after the ablation process occurs. Initially, N_{in1} is applied at grid point 1. When grid point 1 reaches ablation temperature, N_{out1} is turned on by thermal switch 1 (CELAS3 element) and this thermal load is applied to grid point 1. The value of N_{out1} is equal and opposite to that of N_{in1}, essentially cancelling the thermal flux into grid point 1. At the same time N_{out1} is turned on N_{in2}, which is equal to N_{in1}, is turned on, thus moving the input thermal flux from grid point 1 to grid point 2. This process is repeated for grid point 2 to grid point i, when and if each respective grid point reaches ablation temperature.

The equation on the left of figure 2 are used for cancellation of element thermal effects after ablation of the respective element(s) occurs. Initially, none of these thermal loads are applied until the ablation process occurs. When grid point 1 reaches ablation temperature, N_{out2} is turned on by thermal switch 1 and this thermal load is applied to grid point 2. The N_{out2} thermal flux cancels the thermal flux of B_1 and K_1. This process is repeated for grid points 3 to grid point i, when and if each respective grid point reaches ablation temperature.

Multilayer surfaces are modeled in the same manner and are composed of layers of materials having different thicknesses and thermal properties.

NASTRAN PROGRAM EXECUTION

It was found during program execution that using a value of $\beta = 1$ for the Newmark Beta integration algorithm (ref. 1) proved to be the most efficient with respect to computer processing time. The value of β is specified on a NASTRAN PARAM bulk data card. It should also be noted that the DIAG 10 option can be used in the executive control deck, which will result in the use of the alternate nonlinear load technique. This option may allow the user to use larger integration steps depending upon the particular problem.

NASTRAN RESULTS

Figures 3 and 4 show the theoretical thermal response plots of the NASTRAN thermal program using this modeling technique. These curves show the thermal response of the surface (Tsurface) exposed to the thermal flux in, at 25% ($T_{25\%}$), at 50% ($T_{50\%}$), at 75% ($T_{75\%}$) and at the backface ($T_{BACKFACE}$) of a 1-inch-thick steel plate. The flat portion at the top of the curves shows when ablation occurred at each point. The plots were generated using the NASTRAN X-Y plot capability.

These two thermal response plots were used to verify the NASTRAN program against the actual physical results of two test situations. The tests consisted of the impingement of the plume of a Tomahawk missile normal to a 2.54-centimeter (1-inch) thick steel plate for 9 seconds at two different relative positions. For the first test, the nozzle exit plane of the Tomahawk missile was located 5.182 meters (17 feet) from the steel plate. The results of this test showed 75% ablation occurred. This test corresponds to figure 3, which shows the same results.

In the second test, the nozzle exit plane of the Tomahawk missile was located 3.658 meters (12 feet) from the steel plate. The results of this test shows a burn-through (100% ablation). This test corresponds to figure 4, which shows the same results.

Figure 5 is included to show a thermal response comparison between running NASTRAN without this modeling technique and with the modeling technique (figure 4). Large differences between figures 4 and 5 are quite apparent.

CONCLUSIONS

The inclusion of nonlinear loads in the NASTRAN computer program allows the user a tremendous versatility in the solution of dynamic problems. When using these nonlinear loads in conjunction with NASTRAN, the user has a fast and comparatively easy tool in solving problems of thermal transient analysis with surface ablation.

It should also be noted that this type of modeling technique could be applied to other types of dynamic problems when using NASTRAN.

NASTRAN PROGRAM RECOMMENDATIONS

For the solution of thermal transient response problems, as well as for other types of dynamic response problems, the following NASTRAN program recommendations are made:

a. Acceleration-dependent nonlinear loads should be included in the NASTRAN NOLINi bulk data cards.

b. All combinations of the three different dependencies, displacement, velocity, and acceleration, should be included in the NASTRAN NOLIN2 nonlinear load bulk data card.

The implementation of these recommendations into NASTRAN in conjunction with the on/off switch and existing NOLINi nonlinear load dependencies, would allow the user to simulate disconnections (uncoupling of the equations) and/or alter a section or sections of a problem during program computations and continue on to the completion of computations for a complete transient or modal transient analysis.

REFERENCES

1. The NASTRAN Theoretical Manual, NASA SP-221(04), December 1977.

2. Sulyma, Peter, R. Predictions of the Plume Heating Thermal Structural Response of Launch Complex 39 to a Launch Induced Environment LMSC-HREC TR D497145, Lockheed Missiles and Space Company, Huntsville, Alabama, January, 1977.

3. The NASTRAN User's Manual (Level 17.0), NASA SP-222(04), Washington, D.C., December, 1977.

TABLE I. NASTRAN ANALOGIES

Variable	Heater Transfer Representation	Structural Mechanics Representation
u	Temperature	Displacement
$\dot{u}$	Rate of change of temperature (du/dt)	Velocity
P,N	Heat Flux	Forces, Moments
K	Thermal conductance	Stiffness
B	Thermal capacitance	Damping

FIGURE 1. NASTRAN THERMAL MODEL

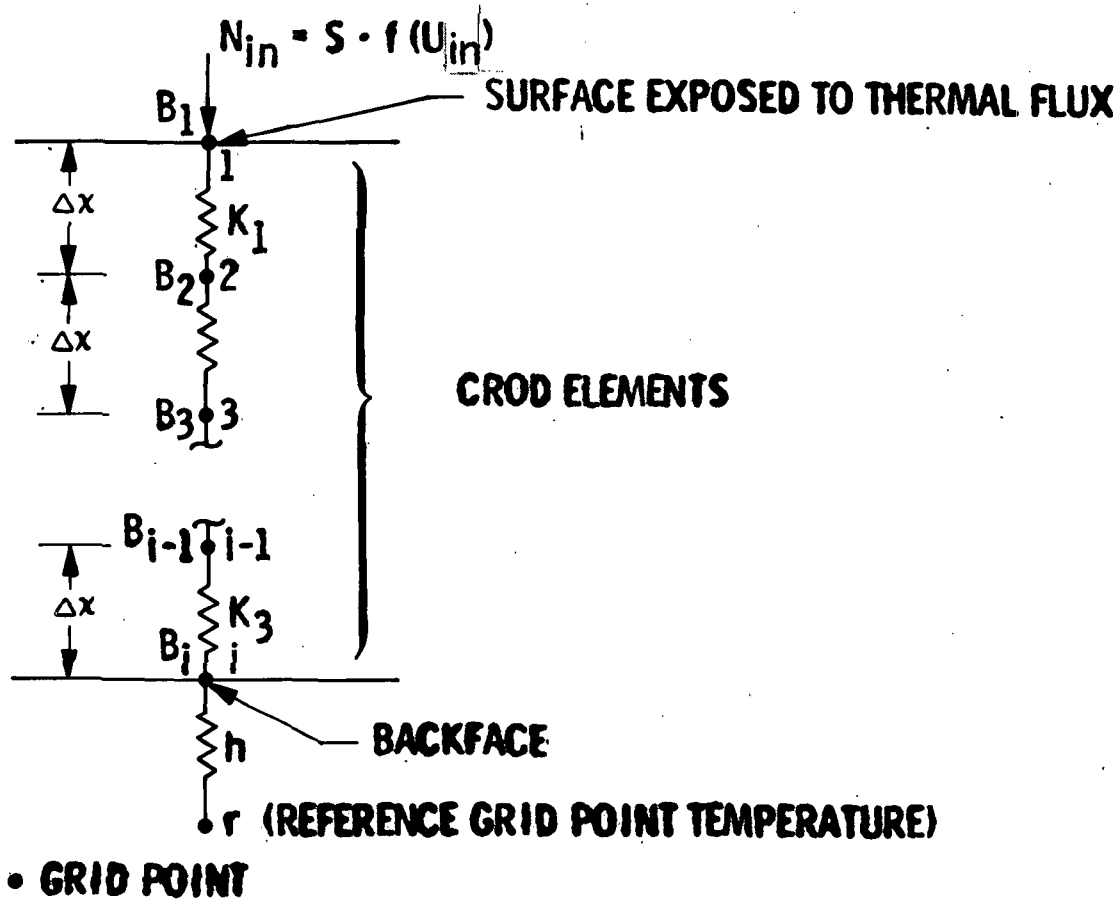


FIGURE 2. NASTRAN ABLATION EQUATIONS MODEL

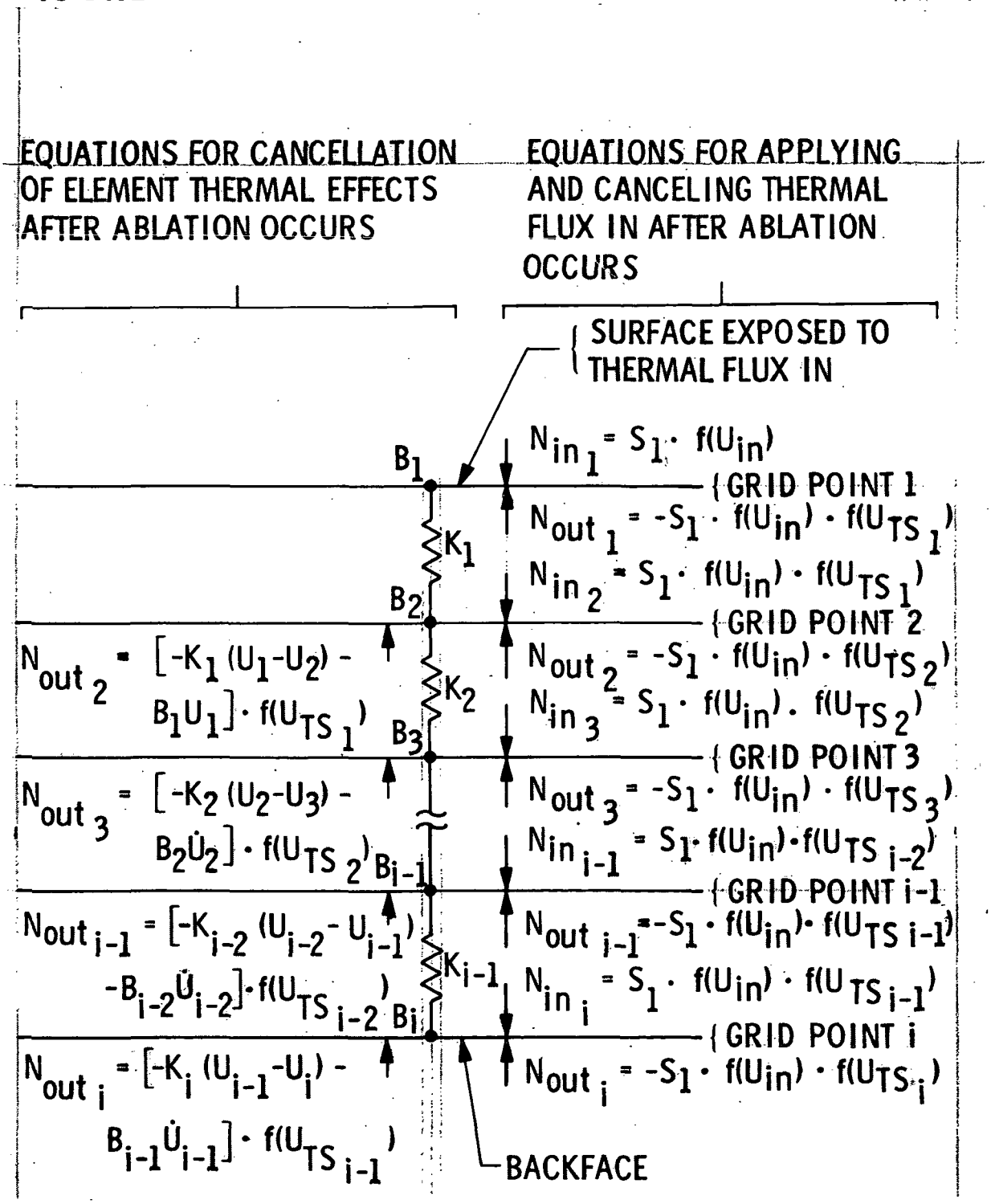


FIGURE 3. NASTRAN X-Y PLOT (TOMAHAWK TEST #1 SIMULATION) MODELING TECHNIQUE FOR ABLATION USED

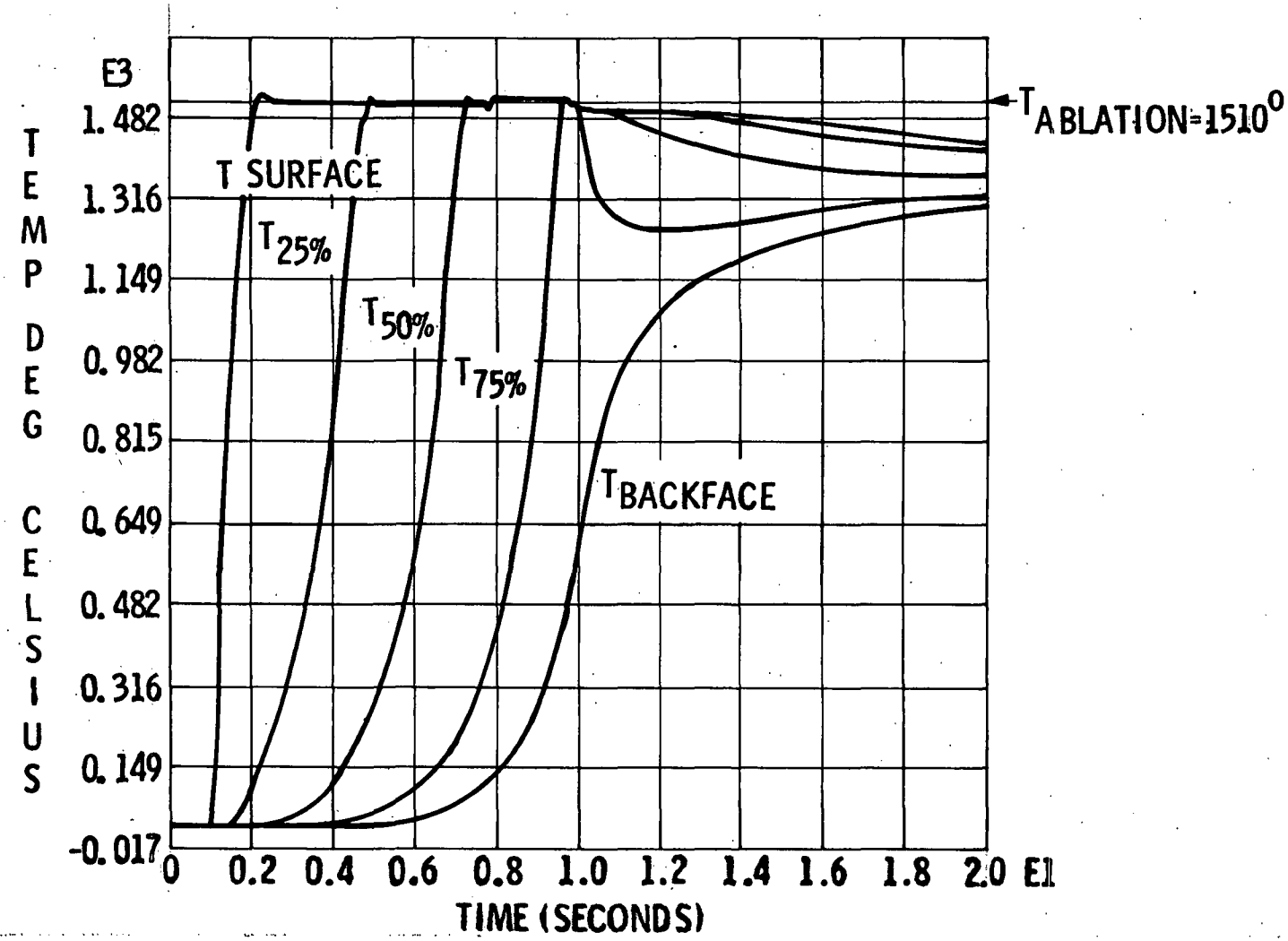


FIGURE 4. NASTRAN X-Y PLOT (TOMAHAWK TEST #2 SIMULATION) MODELING TECHNIQUE FOR ABLATION USED

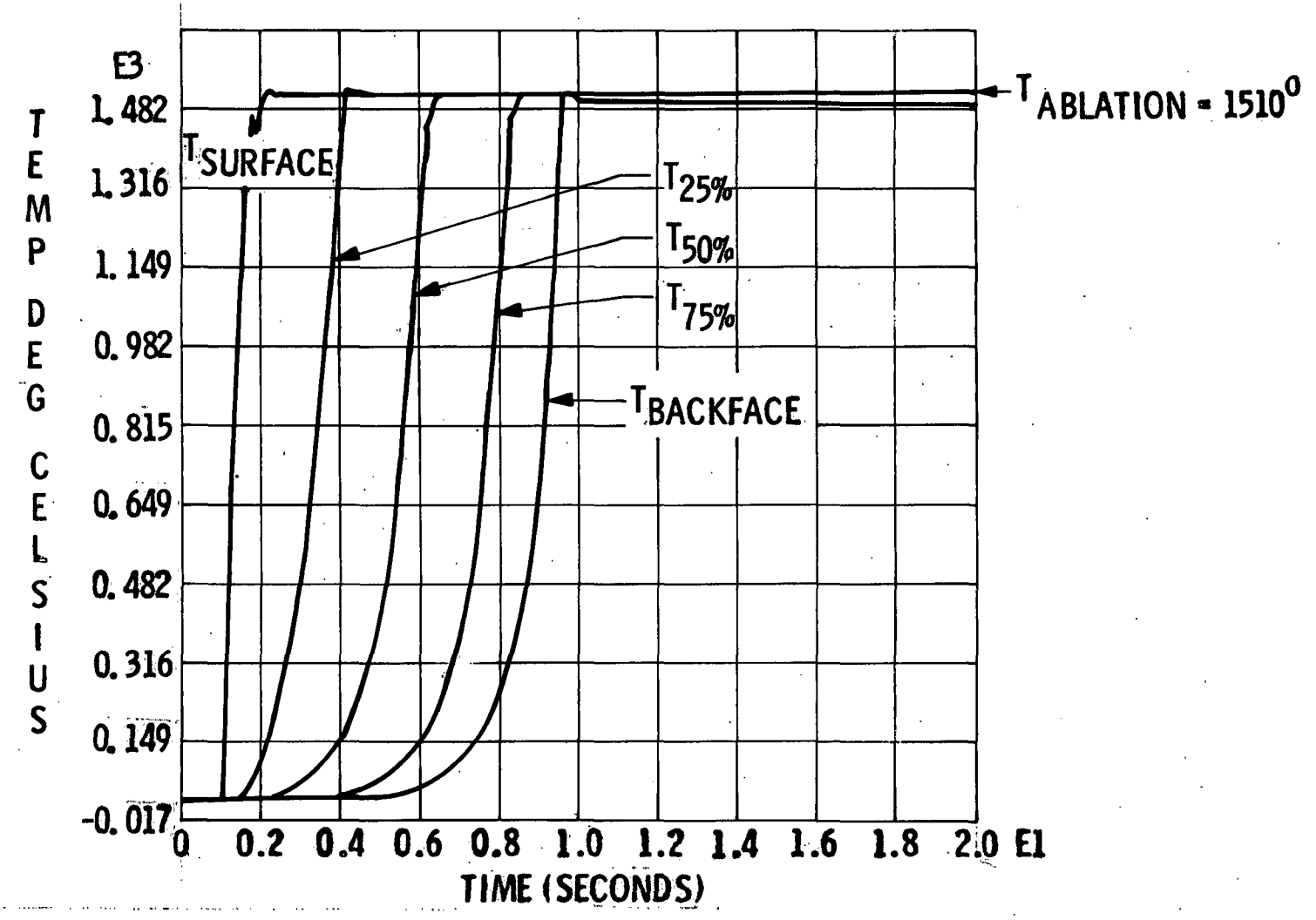
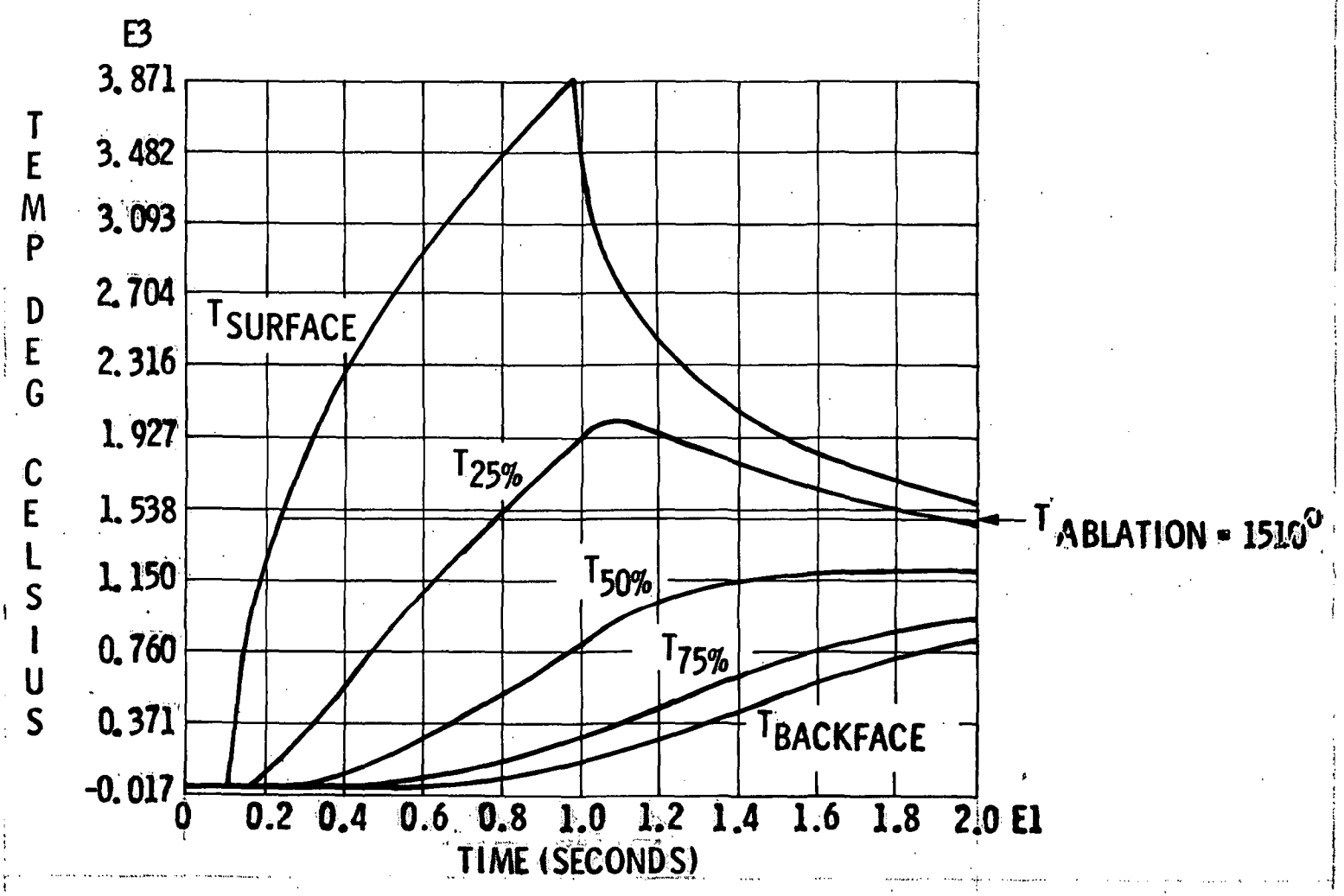


FIGURE 5. NASTRAN X-Y PLOT
MODELING TECHNIQUE FOR ABLATION NOT USED



ON THE APPLICATION OF NONLINEAR LOAD ELEMENTS TO THERMAL ANALYSES USING THE NASTRAN THERMAL ANALYZER

Hwa-Ping Lee
NASA/Goddard Space Flight Center

SUMMARY

Using the nonlinear load elements in thermal analysis to simulate an undocumented nonlinear thermal boundary condition is presented. The treatment of the nonlinearity arising from the temperature-dependent convective film coefficients is shown in detail. As an illustration, emphasis is placed on the modeling techniques and their interrelationships with the solution accuracy as affected by a specific integration algorithm of the transient thermal analysis used in the NASTRAN Thermal Analyzer. Briefly shown is the underlying theory on which the maneuvering of terms pertinent to the modeling depends. This demonstration provides some insight into the intricacies of the method that would be general to all applications. A recommendation is also made to modify a nonlinear load element that will enhance the solution capability and broaden the scope of application.

INTRODUCTION

The known solution capabilities of the NASTRAN Thermal Analyzer (NTA) have been well documented (ref. 1 and 2), and a large part of them have been demonstrated in detail (ref. 3). Despite their frequent encounterment in engineering applications, certain nonlinear solution capabilities have not been provided by the NTA in the transient-state thermal analysis (APP HEAT, SOL 9). Those excluded are, for instance, the temperature-dependent thermophysical properties, temperature-dependent convective film coefficients, etc. Their absence from the original NTA's capability list was solely attributed to the fact that the required computer processing time (CPU and I-O times) would be excessive. Computations involving the reassemblages of the thermal conductance matrix, together with the decompositions of the relevant matrices that contain the nonlinear thermophysical properties, would have to be repeated for all steps of the time increment in integration. The program shortcomings, however, may be compensated by an appropriate application of the available nonlinear load elements (NLLEs) in NASTRAN, although some elements were originally developed for structural analyses. The intrinsic modular structure of NASTRAN permits them to be accessible to the NTA. Those NLLEs, obviously, can function as nonlinear thermal loads, and they can be employed to simulate other complex physical phenomena as well. Otherwise, new modules would have to be developed and implemented, even though an excessive computer processing time was tolerable. Such an alternative would be an expensive and time-consuming

proposition. The user would be deprived of immediate solution meanwhile.

Six NLLEs are currently available in NASTRAN, namely NOLIN1*, NOLIN2, NOLIN3, NOLIN4, NOLIN5 and NFTUBE. The last two were developed specifically for thermal analyses: NOLIN5 is capable of simulating temperature sensitive thermal radiation surface properties, such as a louver (ref. 4), and NFTUBE is capable of dealing with thermal energy transfer between the wall and the fluid flowing inside a tube (ref. 5). The remainders are four nonlinear elements of general type capable of accommodating four different functions with a temperature or heat flux being the argument. NOLIN1 allows to specify a table through TABLEDi (i = 1,2,3,4), and it is the most versatile NLLE in NASTRAN.

The main purpose of this paper is to illustrate how to acquire a non-routine, but essential solution capability by an application of the NLLEs through modeling. For a systematic demonstration, NOLIN1 was employed to simulate the temperature-dependent convective film coefficients, h(T). The importance of this problem is evident in view of a variety of applications ranging from a domestic heating system, a hot gas flowing through a nozzle, a fluid flow in a nuclear reactor, and a flow in a wind tunnel during the start-up to a thermal ablation including a receding surface. They are all characterized by such a nonlinear boundary condition. More specifically, treatments of the temperature-dependent film coefficients based on one and two reference temperatures are given in detail. The dilemma arising from the presence of the MPC to establish an average temperature of two reference temperatures and the reference with that average temperature by the NLLEs was resolved by a trick modeling. A brief review of the theoretical aspect serves not only to provide a basis for the maneuvering in modeling and to provide insight into the intricacies of the method, but to amplify basis of modeling techniques that are equally valid to other applications. A recommendation for the improvement of the NOLIN2 is outlined. It would enhance solution capabilities and broaden the scope of engineering application. Guidelines for the user to apply the NLLEs in thermal analysis are provided.

THE DEMONSTRATIVE PROBLEM

For the convenience of easy reference and discussion, and later on, to demonstrate the modeling techniques, a relatively simple problem without losing any generality has been designed, and it is described below:

The transient temperature responses in a composite slab (pipe) of an infinite extent are to be determined. Initially, the slab has a uniform temperature of 293°K (20°C) throughout its body. The boundary surface of the metallic layer is in contact with a stream of hot gas maintained at 1422°K (1149°C) from time $t = 0$. The dimensions and thermophysical properties of the slab are tabulated in Table 1. The system depiction, together with its finite element model, is shown in Figure 1.

*The names of actual NTA cards are capitalized.

THE THEORETICAL CONSIDERATION

The success of attempting a solution utilizing any NLLEs to simulate a problem with nonlinear thermophysical properties relies upon the following factors:

(1) The nonlinear terms, arising from the temperature-dependent thermophysical properties, which are embraced in the square matrices of the thermal conductance and/or the thermal capacitance, must be extractable and separable from their normal positions. The resulting products of the temperature-dependent coefficients (kA/ℓ and hA or C) and their respectively associated variables (T or $\dot{T}$) must be transposed and merged into the nonlinear thermal load vector.

(2) An NLLE of an appropriate form suited to accommodate the transposed quantities must be selected. Any restraints associated with the use of the NLLEs must be strictly observed.

(3) A full acquaintance with the characteristics of each NLLE and the integration algorithm for the transient heat transfer analysis (APP HEAT, SOL 9) is essential in controlling and accomplishing desired solution accuracy, stability and efficiency at wish.

To serve as the theoretical basis for such a maneuvering, a brief review of pertinent mathematical operations is in order.

A transient thermal analysis in the NTA is based on the general heat equation in the matrix form as follows (ref. 1):

$$[C]\{\dot{T}\} + [K]\{T\} = \{Q^{\ell}\} + \{Q^{n}\} \qquad (1)$$

where

T = the vector of temperatures at grid points
$\dot{T}$ = the rate change of temperature vector
K = the thermal conductance matrix of constant elements
C = the thermal capacitance matrix of constant elements
Q^{ℓ} = the vector of applied linear thermal loads
Q^{n} = the vector of applied nonlinear thermal loads

The convective coupling terms are normally incorporated with the thermal conductive couplings in the thermal conductance matrix $[K]$ regardless of their dependency. When the convective film coefficient is temperature-dependent, the convective coupling terms embracing $h(T)$ have to be separated from their original $[K_o]$, and thus $[K_o]\{T\}$. The extracted terms being a new vector $\{H(T)\}$ are transposed to the other side of the equal sign in equation (1), and they become or merge with the vector of nonlinear thermal loads, $\{Q^{n}\}$. The rearranged new equation conforms exactly to equation (1), meaning that it is then within the solution capability of the NTA. Mathematically, it can be expressed by

$$[K_o(k,h(T))]\{T\} = [K_1(k)]\{T\} + \{H(T)\} \tag{2}$$

The integration algorithm for equation (1) is based on the modified Newmark Beta Method (ref. 6). It is a forward and backward differencing method with a parameter β that enables the user to select a value in the range of $0<\beta<1$. When equation (1) is transformed into such a difference expression, it reads (ref. 1):

$$[K]\{\beta T_{n+1} + (1-\beta)T_n\} + \frac{1}{\Delta t}[C]\{T_{n+1} - T_n\}$$

$$= \{\beta Q^{\ell}_{n+1} + (1-\beta)\{Q^{\ell}_n\}\} + (1+\beta)\{Q^n_n\} - \beta\{Q^n_{n-1}\} \tag{3}$$

the subscript n refers to the nth time step. The accuracy and stability of solution are closely related to β. Equation (3) can be rearranged to give

$$\left[\frac{1}{\Delta t}[C] + \beta[K]\right]\{T_{n+1}\} = \left[\frac{1}{\Delta t}[C] - (1-\beta)[K]\right]\{T_n\}$$

$$+\{\beta Q^{\ell}_{n+1} + (1-\beta)Q^{\ell}_n\} + (1+\beta)\{Q^n_n\} - \beta\{Q^n_{n-1}\} \tag{4}$$

The matrix $[1/\Delta t\ C + \beta\ K]$ is decomposed into its triangular factors and solved at each time step via the forward and backward substitution process. If [K] contains terms of k(T) or h(T), a search from the tables and replacement of new values based on $\{T_n\}$ have to be carried out for all terms residing inside those two square matrices that are associated with $\{T_{n+1}\}$ and $\{T_n\}$. Consequently, the matrix decomposition has to be performed for each time step. This requires a very excessive computer processing time. However, [K] needs be evaluated only once if its constituent terms are all constant. This situation is equally valid in [C], although C(T) is not considered in this study.

Relating the terms of nonlinear thermal load to the specific demonstrative problem, it is obtained from equations (1) and (2) that

$$\{Q^n\} = -\{H(T)\} = -\begin{Bmatrix} H_1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ H_{200} \end{Bmatrix} \tag{5}$$

where

$$\begin{cases} H_1 = h(T_r)AT_1 \\ H_{200} = -h(T_r)AT_{200} \end{cases} \tag{6}$$

and $h(T_r)$ denotes the dependency of the convective film coefficient in a reference temperature T_r which can be the wall temperature or an average temperature of the free stream and the wall temperature as will be discussed and treated in following sections. In the NTA, the NOLIN1 card defines an NLLE of the form

$$Q_i^n(t) = S_i F(T_j(t)) \tag{7}$$

where $Q_i^n(t)$ is the load being applied to the GRID i at time t, S_i is a scale factor. $F(T_j(t)$ denotes a tabulated function in T_j to accommodate $H_1(T_j)$ and $H_2(T_j)$ as defined in equation (6). The equivalent nonlinear thermal load functions are entered through a TABLEDi (i = 1,2,3,4) card. T (t) is any permissible time-varying temperature on which the functions depend. Other NLLEs, NOLIN2, NOLIN3 and NOLIN4, can be selected as long as their specific forms are suitable for respective applications.

THE CONVECTIVE FILM COEFFICIENT $h(T_r)$

The convective film coefficient h governs the magnitude of heat transfer between the hot gas and the boundary surface of structure. h, in general, can be expressed as $h(T_r)$ where the reference temperature T_r may refer to only one temperature, such as the wall temperature, or more than one temperature through an intermediary, such as the average value of the free stream temperature and the surface temperature of the wall. A typical example is a fully developed pipe flow. The expression for h is (ref. 7)

$$h = 0.023 \frac{k_\infty}{D_h} (Re)^{0.8} (Pr)^{0.4} \tag{8}$$

For large temperature differences, a mean temperature is used in evaluating all physical properties except the flow velocity. This procedure would involve very tedious interrations. To simplify computations, it is desirable to separate the temperature-dependent from the independent factors. An alternative form is, therefore, obtained by an application of the equation of state and the continuity equation, which leads to

$$h = 0.023\, C_{p\infty} \mu_\infty^{0.2} Pr_\infty^{-0.6} G_\infty^{0.8} D_h^{-0.2} \left(\frac{T_W}{T_\infty}\right)^{0.8} \tag{9}$$

As the free stream temperature T_∞ is known, all quantities in the preceding equation except T_w are constants. h is, therefore, a function of T_w. For the problem of concern, it is $h = h(T_1)$. The values of this function obtained by using equation (9) are listed in Table 2. Similar expression, but referencing with an average value of the free stream temperature and the surface temperature is also widely used in engineering practices. For this demonstrative problem, it is $h = h(T_r)$ and $T_r = \frac{1}{2}(T_1 + T_{200})$. The values of h vs T_r are given in Table 3.

THE MODELING TECHNIQUES AND DISCUSSION

A one-dimensional finite element model is sufficient to represent the demonstrative problem, and this model is shown in Figure 1. A listing of the input data deck is reproduced in Figure 2. With comments appropriately inserted, functions regarding various segments of the deck are self-explanatory. The space bounded by two rows of asterisks is reserved for adding a packet of cards which simulate the effect of the convective film coefficients. Three cases of h are considered: (1) h of a constant value, (2) $h = h(T_1)$, and (3) $h = h(T_r)$ with $T_r = \frac{1}{2}(T_1 + T_{200})$.

h of a constant Value

To assure correctness of modeling and to evaluate the obtainable accuracy of solution by the nonlinear load approach, the case of a constant film coefficient with $h = 0.467$ W/cm^2-K was tested first. Because, this case is well within the documented capability of the NTA. It can be accomplished using a train of cards CHBDY, PHBDY and MAT4 by the conventional modeling method. The temperature results so obtained serve as a base to compare with that obtained by the nonlinear load approach. The two packets of input data cards, whose images are shown in Figures 3(a) and 3(b), were used in respective computer runs. The equivalent nonlinear thermal loads for the pair of NLLEs through NOLIN1 405 with TABLED1 4051 and TABLED1 4052 are given in Table 4. The effect of selecting the time-step size Δt for integration on solution accuracy was also studied.

Temperature results summarized in Table 5 with the headings of Cases 1 and 3 are solutions yielded by these two approaches. In both cases, the same time-step size (the Δt-set (a) in Figure 4) was used. The largest difference between the two temperature profiles occurs at $t = 0.25$ sec., suggesting that a refinement of the time-step size is warranted. To maximize the efficiency of the computer processing time, refinements of Δt were made only in the beginning segments of time. Three sets of time-step size, which are presented in the NTA TSTEP card format and identified as the Δt-sets (a), (b) and (c), are shown in Figure 4. A total of five cases were computed, two (Cases 1 and 2) by the conventional modeling method and the other three (Cases 3, 10 and 11) by the nonlinear load approach. Temperature results are tabulated in Table 5 for comparison. It is evident that both modeling methods can yield comparable temperature solutions with a desired accuracy, provided an appropriate Δt-set is used (e.g., to compare Case 2 with Case 11).

This test gave confidence in the nonlinear load approach and verified the adequacy of the two sets of time-step size, namely the Δt-sets (b) and (c). They were, therefore, selected in all following computations. Inevitably, efficiencies of those runs using the Δt-sets (b) and (C) were penalized by a certain amount. They are reflected by the CPU and I-O times, which are also summarized in Table 5.

$h = h(T_1)$

The modeling procedure similar to that used in the preceding case of a constant h is equally applicable to this case of a convective film coefficient h, which is dependent on the surface temperature of the wall. The preparation of the equivalent nonlinear thermal loads, however, is much more involved. Computations of those equivalent loads were based on the expressions given in equation (6). The computational steps and results are detailed in Table 3. The relevant packet of input data cards is shown in Figure 3(c).

Two computer runs were executed with the specified Δt-sets (b) and (c). Table 6 displays their temperature results under the headings of Cases 12 and 13. Results are in excellent agreement, meaning that either Δt-set is satisfactory to render accurate solutions. The Δt-set (b) was then selected in studying the effect of the parameter β in the integration algorithm on temperature solutions.

The integration algorithm available in the NTA for its transient thermal analysis is a modified Newmark Beta Method. It uses a fixed time-step size, but permits different time-step sizes to be changed at discrete times upon the user's specification. To avoid excessive matrix decompositions, the number of changes should be kept low. This algorithm is known to be an efficient one for large systems of linear equations, but it is uncertain for problems with strong nonlinearities. As accuracy and stability of a solution are closely related to the parameter β, a numerical investigation with different values of β was conducted. The values of β, including 0, 0.25, 1 and 0.55 (the default value), were executed. Except the case of using $\beta = 0.5$ (the central differences), whose temperature results are virtually identical to that of Case 13 (using $\beta = 0.55$) and, therefore, eliminated from the tabulation in Table 6. Case 16 (using $\beta = 1.0$, the backward differences), yields comparable temperature results as that of Case 13. However, the solution instabilities caused both cases with $\beta = 0.25$ and 0 (Euler integration) to terminate in their mid-courses. They are respectively labeled as Cases 17 and 18 and also included in Table 6.

$h = h(T_r)$ with $T_r = \frac{1}{2}(T_1 + T_{200})$

When the average temperatures T_r of the free stream temperature T_{200} and the surface temperatures of wall T_1 are referenced by the temperature-dependent convective film coefficients $h = h(T_r)$, the MPC (Multipoint constraint) card is suited to provide such a relation. When $T_{100} = T_r$ is assigned, the equivalent nonlinear thermal loads are functions of T_{100}, meaning that $T_{100}(t)$ will be referenced by the nonlinear thermal load tables, if the modeling technique similar to that used in the two preceding cases is followed. Since T_{100} is constrained by an MPC, this modeling has violated one cardinal rule that must be strictly observed by a user in applying the nonlinear loads. The grid points to which the nonlinear loads are applied and the variables (either temperature or heat flux) on which they depend must not be variables eliminated by constraints, i.e., they must be in the solution set. To circumvent this dilemma, a fictitious point GRID 150 was created for T_{150}, and it was connected to T_{100} by a super conductor, so that $T_{100}(t) = T_{150}(t)$. T_{150} instead of T_{100} was then referenced by $h(T_r)$, i.e., $h(T_r) = h(T_{100}) = h(T_{150})$, which, in turn, produced the equivalent nonlinear thermal loads. Figure 3(d) shows all relevant cards

and their relationships in detail, and the entered quantities in the nonlinear load tables are easily identifiable in comparison with those values given in Table 4.

The two t-sets (b) and (c) were again selected to execute the computer runs. Results with the labeling of Cases 14 and 15 are shown in Table 7. Temperatures are still overall in good agreement with each other, but they no longer show the same quality as evidenced in previous cases. A scrutiny of the results reveals that relatively large temperature differences exist between these two temperature profiles in the range of time from 1.0 to 3.0 sec. This is a typical example showing the intricacies of this method, the interplays of the selected time-step sizes with the integration algorithm, and the effect on temperature results due to the presence of the MPC.

Equation (4) shows that an inherent time lag error exists in applying the nonlinear thermal loads. They are computed for T of the (n+1)th time step based on temperature values at the nth and (n-1)th time steps. The larger the time-step size, the greater the deviation of the nonlinear thermal load from the desired value. When MPC is employed to relate a floating temperature (e.g., T_1) to a reference temperature (e.g., T_{150}) on which values of the nonlinear thermal loads depend indirectly, it introduces an additional step of time lag, and thus, an aggregated error for the nonlinear thermal load value. The errors are now induced by the time lag steps spanning from the (n+1)th to (n-2)th steps. Consequently, the selection of a finer time step size is appropriate in applying the nonlinear thermal loads, especially when an MPC is present to average two temperatures.

OTHER APPLICATIONS AND RECOMMENDATIONS

The modeling techniques of using the NLLEs to simulate the temperature-dependent convective film coefficients have been methodically illustrated. The demonstrated equivalent nonlinear thermal loads are all functions of temperature. If a simulation calls for the nonlinear thermal load to be a function of heat flux, a combination of the EPOINT and TF cards, in addition to the NOLIN1s and TABLED1s, have to be used. In all cases, values for the equivalent nonlinear thermal loads must be prepared by the user manually. This is an error-prone and elaborate process inherent in using the NLLEs. This deficiency, however, can be removed with only a relatively simple change made to the NOLIN1. It will further broaden its capability greatly. The recommendation is to modify the computer code pertinent to S, the fifth field, in the NOLIN1 card so that it will accept any time-varying temperature, whether specified or computed values at some grid points in the model. The expanded capabilities will include, for instance, the same demonstrative problem with the hot-gas temperatures being a specified time-dependent temperature function.

CONCLUSIONS

The nonlinear load elements have been illustrated to simulate nonlinear boundary conditions, in particular the temperature-dependent convective film coefficients in transient thermal analyses. The following remarks also serve as guidelines for those who attempt to use the NLLEs:

(1) Prior to applying the nonlinear thermal loads to an NTA model in a transient thermal problem, a trial run on a simplified model is advisable. Only the linear thermal loads equal in magnitude to that of the nonlinear load set are necessary to apply to this model, and a relatively coarse time-step size may be used in the execution. The profile of the temperature results will provide a good indication on which the final selection of appropriate time-step sizes in runs with the real NLLEs depends.

(2) Taking into consideration of the time lag errors, very small time-step sizes are always desirable. However, it is not necessary to select the same fine size uniformly over the entire range of time for integration. To maximize the processing efficiency, the selection of nonuniform time-step sizes according to the results of the trial run is a sensible and practical approach. Very fine time-step sizes should be reserved to concentrate at which the temperature profile has the largest slope in its rate change of temperature curve.

(3) When MPC (also TF, if used) is present in the model for establishing a reference temperature that is referenced by the NLLEs, such as NOLIN1, a fictitious super thermal conductor has to be used to connect to that original grid point of referencing. A new grid point at the other end of the super thermal conductor is then to be referenced by any NLLEs so as not to violate the cardinal rule in applying the nonlinear loads.

(4) For solution stability, which also affects accuracy, the proper selection of a value for the parameter β of the integration algorithm depends on the degree of nonlinearity in presence. However, only the range of $0.5 < \beta < 1$ should be considered.

REFERENCES

1. Lee. H.P.: NASTRAN Thermal Analyzer - Theory and Application Including a Guide to Modeling Engineering Problems, Vol. 1. NASA/Goddard Space Flight Center, X-322-76-16, Dec. 1975; Also NASA TMX-3505, April 1977.

2. The NASTRAN User's Manual (Level 17.0), NASA SP-222(04), December 1977, Washington, D.C.

3. Jackson, C.E., Jr.: NASTRAN Thermal Analyzer - Theory and Application Including a Guide to Modeling Engineering Problems, Vol. 2: A Guide to the GSFC NASTRAN Thermal Analyzer Sample Problem Library, NASA/Goddard Space

Flight Center, X-322-76-17, December 1975.

4. Lee, H.P., and Harder, R.L.: The GSFC NASTRAN Thermal Analyzer New Capabilities. The Fifth NASTRAN User's Colloquium, NASA TMX-3428, October 1976, pp. 119-126.

5. Lee, H.P.: On the Thermo-Fluid Elements and Their Applications. The Sixth NASTRAN User's Colloquium, NASA Conference Publication 2018, October 1977, pp. 25; Also NASA/GSFC Technical Report, October 1977.

6. Newmark, N.M.: A Method of Computation for Structural Dynamics, Proceeding ASCE, Journal of the Engineering Mechanics Div., EM-3, July 1959, pp. 67-94.

7. McAdams, W.H.: Heat Transmission, 3rd Edition, McGraw-Hill Book Company, 1954.

TABLE 1

DIMENSIONS AND THERMOPHYSICAL
PROPERTIES OF THE COMPOSITE SLAB (PIPE)

Material	Stainless Steel (a)	Insulation-Phenolic (b)
Thermal Conductivity k (W/cm-k)	0.16258	0.00502
Thermal Capacitance ρCp (J/cm^3-k)	4.029	1.725
Thickness ΔX (cm)	0.30	0.40
Cross-section A (cm^2)	1.0	1.0

TABLE 2

THE EQUIVALENT NONLINEAR LOADS FOR
THE CASE OF h = 0.467 W/cm^2-k

T_{200}	$N_{200} = hAT_{200}$	T_1	$-N_1 = hAT_1$
1422.0	664.074	293	136.831
		589	275.063
		867	404.889
		1144	534.248
		1422	664.074

TABLE 3

THE EQUIVALENT NONLINEAR LOADS
FOR THE CASE OF h = $h(T_1)$

T_1	$h(T_1)$	$-N_1(T_1) = hAT_1$	T_{200}	$N_{200}(T_1) = hAT_{200}$
293	0.132	38.676	1422	187.704
589	0.231	136.059		328.482
867	0.314	272.238		446.508
1144	0.393	449.592		558.846
1422	0.467	664.074		664.074

TABLE 4

THE EQUIVALENT NONLINEAR LOADS
FOR THE CASE OF $h = h(T_{ave})$

T_{200}	T_1	T_{ave}	$h(T_{ave})$	$N_{200}(T_{ave}) = hAT_{200}$	$-N_1(T_{ave}) = hAT_1$
1422	293	857.5	0.132	187.704	113.190
↓	589	1005.5	0.231	328.482	136.059
↓	867	1144.5	0.314	446.508	272.238
↓	1144	1283.0	0.393	558.846	449.592
↓	1422	1422.0	0.467	664.074	664.074

TABLE 5

A COMPARISON OF TEMPERATURE RESULTS
FOR THE CASES WITH A CONSTANT h

Time (sec)	Case No.				
	1	2	3	10	11
0	293.00	293.00	293.00	293.00	293.00
0.25	575.82	580.41	578.73	582.08	581.00
0.50	665.64	668.29	667.90	669.82	668.75
0.75	723.79	725.67	725.68	727.06	726.06
1.00	768.38	769.88	770.07	771.19	770.23
1.25	806.14	807.42	808.22	809.19	807.75
1.50	839.92	841.08	842.04	842.93	841.40
1.75	871.04	872.11	873.17	873.99	872.42
2.00	900.10	901.10	902.23	903.00	901.41
3.00	1000.39	1001.21	1003.56	1004.18	1002.48
4.00	1080.22	1080.90	1083.37	1083.86	1082.23
5.00	1143.93	1144.50	1146.93	1147.32	1145.81
6.00	1194.83	1195.29	1197.61	1197.92	1196.53
7.00	1235.52	1235.90	1238.05	1238.29	1237.04
8.00	1268.08	1268.39	1270.35	1270.55	1269.43
Δt-set	(A)	(C)	(A)	(B)	(C)
CPU/I-O (min)	0.125/1.170	0.189/1.133	0.154/1.309	0.177/1.368	0.198/1.256

TABLE 6

A COMPARISON OF TEMPERATURE RESULTS
FOR THE CASES WITH $h = h(T_1)$

Time (sec)	Case No.				
	12	13	16	17	18
0	293.00	293.00	293.00	↓	↓
0.25	401.14	401.27	401.06		
0.50	445.89	446.07	445.82		
0.75	478.54	478.76	478.61		
1.00	505.58	505.83	505.82		
1.25	529.69	530.07	530.17		
1.50	552.17	552.62	552.80		
1.75	573.65	574.15	574.40		
2.00	594.44	594.99	595.28		
3.00	673.30	673.96	674.38		
4.00	746.00	746.73	747.09		
5.00	813.37	814.15	814.46		
6.00	875.67	876.52	876.80		
7.00	929.48	930.38	930.66		
8.00	977.03	977.96	978.18		
Δt-sec	(C)	(B)	(B)	(B)	(B)
CPU/I-O (min)	0.224/1.487	0.200/1.606	0.192/1.615	-	-
Parameter β	0.55	0.55	1.0	0.25	0

TABLE 7

A COMPARISON OF TEMPERATURE RESULTS
FOR THE CASES WITH $h = h(T_{ave})$

Time (sec)	Case No.	
	14	15
0	293.00	293.00
0.25	350.94	351.29
0.50	387.00	387.63
0.75	420.17	420.99
1.00	453.04	454.05
1.25	486.41	488.16
1.50	521.83	524.19
1.75	559.79	562.77
2.00	600.52	603.20
3.00	724.14	725.32
4.00	832.04	833.01
5.00	928.13	928.48
6.00	1006.74	1006.81
7.00	1075.18	1075.07
8.00	1135.05	1134.79
Δt-sec	(B)	(C)
CPU/I-O (min)	0.192/1.661	0.235/1.490

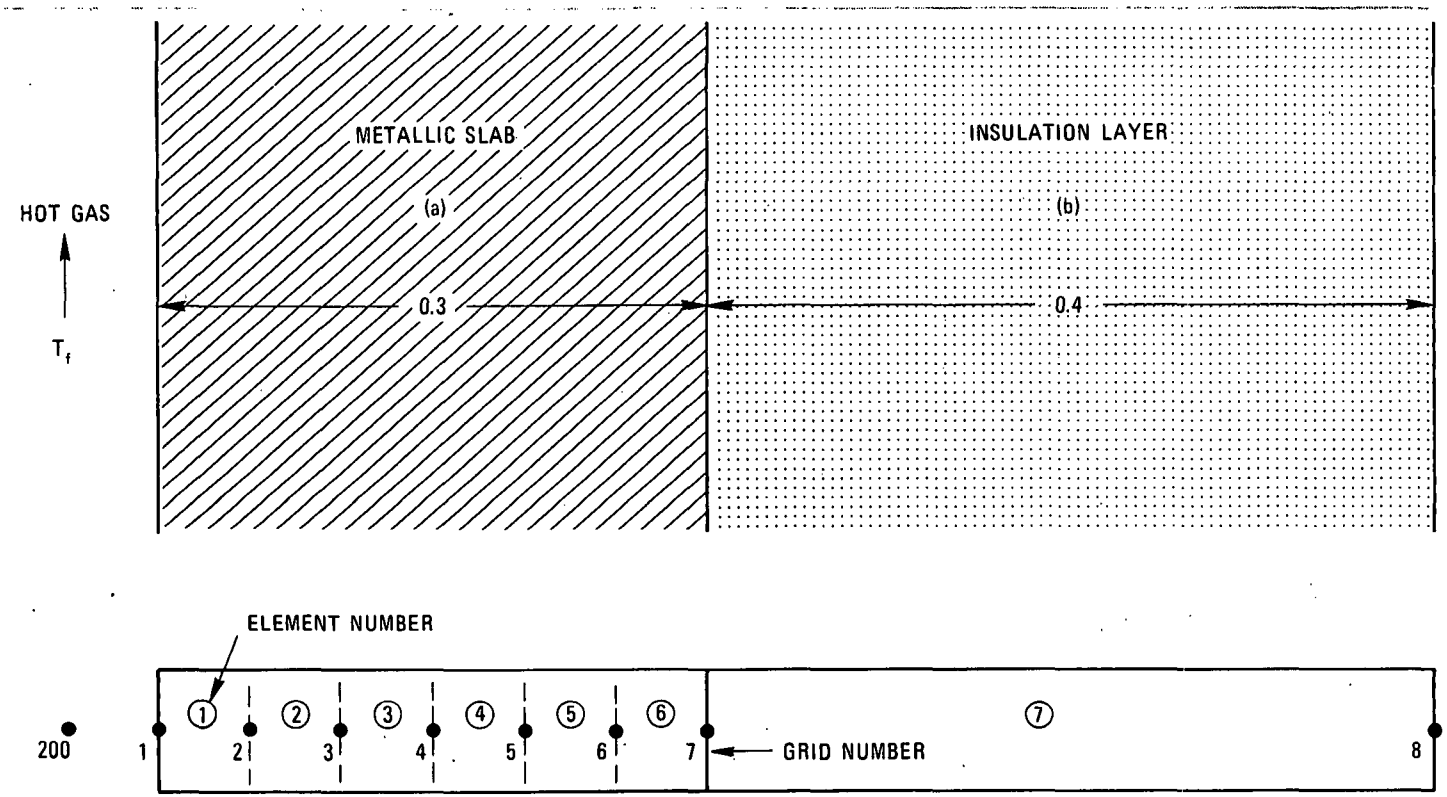


Figure 1 - The Composite Slab (Pipe) and its Finite Element Model

```
$  EXECUTIVE CONTROL DECK
ID H P LEE, NASA-GSFC, SPACE TECHNOLOGY DIVISION
APP HEAT
SOL 59
CEND
$ CASE CONTROL DECK
ECHO=BOTH
LINES=75
TITLE= DEMONSTRATION OF TEMPERATURE-DEPENDENT CONVECTIVE FILM COEFFICIENT
SUBTITLE= WITH A CONSTANT HOT GAS TEMPERATURE AT GRID 200
MPC=305
DLOAD=401
NONLINEAR=405
TSTEP=601
IC=701
TEMP(MATERIAL)=705
OUTPUT
SET 20=1,2,200
ULOAD=20
ELFORCE=20
NLLOAD=20
SET 40= 1,4,7,8,100,150,200
THERMAL= 40
BEGIN BULK
$  BULK DATA DECK
$  GRID POINTS
GRID     1                0.00
GRID     2                0.05
GRID     3                0.10
GRID     4                0.15
GRID     5                0.20
GRID     6                0.25
GRID     7                0.30
GRID     8                0.70
GRID     200              -1.0
$  CONDUCTION ELEMENTS
CROD     1        101     1        2        2        101      2        3
CROD     3        101     3        4        4        101      4        5
CROD     5        101     5        6        6        101      6        7
PROD     101      102     1.0
MAT4     102      0.16258 4.029
CROD     7        103     7        8
PROD     103      104     1.0
MAT4     104      0.00502 1.725
$  INITIAL CONDITIONS
TEMP     701      200     1422.0
TEMPD    701      293.0
$  ESTIMATED TEMPERATURES
TEMPD    705      1422.0
$  SPECIFIED CONSTANT TEMPERATURE FOR THE HOT GAS AT GRID 200
CELAS2   301      1.E6    200      1
DAREA    501      200     1        1422.E6
TLOAD2   401      501     0                 0.0      1.E5     0.0      0.0      +LD1
+LD1     0.0      0.0
$  TIME INCREMENTS AND INTEGRATION STEPS
TSTEP    601      25      0.005    10                                          +TS1
+TS1              35      0.025    5                                           +TS2
+TS2              20      0.05     5                                           +TS3
+TS3              120     0.10     10
$ ********************************************************************************


$ ********************************************************************************
ENDDATA
/*
```

Figure 2 – A Listing of the Input Data Deck of the Demonstrative Problem

```
$  CONVECTIVE BOUNDARY IN CONTACT WITH HOT GAS
CHBDY    1001     1002     POINT    1                                        +CY
+CY      200
PHBDY    1002     1003     1.0
MAT4     1003     0.467
```

(a) A constant h by the conventional modeling technique

```
NOLIN1   405      1        1        -1.0     1        1        4051
TABLED1  4051                                                              +TB1
+TB1     293.0    136.831  589.0    275.063  867.0    404.889  1144.0   534.248 +TB2
+TB2     1422.0   664.074  ENDT
NOLIN1   405      1        1        664.074  1        1        4052
TABLED1  4052                                                              +TB52
+TB52    293.0    1.0      1450.0   1.0      ENDT
```

(b) A constant h by the equivalent nonlinear thermal load approach

```
NOLIN1   405      1        1        1.0      150      1        4054
TABLED1  4054                                                              +TB7
+TB7     293.0    187.704  589.0    328.482  867.0    446.508  1144.0   558.846 +TB8
+TB8     1422.0   664.074  ENDT
NOLIN1   405      1        1        -1.0     150      1        4055
TABLED1  4055                                                              +TB55
+TB55    293.0    38.676   589.0    136.059  867.0    272.238  1144.0   449.592 +TB56
+TB56    1422.0   664.074  ENDT
```

(c) $h = h(T_1)$ by the equivalent nonlinear thermal load approach

```
$  USING MPC TO CREATE AN AVERAGE TEMPERATURE AT GRID 100 FOR GRIDS 1 AND 200
MPC      305      100      1        1.0      1        1        -0.5       +MC
+MC               200      1        -0.5
$  CREATING GRID 150 TO AVOID GRID 100 BEING DIRECTLY REFERENCED BY NOLIN1
GRID     100               -0.5     0.0      0.0
GRID     150               -0.5     1.0      0.0
CROD     8        105      100      150
PROD     105      106      1.0
MAT4     106      5.0      0.1
TEMP     701      100      857.5    150      857.5
$ ********************************************************************
$  SIMULATING TEMPERATURE-DEPENDENT FILM COEFFICIENTS BASED ON T(150)
NOLIN1   405      1        1        1.00     150      1        4061
TABLED1  4061                                                              +TB1
+TB1     857.0    187.704  1005.5   328.482  1144.5   446.508  1283.0   558.846 +TB2
+TB2     1422.0   664.074  ENDT
NOLIN1   405      1        1        -1.0     150      1        4062
TABLED1  4062                                                              +TB3
+TB3     857.0    113.19   1005.5   136.059  1144.5   272.238  1283.0   449.592 +TB4
+TB4     1422.0   667.074  ENDT
```

(d) $h = h(T_r)$ by the equivalent nonlinear thermal load approach

Figure 3 - The Images of Packets of Data Cards for Simulating Various Convective Film Coefficients

```
$  TIME INCREMENTS AND INTEGRATION STEPS
TSTEP   601      40       0.025    10                    +TS3
+TS3             20       0.05     5                     +TS4
+TS4             60       0.10     10
```

Δt-set (a)

```
$  TIME INCREMENTS AND INTEGRATION STEPS
TSTEP   601      25       0.005    10                    +TS1
+TS1             35       0.025    5                     +TS2
+TS2             20       0.05     5                     +TS3
+TS3             120      0.10     10
```

Δt-set (b)

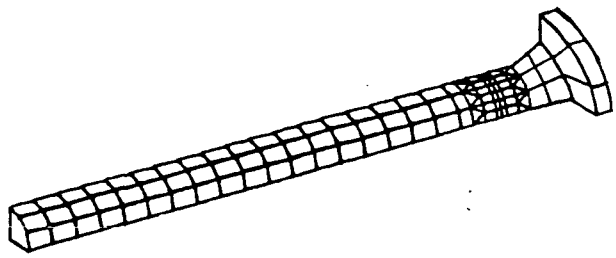
```
$  TIME INCREMENTS IN INTEGRATIONS AND PRINT-OUT
TSTEP   601      400      0.005    50                    +T1
+T1              120      0.05     20
```

Δt-set (c)

Figure 4 - Three Sets of Time-Step Size Used in Computer Executions

NASTRAN ANALYSIS OF HEAT-TRANSFER FLUID FILL PIPE FAILURES

J. Ronald Winter
Tennessee Eastman Company



SUMMARY

Failure analysis can be a very demanding endeavor that requires a great deal of interdisciplinary assistance if the correct solution to a particular problem is to be found. This paper presents an example that shows the difficulties one can encounter in such analyses and the advantage of the finite element method (NASTRAN) in assisting one in determining the true cause of a failure. In this example, cracks were developing along a pipe weld. After discarding several possible causes for the failures, it was finally determined that the problem was due to stress-corrosion-cracking associated with a rather unusual and novel environmental condition.

INTRODUCTION

In performing failure analysis an engineer often investigates numerous possible causes of a failure before he finally determines the real reason or reasons for the particular failure. This is usually due to either the lack of sufficient information or incorrect information. In many cases he must also either prove or disprove the reasons for the failure put forth by others. This situation is especially true in domestic industry and to a somewhat lesser extent in non-domestic establishments. In this paper, all of these important facts are presented rather than just the final solution. This should help new engineers in developing the logic for attacking such problems, and to be aware of the pitfalls and other oddities with which one must contend in order to obtain a successful solution. Hopefully, it will also make new engineers more aware of the need for assistance from other fields of engineering. Such interdisciplinary assistance is often a necessity for solving certain problems.

BACKGROUND

At the time this particular problem was presented to the author, five previous failures had been encountered. The failures consisted of cracks in the fill pipe (stand pipe) of a closed (noncirculating) Dowtherm* (heat-transfer

*Trademark of Dow Chemical Company.

fluid) system used to maintain the temperature of several spin blocks in a polyester spinning plant. The basic layout of the assembly which is relevant to this presentation is shown in Figure 1. The fill pipe and the location of the failure are shown in Figure 2. Detailed dimensions of the fill pipe are shown in Figure 3. The initial five failures consisted of cracks that mimicked bending fatigue failures. This had led several persons to believe that the failures were due to vibration induced by surrounding equipment or from some large vibrating dryers at another location in the building. These particular dryers had previously produced severe structural oscillations during transient conditions (start-up, shutdown, and coast down). This was the basic situation that existed when the author started the investigation.

The need for an accurate solution to this problem increased dramatically as the number and frequency of failures increased due to the associated increase in lost production and the fire hazard associated with heat-transfer fluid leaks. Several minor fires were encountered before the problem was corrected.

INITIAL ANALYSIS

Although the failures did have the appearance of bending fatigue, I doubted this explanation because of the relatively large stiffness of the fill pipe assembly and the fact that any large oscillations that had been encountered were in the Z direction (axial to the pipe) rather than in the X or Y (lateral) directions necessary to cause bending fatigue failures. But this remained to be proven.

Since I was working on numerous other jobs at this time, the easiest and fastest way to perform the required vibration analysis was undertaken. This involved using a very simple NASTRAN finite element model* of the fill pipe. Using this method allowed me to easily account for the gusset stiffness and the tapered section of the flange. First, a normal mode (real eigenvalue) analysis was performed. This showed that the first natural frequency of this system was approximately 65 cps which, as expected, was far beyond the operating frequency of any of the vibrating equipment or electrical motors. The same model was then used to perform a forced response analysis using the transient oscillations associated with the vibrating dryers as the forcing function. See Figure 4 for a typical response plot. This further indicated that vibration was not the problem. A frequency response analysis was also performed via NASTRAN which indicated the same results. With this data in hand, it was then concluded that the failures were not associated with vibration. This fact was forwarded to the production and maintenance personnel along with a request to save the next failure for careful inspection of the failure surfaces. All previous failures had been repaired by either cutting out the cracked region and rewelding or by installing a new section of pipe.

*The model consisted of BAR elements for the pipe, plate elements for the gussets and a CØNM element for the heavy flange. See Figure 3.

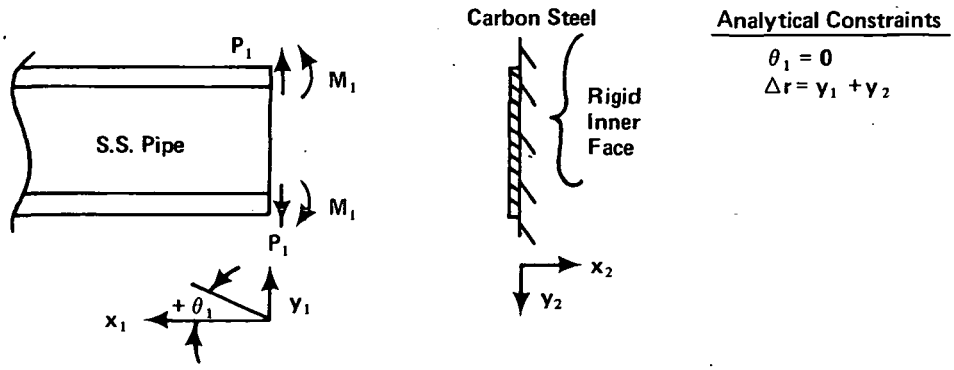
Within a few weeks, two additional failures were encountered. One occurred on a new assembly and one was a repeat failure. The repeat failure had been in service about three months. Most of the previous failures had been encountered in three to six month intervals. But there were several assemblies that had not failed even though they had been in operation for over a year. Thus, it became evident that a considerable amount of information was missing.

As requested, the last two failures were held for me. They were taken to the materials laboratory where a micro-hardness test was performed. The micro-hardness test results indicated a substantial decrease in ductility in the weld region near the cracks. Due to an unusually heavy materials lab work load, no further metallurgical examinations were performed at this time.

From the preparation of the initial NASTRAN model, I had become quite aware of the different materials of construction; i.e., stainless steel and carbon steel. The difference in the coefficients of thermal expansion for these materials would result in a differential thermal dilation (radial expansion) in the weld region. Such a situation results in the development of discontinuity stresses. Such discontinuity stress might at least partially explain the large hardness variations in the axial direction at the weld. Of course, some hardness variations were expected due to the residual stresses associated with welding. However, the variations were greater than anticipated.

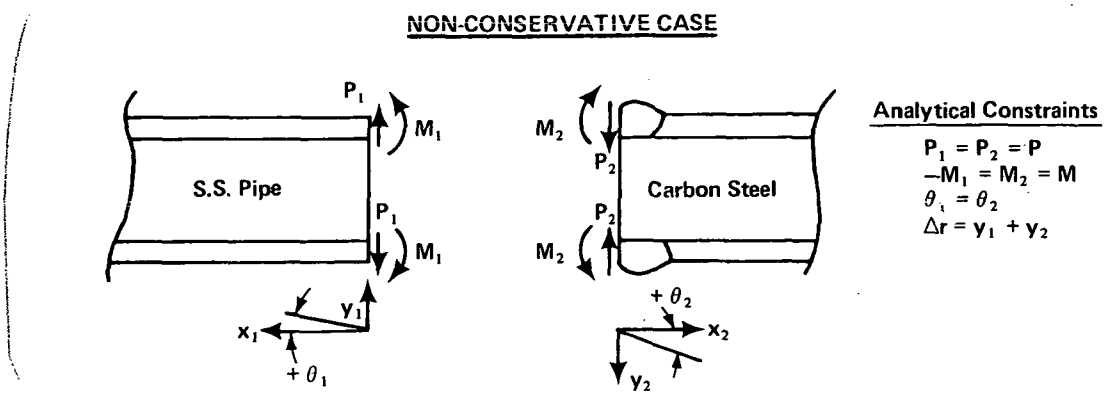
To evaluate the discontinuity stresses in the material transition region, I decided to apply the theory of beams on elastic foundations to this problem.[1,2] Two cases were employed in an attempt to bound the cause of the failures; i.e., one situation with conservative boundary conditions and the other with nonconservative boundary conditions. The first case consisted of assuming that the carbon section (flange) was infinitely stiff relative to the stainless steel pipe. These boundary conditions are shown below. The applicable equations are shown in Appendix A, Case A.

CONSERVATIVE CASE



These relationships predicted stress levels far beyond the yield point of the material [>206.85 MPa (30,000 psi)]. Since there was no evidence of such exceedingly high stress levels, this result was discarded.

The second and hopefully nonconservative boundary conditions consisted of treating the weld region as the juncture of two infinitely long pieces of pipe, one made of stainless steel, the other of carbon steel. The boundary conditions for this situation are shown below. The applicable equations are presented in Appendix A, Case B.



This analytical condition also predicted stress levels well beyond yield. From past experience, I knew that stress levels of this magnitude would produce local deformation that one could easily feel and in most cases could be seen by the naked eye. However, a careful inspection of the samples did not reveal the evidence necessary to support these conditions.

From this dilemma came the realization that the assumption of an abrupt change in material properties at the weld was actually incorrect. Proper treatment of the material properties along the fill pipe especially in the weld region should yield a more reasonable answer. This would, however, be a relatively time consuming, if not impossible, task to employ using the theory of beams on elastic foundations. But the finite element method would allow one to easily simulate materials variations and thus obtain reasonably good results even with a relatively coarse model. The more refined the model, the more accurate the solution. The axisymmetric nature of the stand pipe led to the development of two quarter models as shown in Figures 5 and 6.

The most important ingredients needed to obtain accurate analytical results involved the determination of the material properties and a reasonable estimate of the wall thickness of the elements in the weld region. The thickness data was obtained by measuring several sliced sections from a typical pipe weld. An estimate of the material properties across the weld was obtained from some special reports supplied by our metallurgist.

Both of the resulting models allowed one to vary the thickness and material properties as a function of the axial direction, Z or X_i. (See

Appendix B for more model details.) In both models the following assumptions were made:

1. The loads (thermal deformations) are axisymmetric with respect to the X_i axis.

2. The material properties vary along the weld region in the X_i direction but are constant in the circumferential (hoop) direction.

3. Hookes law is obeyed.

Model "a" consists of two straight sections of pipe, one 304 stainless steel, the other carbon steel. This situation, which is very similar to the second classical analysis (Case B), resulted in thermally induced stress levels below the yield point of the materials.

Model "b" is very similar to Model "a" but in this case the boundary conditions in the tapered section of the carbon steel were modified to simulate the rigidity of the flange. This model very closely simulates the real boundary conditions. The stress levels were somewhat higher than those determined from Model "a," but still below yield. Close inspection of the weld region indicated that a much more refined model would be required to determine the peak stress levels at the beginning of the weld near the heat affected zone. A plot showing the thermally deformed weld region is presented in Figure 7.

To compensate for the relatively coarse grid network used in Models "a" and "b", a suitable stress concentration factor was determined. Application of this stress amplication factor indicated that the peak stress levels in the region of the failure would be between 137.9 MPa and 172.37 MPa (20,000 to 25,000 psi) depending on the quality (roughness) of the weld.

These results definitely showed that one must reasonably account for material property variation across a weld joining two different materials if there is to be any hope of obtaining a reasonable answer. However, these stress levels, which agreed with the deformation patterns determined from visual inspection of two failed pipes, left me with another dilemma. At these stress levels and only a few load cycles, failure should not occur. In addition, the ductility should not have decreased substantially. This region would have had to be precracked during fabrication or there was something yet to be uncovered.

At this point, I talked with some of the maintenance personnel who were performing the field welding. They stated that occasionally when they tried to reweld the cracked region after proper grinding of the crack, the material would essentially crumble when they began welding. The material was seemingly very brittle, i.e., had lost its ductility. This was further evidence that some additional mechanism was contributing to the failures.

Fortunately, by this time the TEC Materials Laboratory was able to mount samples of a typical failure and obtain metallographic* photos. From these photos, the reason for the failures became obvious. It was stress-corrosion-cracking, SCC[3,4,5,6], as is evident in Figure 8. By this time failures of 304 S.S. bourdon tubes and stainless steel rupture disks were also being encountered. Typical metallographic photos of these failures are shown in Figures 9 and 10. Thus, the puzzle began to fit together. We had determined that relatively high stress levels existed in the weld region. We were also aware that relatively large residual stresses could be present in the heat affected zone below the weld. Such stress levels in the presence of the proper chemical would readily explain the fill pipe failures. Usually, chlorides are the chief suspect for SCC in 300 series stainless steels, but no chloride source was immediately obvious. Thus we had to establish that chlorides were at fault and not some other chemical, then determine the source of the chlorides or other chemical.

This led to a review of the procedure for manufacturing the spin blocks, the cleaning procedure and chemicals used for cleaning, and a check of the chloride concentration in the heat-transfer fluid being used in the spin block.

This review was essentially fruitless. There was nothing that obviously stood out as a source of the chlorides. The chloride level in the heat-transfer fluid was found to be between 2 and 3 ppm. This concentration level was not nearly sufficient to cause the SCC being encountered.

Again, we had a dilemma. At this time we contacted the Kodak Materials Laboratory at Kodak Park, Rochester, New York. Discussions with Kodak Material's engineers revealed that they had encountered corrosion problems in a closed system where too much moisture (H_2O) was present. They had found that the moisture would move to the coolest part of the system where condensation and revaporization would occur repeatedly. In our system such an internal reflux condition would serve to leach chlorides from the heat-transfer fluid and concentrate them in the revaporization zone.

Subsequent infra-red temperature measurements substantiated the reflux theory. As shown in Figure 11, the temperatures in the upper carbon steel portion of the fill pipes that had previously failed were quite cool in relation to the heat-transfer fluid temperature. Thus, the failures in this relatively high temperature environment were attributed to stress-corrosion-cracking which was a result of:

1. The welding of two materials with different coefficients of thermal expansion;

2. The high operating temperature and the associated stress fields; and

3. The presence of a mechanism to concentrate chlorides, i.e., a water/steam reflux cycle.

*A standard polishing procedure with an acid etch was employed.

The fill pipes that had not failed, did not show a temperature change in the weld to flange region. This indicated that these units did not contain sufficient moisture to establish the required water/steam reflux system for concentrating chlorides.

From these results it was concluded that the problem could be corrected by either changing the flange material or by removing the moisture from the system. The latter corrective action was taken.

Thus, through the cooperation and assistance from maintenance personnel (welders), production personnel, and material engineers at TEC and Kodak, I was finally able to establish the complete mechanism that was causing the failures. Such, interdisciplinary interactions are often a necessity in solving such unusual problems.

CONCLUSION

This analysis clearly shows the merits of the finite element method as available in NASTRAN. This capability allows the engineer to much more closely simulate real life situations and thus obtain more reasonable/accurate results. This is particularly helpful in failure analysis since conservative results do little more than cover up the real cause of a failure. Only in the design process should one consider conservative methods. But even in this case, one must be very cautious.

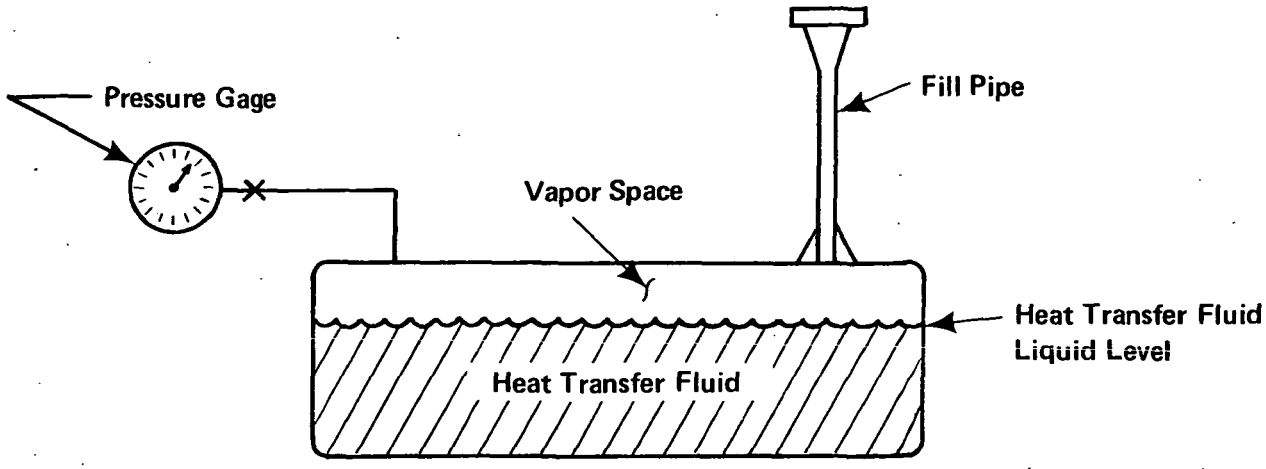


Figure 1: SPIN Block Schematic

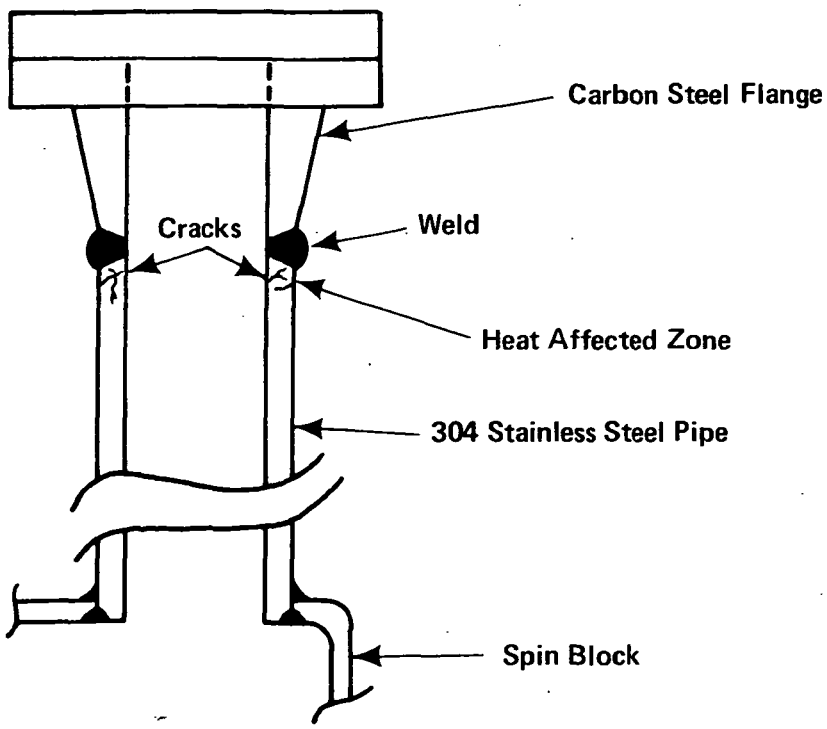


Figure 2: Fill Pipe Detail

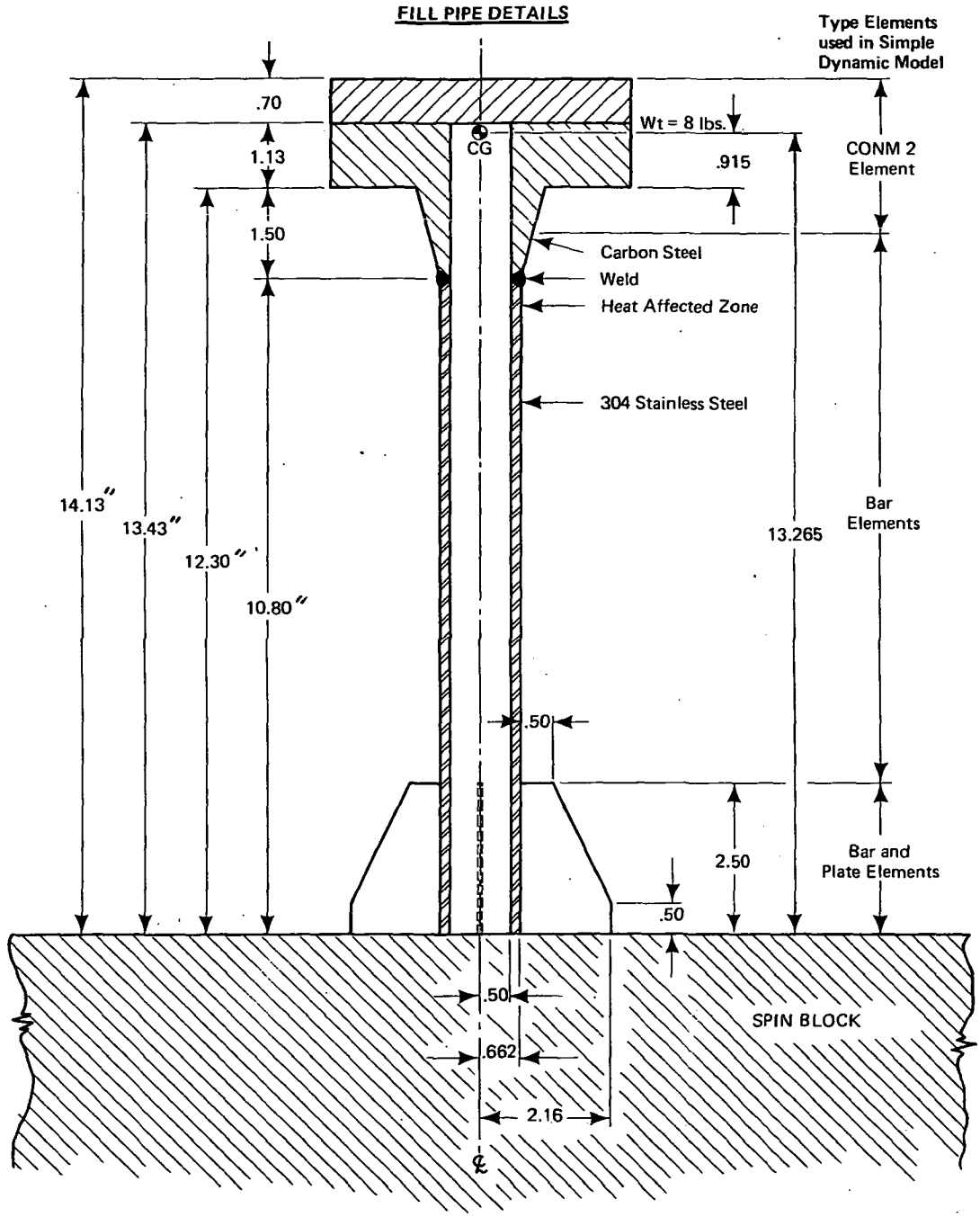


Figure 3: Fill Pipe Schematic Drawing

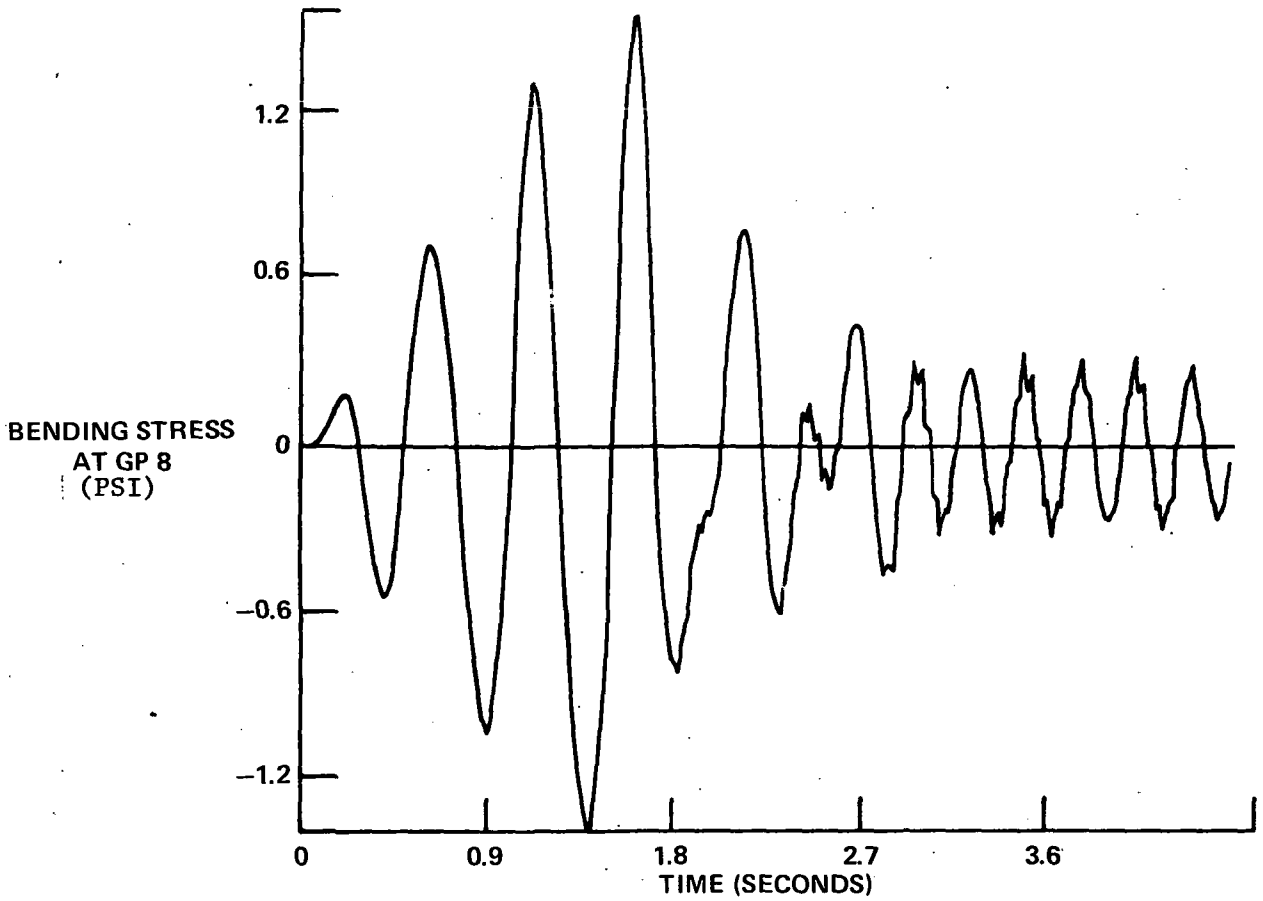


Figure 4: Bending Stress vs Time (Force Response Analysis)

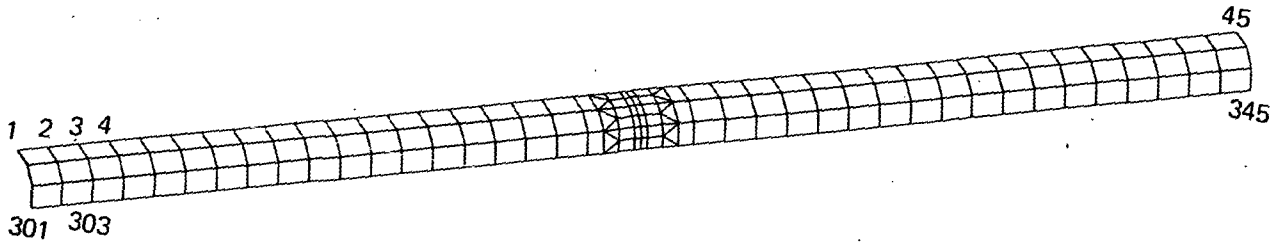


Figure 5: NASTRAN Model a

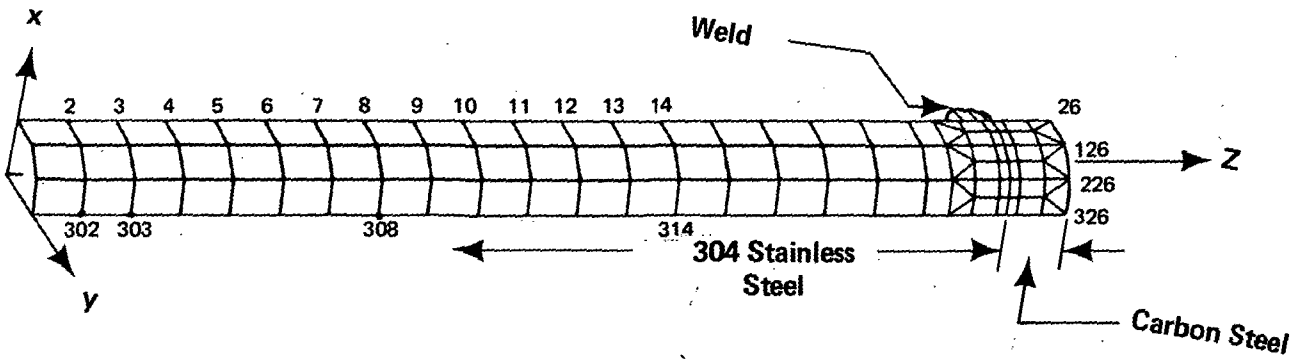


Figure 6: NASTRAN Model b

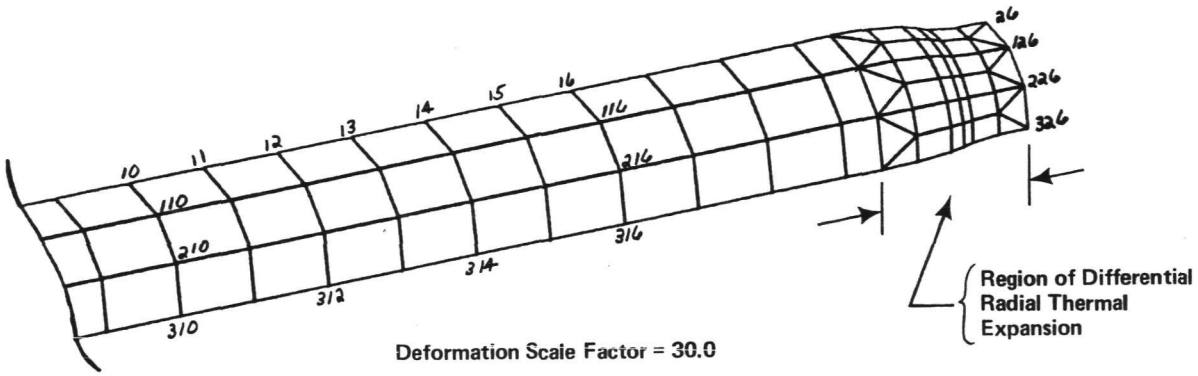


Figure 7: NASTRAN Deformed Plot of Model b

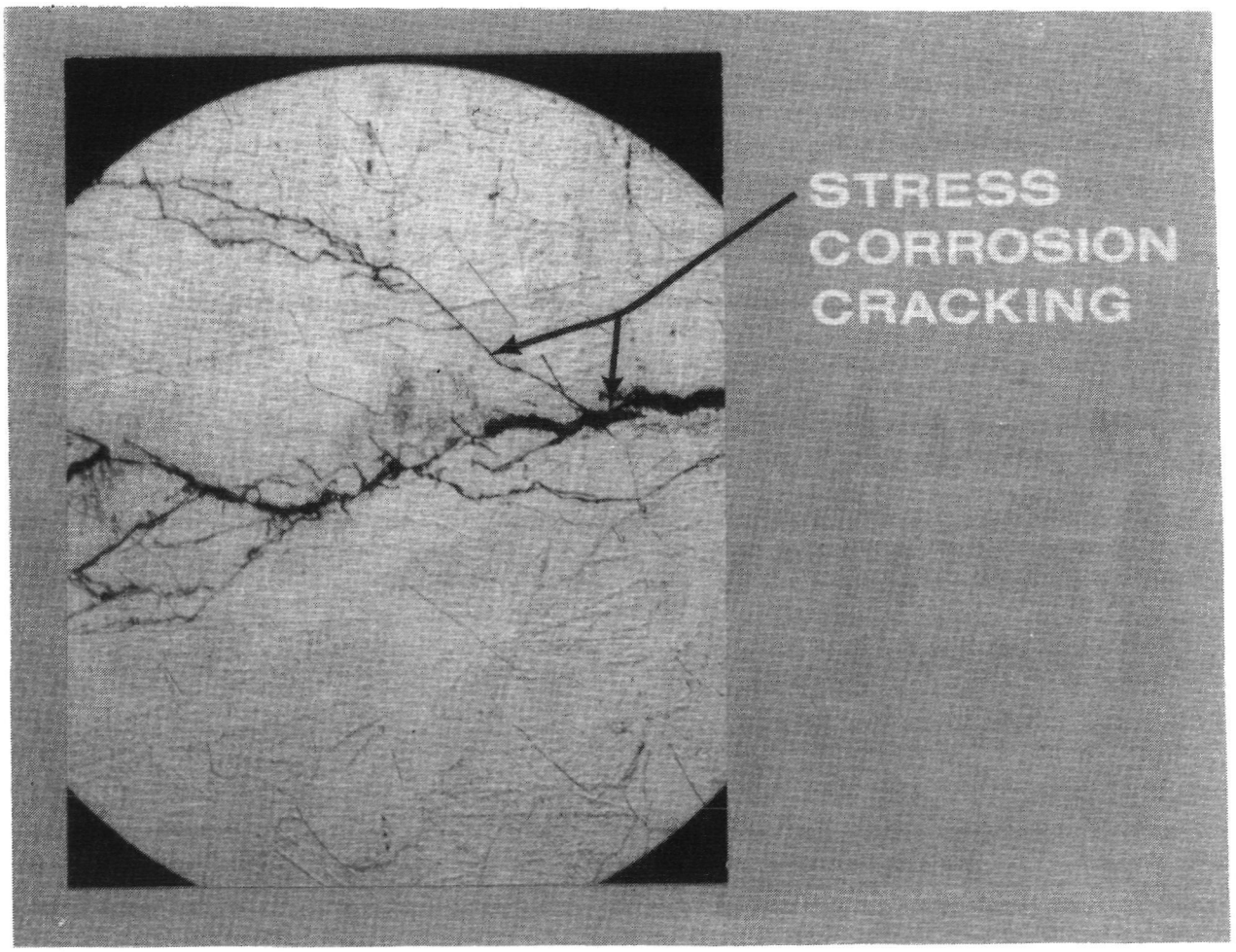


Figure 8: Photomicrograph Showing Stress Corrosion Cracking in the 304 Stainless Steel Transition Region of the Fill Pipe

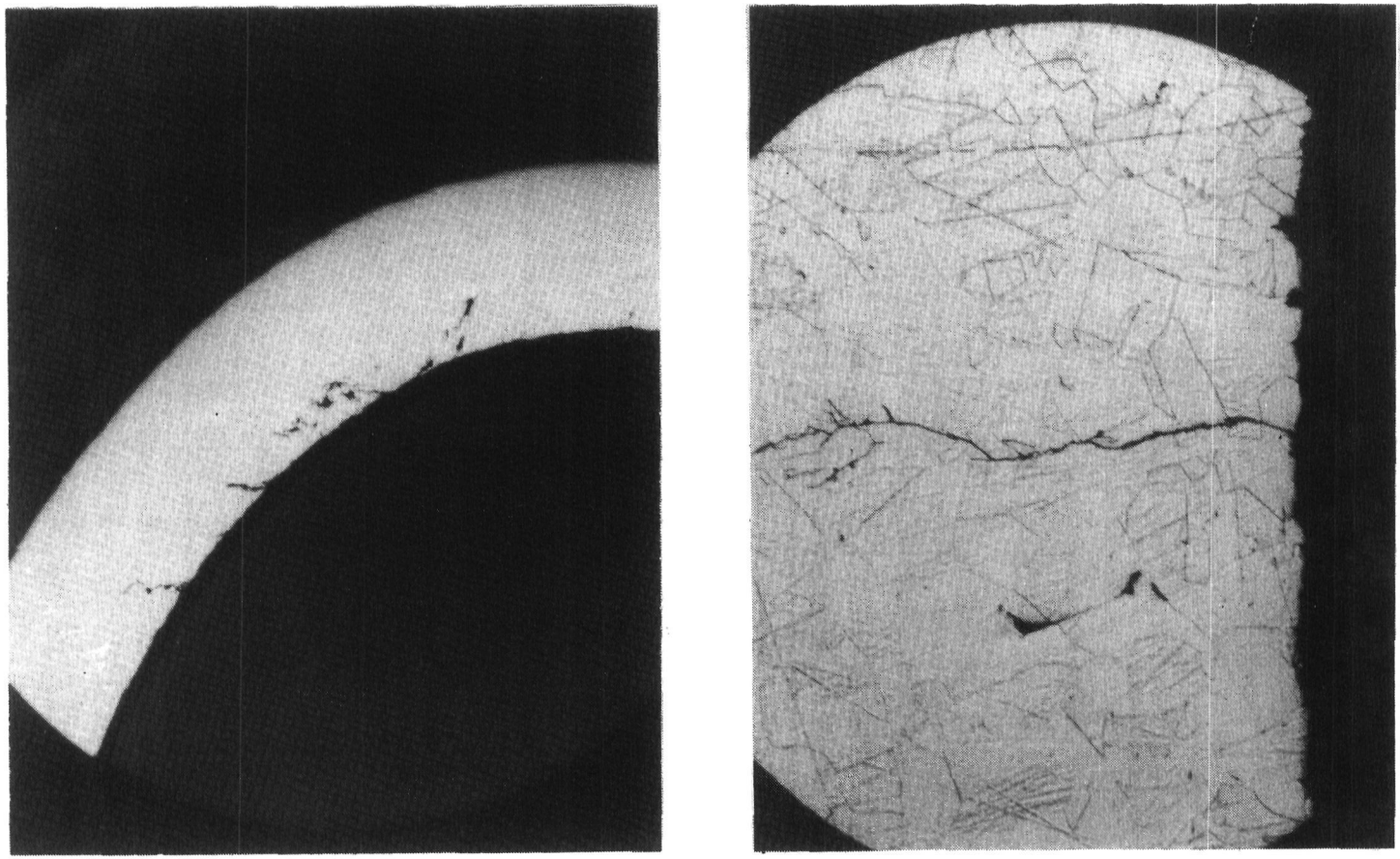


Figure 9: Photomicrographs Showing Stress Corrosion Cracking in a 304 Stainless Steel Bourdon Tube

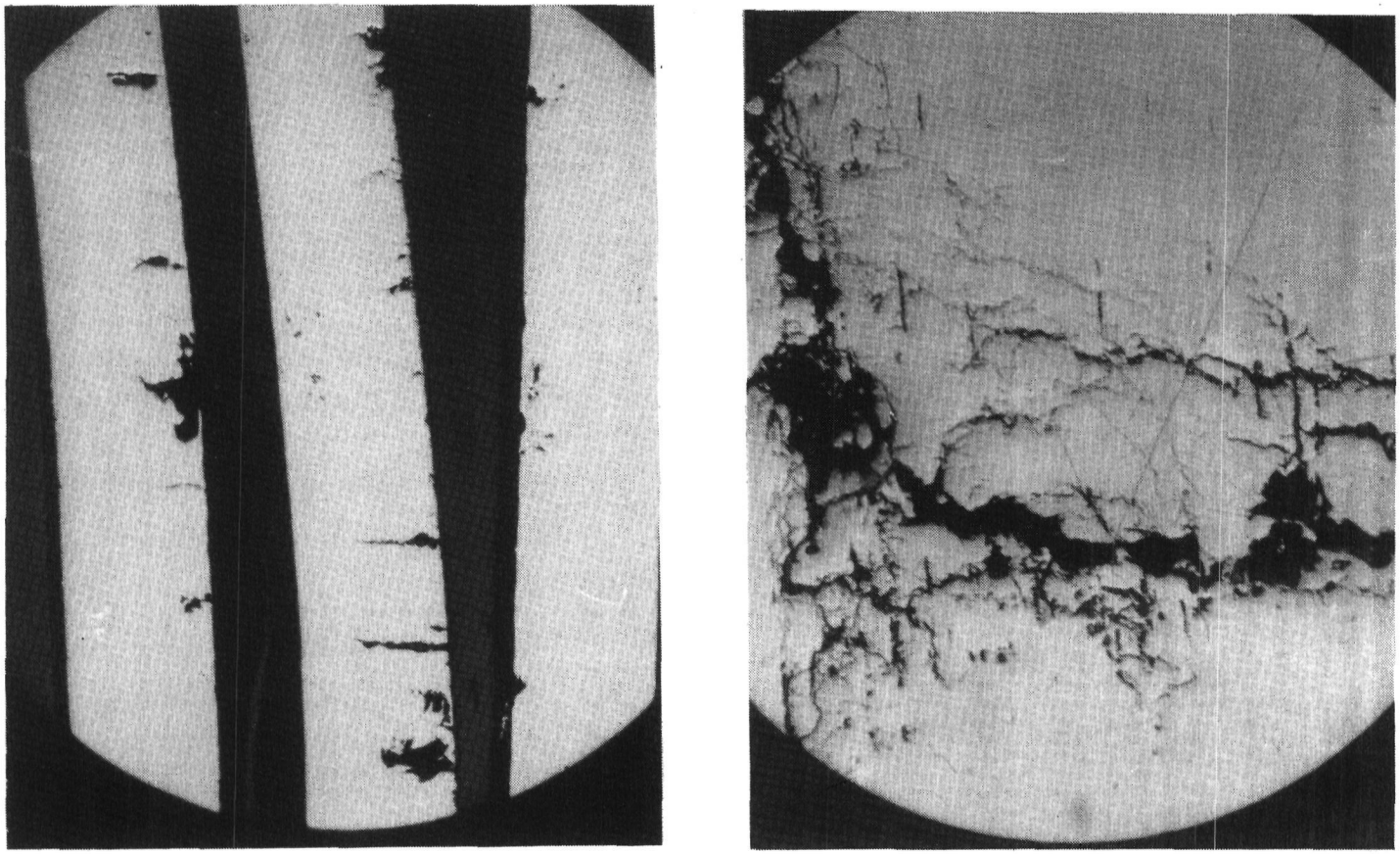


Figure 10: Additional Photomicrographs Showing Stress Corrosion Cracking in Other Fill Pipes and in the 304 Stainless Steel Pressure Relief Piping System

Temperature (°F)					
Spin Block No.	T_1	T_2	T_3	T_4	T_5
2-7	300	200	<200	<200	<200
2-8	290	200	<200	<200	<200
2-9	470	475	475	320	200
3-8	–	200	<200	<200	<200

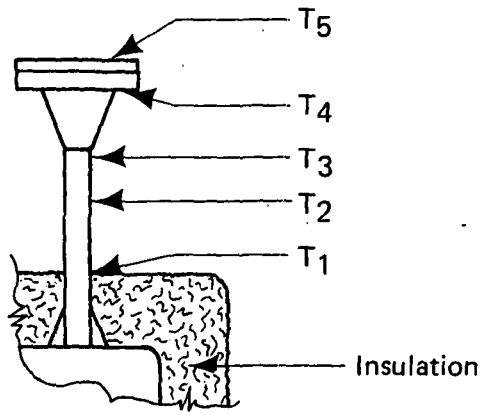


Figure 11: Thermal Profiles of Fill Pipes That Had Previously Failed

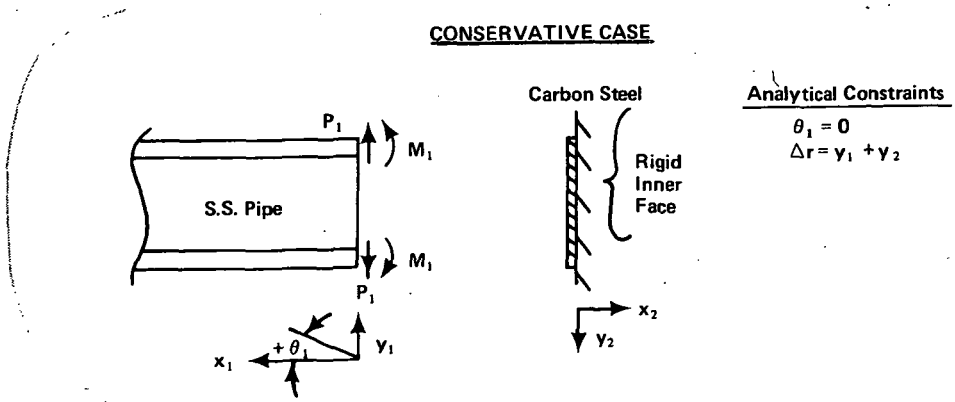
APPENDIX A

Nomenclature: (See References 1 and 2.)

P_i = Shear load (lbf)
M_i = Moment (in-lbs)
E_i = Modulus of Elasticity (psi)
υ = Poisson's Ration
r_i = Outside Radius of Pipe (inches)
t_i = Wall Thickness (inches)
T = Temperature (°F)
Θ_i = Slope
y_i = Deflection (inches)
$I_i = bt^3/12 = t^3/12$

Case A: Conservative Case
Stainless Steel Pipe to a Rigid Carbon Steel Block

Assumption: Stainless steel pipe is infinitely long, thus semi-infinite beam on elastic foundation theory is applicable.



Analytical Conditions and Data

$y \neq 0$
$\Theta_1 = 0$
$T = 536°F = T_1 = T_2$
$T_0 = 72°F$
$\alpha_1 = 9.76 \times 10^{-6}$ in/in/°F
$\alpha_2 = 7.12 \times 10^{-6}$ in/in/°F
$r_1 = .73"$
$t_1 = .137"$

Differential Radial Expansion

$$\Delta r = (r_1\alpha_1 - r_2\alpha_2)(T - T_0)$$

Applicable Equations

$$y_1 = \left[P_1/2E_1I_1\beta_1^3 \right] D_{\beta x} - \left[M_1/2E_1I_1\beta_1^2 \right] C_{\beta x}$$

$$\theta_1 = -\left[P_1/2E_1I_1\beta_1^2 \right] A_{\beta x} + \left[M_1/\beta_1 E_1 I_1 \right] D_{\beta x}$$

Final Equations

(1) $y_1 = \Delta r$

which leads to

$$-.175P_1 + .716M_1 = -14,400$$

(2) $\theta_1 = 0$

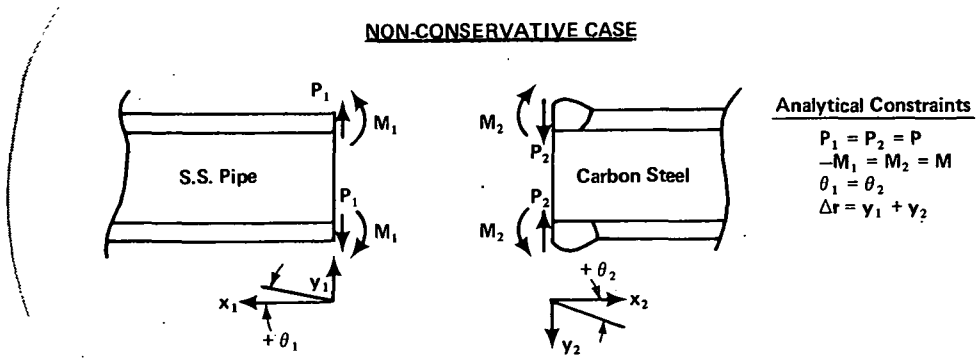
which leads to

$$P = 2\beta_1 M_1 = 8.18M_1$$

Case B: Non-Conservative Case

Carbon Steel - Stainless Steel Pipe Junction

Assumption: Both pipes are infinitely long, i.e., semi-infinite beams on an elastic foundation.



Analytical Conditions and Data

$-M_1 = M_2 = M$
$P_1 = P_2 = P$
$\Theta_1 = \Theta_2$
$\Delta r = y_1 + y_2$
$T = 536°F = T_1 = T_2$
$\Delta T_1 = \Delta T_2$

$\alpha_1 = 9.76 \times 10^{-6}$ in/in/°F
$\alpha_2 = 7.12 \times 10^{-6}$ in/in/°F
$r_1 = .73"$
$r_2 = .783"$
$t_1 = .135"$
$t_2 = .240"$

Differential Radial Expansion

$$\Delta r = r_1 d_1 (\Delta T_1) - r_2 \alpha_2 \Delta T_2 = y_1 + y_2$$
$$\Delta r = \Delta T(r_1 d_1 - r_2 d_2)(T - T_0)$$

Applicable Equations

$$y_i = \left[P_i / 2E_i I_i \beta_i^3\right] D_{\beta x} - \left[M_i / 2E_i I_i \beta_i^2\right] C_{\beta x}$$
$$\Theta_i = -\left[P_i / 2E_i I_i \beta_i^2\right] A_{\beta x} + \left[M_i / \beta_i E_i I_i\right] D_{\beta x}$$

$$\beta_i = \sqrt[4]{3(1 - \upsilon^2)/r^2 t^2}$$

$$\begin{aligned} A_{\beta x} &= e^{-\beta x}\,[\cos(\beta x) + \sin(\beta x)] \\ B_{\beta x} &= e^{-\beta x}\,[\sin(\beta x)] \\ C_{\beta x} &= e^{-\beta x}\,[\cos(\beta x) - \sin(\beta x)] \\ D_{\beta x} &= e^{-\beta x}\,[\cos(\beta x)] \end{aligned}$$

Final Equations

(1) $\Delta r = y_1 + y_2$

which leads to

$.5424P - .373M = 7200$

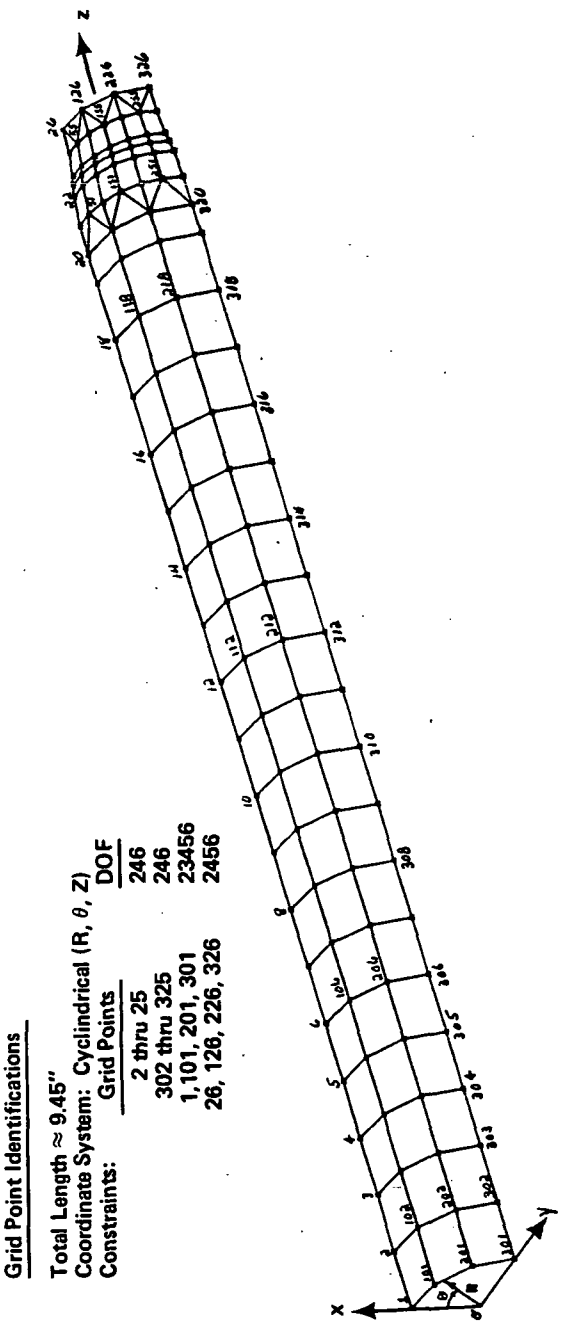
(2) $\Theta_1 = \Theta_2$ or $\Theta_1 - \Theta_2 = 0$

which leads to

$P = 33.04M$

APPENDIX B

Additional NASTRAN Model Details



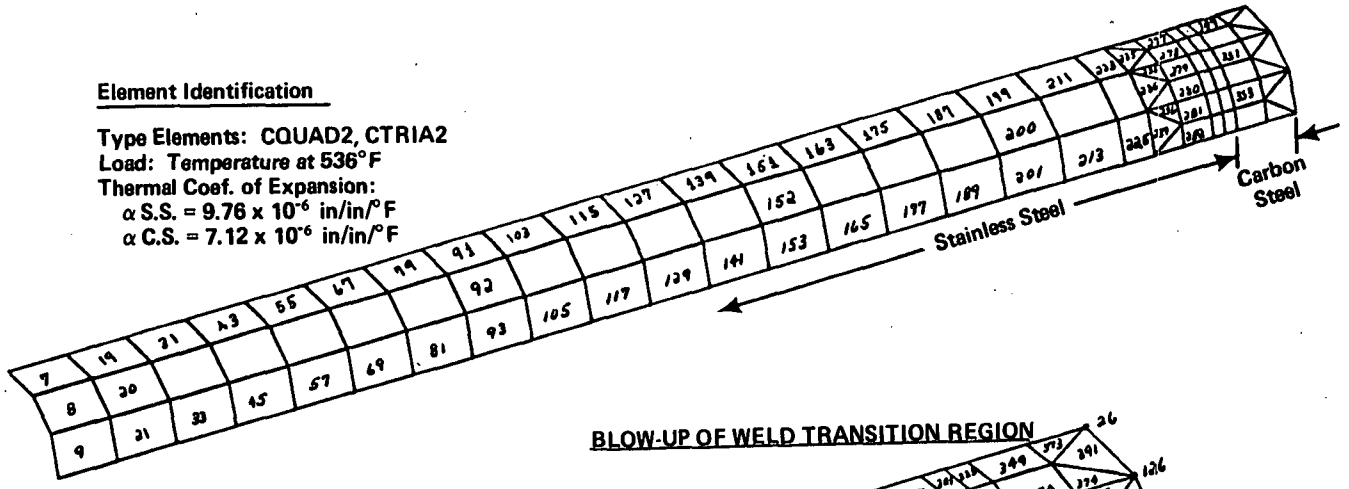
Grid Point Identifications

Total Length ≈ 9.45"
Coordinate System: Cylindrical (R, θ, Z)
Constraints:

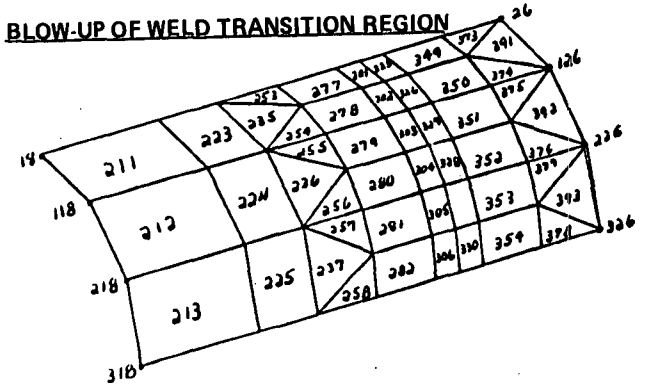
Grid Points	DOF
2 thru 25	246
302 thru 325	246
1, 101, 201, 301	23456
26, 126, 226, 326	2456

Element Identification

Type Elements: CQUAD2, CTRIA2
Load: Temperature at 536°F
Thermal Coef. of Expansion:
α S.S. = 9.76 x 10^{-6} in/in/°F
α C.S. = 7.12 x 10^{-6} in/in/°F



BLOW-UP OF WELD TRANSITION REGION



REFERENCES

1. Theory and Design of Modern Pressure Vessels, 2nd Edition, by John F. Harvey, pp. 134-166, especially Section 4.8.

2. Beams on Elastic Foundations, by M. Hetenyi, pp. 30-37.

3. Stress Corrosion Cracking in High Strength Steels and in Titanium and Aluminum Alloys, edited by B. F. Brown, Naval Research Laboratory. (U.S. Government Printing Office, Stock No. 0851-0058).

4. Stress Corrosion Cracking Control Measures, National Bureau of Standards, Monograph 156, B. F. Brown, 1977.

5. Success by Design; Progress Through Failure Analysis, National Bureau of Standards (NBS), Publication 433, MPFG 25th Meeting, edited by T. R. Shives and W. A. Willard, 1974.

6. Engineering Design, MFPG 25th Meeting, National Bureau of Standards, Special Publication 487, 1977.

7. Mechanical Failure; Definition of the Problem, MFPG 20th Meeting, National Bureau of Standards Special Publication 423, 1976.

8. NASTRAN User's Manual, NASA SP-222(05), Available from COSMIC, University of Georgia.

A NASTRAN INVESTIGATION OF SIMULATED PROJECTILE DAMAGE EFFECTS ON A UH-1B TAIL BOOM MODEL

Arnold T. Futterer
U. S. Army Armament Research and Development Command
Ballistic Research Labratory

SUMMARY

A NASTRAN model of a UH-1B tail boom that had been designed for a another project was used to investigate the effect on structural integrity of simulated projectile damage. Elements representing skin, and sections of stringers, longerons and bulkheads were systematically deleted to represent projectile damage. The structure was loaded in a manner to represent the flight loads that would be imposed on the tail boom at a 130 knot cruise. The deflection of four points on the rear of the tail boom relative to the position of these points for the unloaded, undamaged condition of the tail boom was used as a measure of the loss of structural rigidity. The same procedure was then used with the material properties of the aluminum alloys replaced with the material properties of T300/5208 high strength graphite/epoxy fibrous composite material, $(0,\pm45,90)_s$ for the skin and $(0,\pm45)_s$ for the longerons, stringers, and bulk heads.

INTRODUCTTON

This investigation had a two-fold objective:

1. To determine the effect on the structural integrity of the UH-1B tail boom caused by threat projectile damage.

2. To estimate the effect of composite materials on the stiffness of the tail boom.

The model of the UH-1B tail boom used in the analysis was originally prepared under contract by Kamen AviDyne[1] (KAD) for the Ballistic Research Laboratory. The model consisted of beams representing sections of the stringers, bulk heads, and longerons and thin plates representing the skin. The KAD report describes the model in good detail. Figures 1, 2, and 3 taken from the KAD report illustrate the NASTRAN[2] model developed by KAD and give the numbering schemes for the grid points, beam elements and plate elements, respectively. The skin is made of 2024 T3 aluminum alloy with a modulus of elasticity of 7.31×10^4 MPa (10.6×10^6 psi) and a mass density equal to 271.15 kg sec^2/m^4 (0.00025 lb sec^2/in^4). The stringers,bulk heads and longerons are made of 7075 T6 aluminum alloy with a modulus of elasticity of 7.10×10^4 MPa (10.3×10^6 psi) and a mass density of 271.15 kg sec^2/m^4 (.00025 lb sec^2/in^4). For further detail on the assumptions that went into preparing the

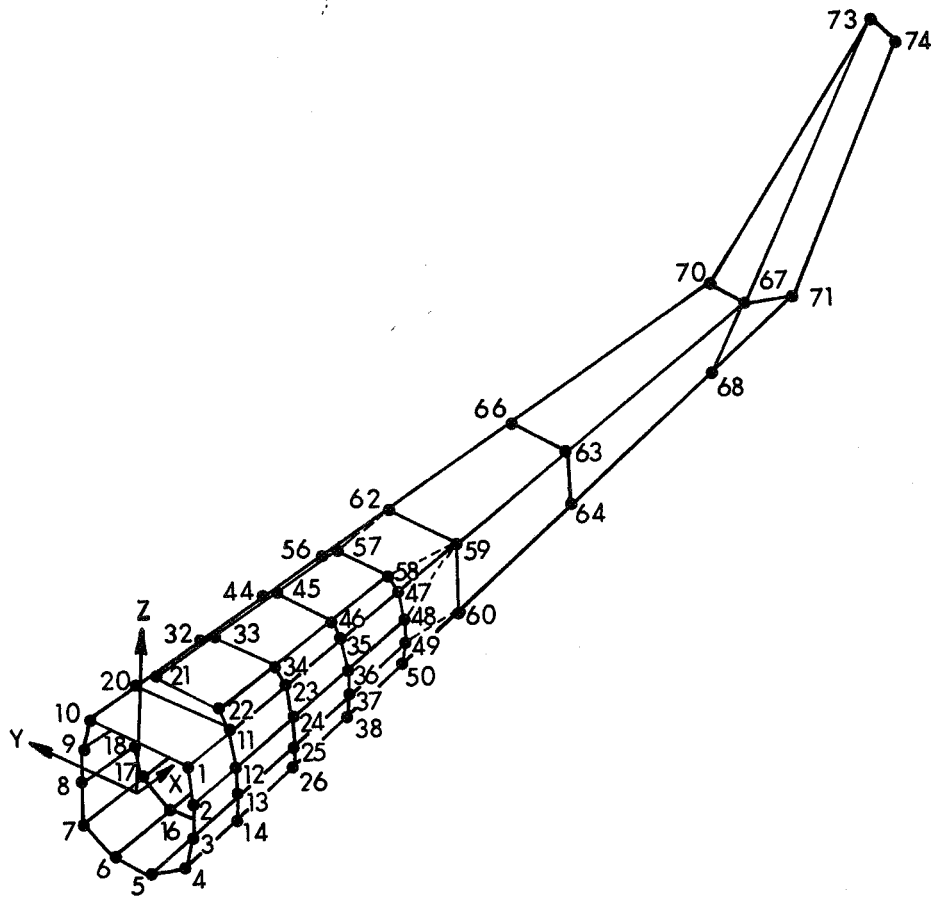


FIGURE 1. UH-1B TAIL BOOM MODEL GRID POINTS AND NUMBERING SEQUENCE

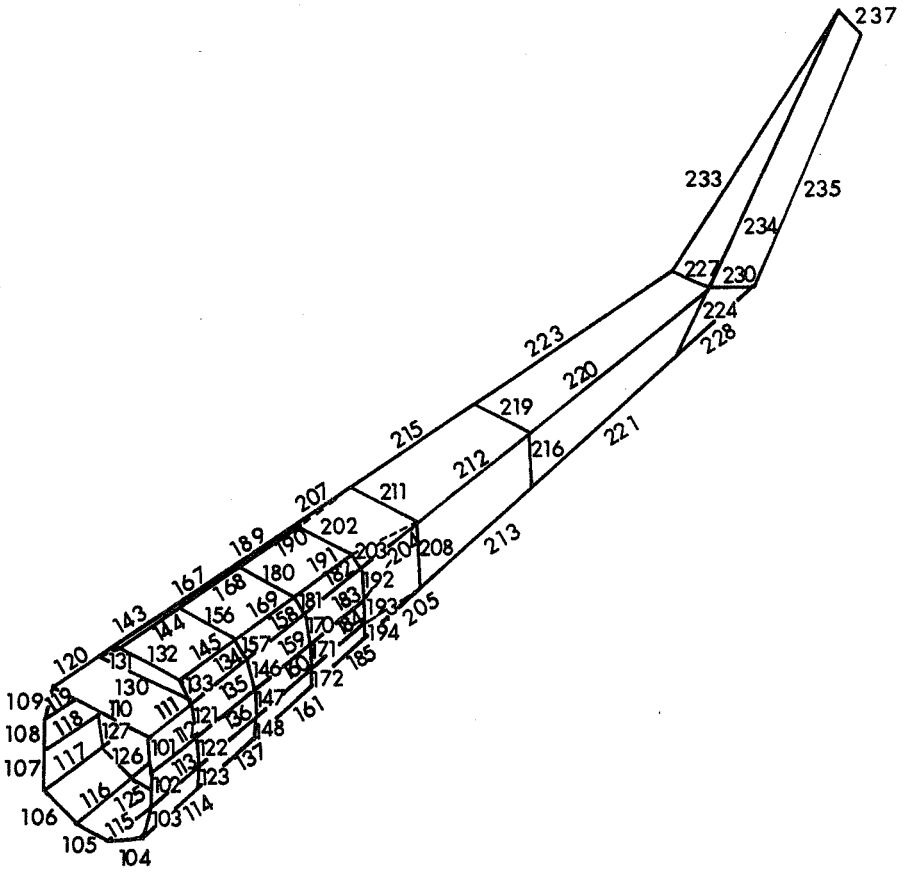


FIGURE 2. UH-1B TAIL BOOM MODEL BEAM ELEMENT IDENTIFICATION

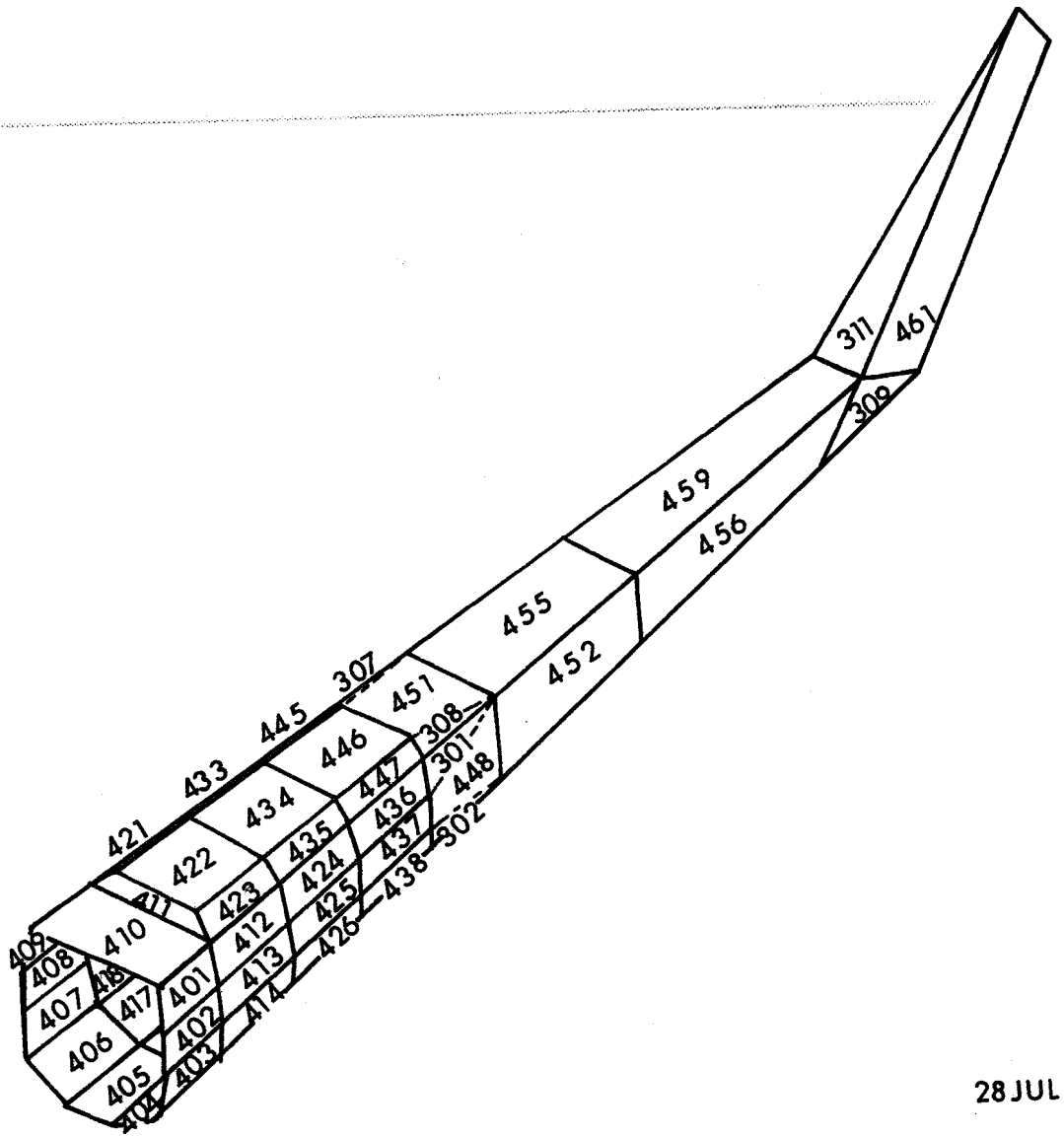


FIGURE 3. UH-1B TAIL BOOM MODEL PLATE ELEMENT IDENTIFICATION

model it is recommended that a copy of the report be obtained from Defense Technical Information Center.

PROCEDURE

The investigation was accomplished by using the NASTRAN code, a complex finite element method computer program. In all 35 NASTRAN calculations were made, 17 using the aluminum alloy construction and 18 using the same structure but using the material properties of T300/5208 high strength, graphite-epoxy fibrous composite. This paper describes the results from 18 calculations (9 aluminum alloy construction and 9 composite construction) The relative displacements, compared to the non-damaged, non-loaded structure, of the four points at the base of the tail fin with the front of the tail boom anchored securely were used as a measure of the deterioration of the structural integrity of the tail boom due to projectile damage. The flight loads that would be imposed on a helicopter cruising at 130 knots were simulated by loading the structure to simulate the loads and torques that the rotor thrust and elevator loading would cause at cruise velocity. The assumption was made that a large hole or tear in a skin panel or a break in a longeron, stringer or bulkhead section would destroy the structural integrity of the element representing that skin panel or section. In the model that damaged element was then deleted. The investigation was conducted by systematically deleting plate and beam elements to simulate greater damage. Table I gives the nomenclature of the damage configurations that were investigated. Damage to

TABLE I. NOMENCLATURE FOR DAMAGE CONFIGURATIONS

Nomenclature	Elements deleted
0	No elements deleted
1	419
2	419, 141, 418
3	431, 153, 419
4	431, 153, 419, 141, 418
5	419, 154, 431, 165, 430, 153, 418, 141
6	419, 154, 431, 165, 430, 153, 418, 141, 164, 140
7	458
8	458, 222

the left and to the right side of the tail boom were studied because the right side has thicker skin than the left. Furthermore, the longitudinal strains were generally compressive on the right side and tensile on the left side. This paper discusses the results of calculations for damage to the right side of the helicopter. The numbers of the elements deleted refer to Figures 2 and 3. As may be noted, the 100 and 200 series numbers refer to the beam elements and the 400 series to the rectangular plate elements.

Tables II, and III are similar in construction. They give the displacements of the points at the rear of the tail boom of the loaded, undamaged tail boom and the loaded tail boom with simulated damage relative to the undamaged, unloaded state of the tail boom. Nomenclature refers to Table I where the damage configurations are set forth. Material lost refers to the mass of the deleted elements. "Direction",X, Y, and Z, gives the displacement of the grid points in the three coordinate directions given on Figure 1 and "R", which is the square root of the sum of the squares of the three coordinate displacements, gives the total displacement of the grid points specified. The displacements are given in inches and millimeters. The displacement values are given in exponential format, i.e., 2.05-2 means 2.05×10^{-2}. Table II contains the results of the calculations with the tail boom constructed of aluminum alloy and the damage is to the forward right side of the helicopter. Figure 4 is a graph of deflection of points on the rear of the tail boom versus mass of aluminum alloy removed from the forward right side of the tail boom. As may be noted, the deflection is quite linear with mass removed until about 1 kg and then further removal causes the displacements to become non-linear suggesting a more rapid approach to failure with further loss of material.

After calculating the displacements for the various damage configurations with the tail boom constructed of aluminum alloys, the calculations were repeated using the material properties of T300/5208 which is a high strength, graphite-epoxy fibrous composite. The skin plates were assumed to be constructed of $[0,\pm45,90]_s$ layered composite and the beam elements of $[0,\pm45]_s$ layered composite. T300/5208 was recommended[3] as being high strength and considerably less expensive than the ultra-high modulus graphite-epoxy. Since the composites have less strength in compression than in tension, the material moduli of elasticity for both compression and tension were used in the calculations. For damage on the left side the tensile moduli, 5.59×10^4 MPa (8.11×10^6 psi) for the plate elements and 6.50×10^4 MPa (9.42×10^6 psi) for the beam elements, were used. For damage on the right side the compressive moduli, 5.38×10^4 MPa (7.81×10^6 psi) for the plate elements and 6.43×10^4 MPa (9.32×10^6 psi) for beam elements, were used. A mass density of 162.69 kg sec^2/m^4 (.00015 lb $sec^2/in.^4$) was used for the beam and plate elements. This paper reports on the more extreme of the two cases, the right side damage and using the lower compressive moduli in the calculations. The total displacements for the various damage configurations for damage done to the forward right side of a tail boom constructed of T300/5208 composite are

TABLE II. DAMAGE TO THE RIGHT SIDE OF HELICOPTER TAIL BOOM CONSTRUCTED OF ALUMINUM ALLOY.

Nomenclature	Material Lost lb/kg	Direction	Grid Point Displacements							
			67		70		71		72	
			in.	mm	in.	mm	in.	mm	in.	mm
0	.0	X	2.18-2	5.54-1	2.45-3	6.22-2	7.52-3	1.91-1	-1.23-2	-3.12-1
	.0	Y	3.30-1	8.38+0	3.30-1	8.38+0	3.23-1	8.20+0	3.23-1	8.20+0
		Z	-2.22-1	-5.64+0	-2.47-1	-6.27+0	-2.46-1	-6.25+0	-2.70-1	-6.86+0
		R	3.98-1	1.01+1	4.12-1	1.05+1	4.06-1	1.03+1	4.21-1	1.07+1
1	.71	X	2.17-2	5.51-1	2.02-3	5.13-2	7.35-3	1.87-1	-1.27-2	-3.23-1
	.32	Y	3.36-1	8.53+0	3.36-1	8.53+0	3.29-1	8.36+0	3.29-1	8.36+0
		Z	-2.24-1	-5.69+0	-2.50-1	-6.35+0	-2.50-1	-6.35+0	-2.74-1	-6.96+0
		R	4.04-1	1.03+1	4.19-1	1.06+1	4.13-1	1.05+1	4.28-1	1.08+1
2	1.46	X	2.14-2	5.44-1	2.21-3	5.61-2	6.49-3	1.65-1	-1.49-2	-3.78-1
	.66	Y	3.60-1	9.14+0	3.60-1	9.14+0	3.53-1	8.97+0	3.53-1	8.97+0
		Z	-2.36-1	-5.99+0	-2.64-1	-6.71+0	-2.62-1	-6.65+0	-2.88-1	-7.32+0
		R	4.30-1	1.09+1	4.46-1	1.13+1	4.40-1	1.12+1	4.56-1	1.16+1
3	1.46	X	2.17-2	5.51-1	1.55-3	3.89-2	7.29-3	1.85-1	-1.31-2	-3.33-1
	.66	Y	3.43-1	8.71+0	3.43-1	8.71+0	3.34-1	8.48+0	3.34-1	8.48+0
		Z	-2.28-1	-5.79+0	-2.55-1	-6.48+0	-2.53-1	-6.43+0	-2.79-1	-7.09+0
		R	4.12-1	1.05+1	4.27-1	1.08+1	4.19-1	1.06+1	4.35-1	1.10+1
4	2.33	X	2.13-2	5.41-1	-1.99-3	-5.05-2	6.63-3	1.68-1	-1.50-2	-3.81-1
	1.06	Y	3.64-1	9.25+0	3.64-1	9.25+0	3.54-1	8.99+0	3.54-1	8.99+0
		Z	-2.40-1	-6.40+0	-2.70-1	-6.86+0	-2.66-1	-6.76+0	-2.94-1	-7.47+0
		R	4.36-1	1.11+1	4.53-1	1.15+1	4.43-1	1.13+1	4.60-1	1.17+1
5	3.00	X	2.12-2	5.38-1	-1.24-3	-3.15-2	6.04-3	1.53-1	-1.64-2	-4.11-1
	1.36	Y	3.78-1	9.60+0	3.78-1	9.60+0	3.69-1	9.37+0	3.69-1	9.37+0
		Z	-2.51-1	-6.38+0	-2.82-1	-7.16+0	-2.77-1	-7.04+0	-3.06-1	-7.77+0
		R	4.54-1	1.15+1	4.72-1	1.20+1	4.61-1	1.17+1	4.79-1	1.22+1

TABLE II CONT. DAMAGE TO THE RIGHT SIDE OF HELICOPTER TAIL BOOM CONSTRUCTED OF ALUMINUM ALLOY.

Nomenclature	Material Lost lb/kg	Direction	Grid Point Displacements							
			67		70		71		72	
			in.	mm	in.	mm	in.	mm	in.	mm
6	3.90	X	2.12-2	5.38-1	-3.25-3	-8.25-2	4.45-3	1.13-1	-1.96-2	-4.98-1
	1.77	Y	4.08-1	1.04+1	4.08-1	1.04+1	4.02-1	1.02+1	4.02-1	1.02+1
		Z	-2.73-1	-6.93+0	-3.04-1	-7.72+0	-3.01-1	-7.65+0	-3.31-1	-8.41+0
		R	4.91-1	1.25+1	5.09-1	1.29+1	5.02-1	1.28+1	5.21-1	1.32+1
7	3.18	X	2.62-2	6.65-1	-6.38-4	-1.62-2	8.04-3	2.04-1	-1.07-2	2.72-1
	1.44	Y	3.51-1	8.92+0	3.51-1	8.92+0	3.03-1	7.70+0	3.03-1	7.70+0
		Z	-2.47-1	-6.27+0	-3.33-1	-8.46+0	-2.74-1	-6.96+0	-3.48-1	-8.84+0
		R	4.30-1	1.09+1	4.84-1	1.23+1	4.09-1	1.04+1	4.62-1	1.17+1
8	4.25	X	2.24-2	5.69-1	4.42-4	1.12-2	7.77-3	1.97-1	-1.10-2	-2.79-1
	1.93	Y	3.53-1	8.97+0	3.53-1	8.97+0	3.03-1	7.70+0	3.03-1	7.70+0
		Z	-2.48-1	-6.30+0	-3.37-1	-8.56+0	-2.76-1	-7.01+0	-3.52-1	-8.94+0
		R	4.32-1	1.10+1	4.88-1	1.24+1	4.10-1	1.04+1	4.65-1	1.18+1

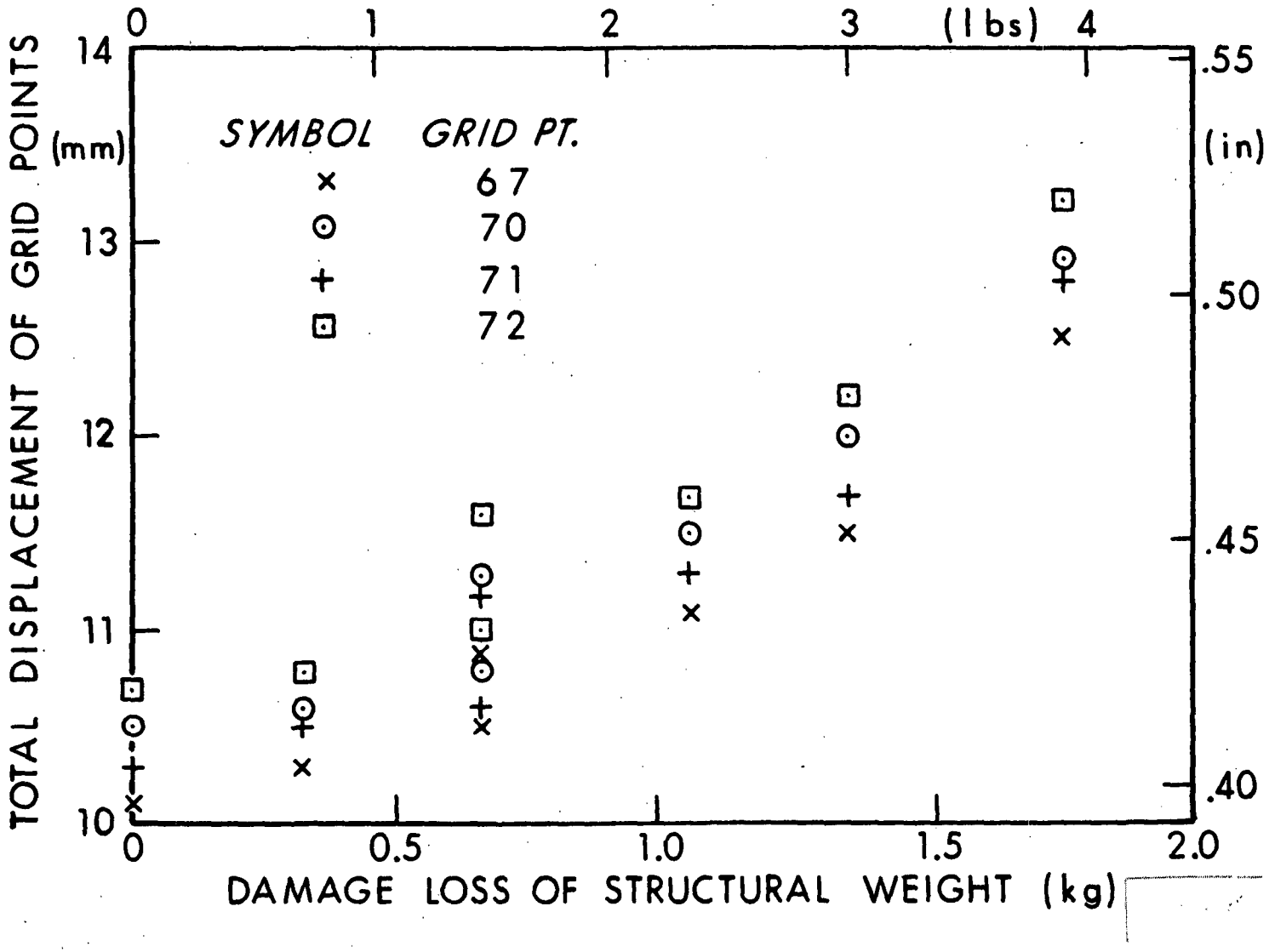


FIGURE 4. GRID POINT DISPLACEMENT VS. DAMAGE LOSS OF STRUCTURAL WEIGHT--ALUMINUM ALLOY CONSTRUCTION

compiled in Table III. Figure 5 is a graph of the relative displacement of points on the rear of the tail boom versus composite material lost through simulated projectile damage to the forward right side. As is to be expected, the non-linear behavior of the curve shows up at lower mass loss due to the lower strength of the material for a given cross-section.

A further consideration is that the tail boom constructed of aluminum alloy had a structural weight of 57.7 kg (127.1 lb) and in the model additional non-structural weight amounting to 24.0 kg (52.9 lb) was distributed along the length of the tail boom. Replacement by the lighter weight composite would result in a weight saving of from 23.1 kg (50.9 lb) up to 32.7 kg (72.0 lb) depending to what extent the aluminum alloy could be replaced by the composite material. This reduction in weight could make the helicopter more maneuverable or able to carry a larger payload. A further consideration is whether use of the more expensive ultra-modulus composite with a modulus almost twice that of T300/5208 and about 50% greater than the aluminum alloy might be warranted. The density of the ultra-modulus graphite/epoxy is only about two percent greater than that of T300/5208.

CONCLUDING REMARKS

The study completed here leaves many questions while providing some answers. Subjects of further study should be to what extent: the loss of structural rigidity of the tail boom can be tolerated; does a composite react to projectile damage better or worse than the aluminum alloy. Can the price differential of the ultra-high modulus composite be tolerated in the construction of the tail boom taking into consideration its much better material qualities?

REFERENCES

1. The NASTRAN User's Manual (Level 17.0),NASA SP-222(04),Dec. 31, 1977, National Aeronautics and Space Agency,Wash., DC

2. Yeghiayan, Raffi P.: Modeling of the UH-1B Tail Boom for Analysis by the NASTRAN Computer Program, ARBRL-CR-00358, Feb. 1978, Kamen AviDyne, Burlington, MA, AD# A052303

3. Duhl, Michael, Air Force Material Laboratory, Wright Patterson Air Force Base, Dayton, OH

TABLE III. DAMAGE TO THE RIGHT SIDE OF HELICOPTER TAIL BOOM CONSTRUCTED OF T300/5208 COMPOSITE.

Nomenclature	Material Lost lb/kg	Direction	Grid Point Displacements							
			67		70		71		72	
			in.	mm	in.	mm	in.	mm	in.	mm
0	.0	X	2.78-2	7.06-1	2.83-3	7.06-2	9.36-3	2.45-1	-1.56-2	-3.96-1
	.0	Y	4.22-1	1.07+1	4.22-1	1.07+1	4.12-1	1.05+1	4.12-1	1.05+1
		Z	-2.79-1	-7.09+0	-3.17-1	-8.05+0	-3.11-1	-7.90+0	-3.46-1	-8.79+0
		R	5.06-1	1.29+1	5.28-1	1.34+1	5.16-1	1.31+1	5.38-1	1.37+1
1	.43	X	2.77-2	7.04-1	2.28-3	5.79-2	9.17-3	2.33-1	-1.61-2	-4.09-1
	.19	Y	4.29-1	1.09+1	4.29-1	1.09+1	4.19-1	1.06+1	4.19-1	1.06+1
		Z	-2.82-1	-7.16+0	-3.21-1	-8.15+0	-3.15-1	-8.00+0	-3.50-1	-8.89+0
		R	5.13-1	1.30+1	5.35-1	1.36+1	5.24-1	1.33+1	5.46-1	1.39+1
2	.88	X	2.73-2	6.93-1	8.68-5	2.20-3	8.14-3	2.07-1	-1.87-2	-4.75-1
	.40	Y	4.59-1	1.17+1	4.59-1	1.17+1	4.49-1	1.14+1	4.49-1	1.14+1
		Z	-2.97-1	-7.54+0	-3.37-1	-8.56+0	-3.30-1	-8.38+0	-3.68-1	-9.35+0
		R	5.47-1	1.39+1	5.69-1	1.45+1	5.57-1	1.41+1	5.81-1	1.48+1
3	.88	X	2.76-2	7.01-1	1.72-3	4.37-2	9.12-3	2.32-1	-1.66-2	-4.22-1
	.40	Y	4.37-1	1.11+1	4.37-1	1.11+1	4.25-1	1.08+1	4.25-1	1.08+1
		Z	-2.87-1	-7.29+0	-3.28-1	-8.33+0	-3.19-1	-8.10+0	-3.57-1	-9.07+0
		R	5.23-1	1.33+1	5.46-1	1.39+1	5.31-1	1.35+1	5.55-1	1.41+1
4	1.40	X	2.72-2	6.91-1	-3.99-4	-1.01-2	8.37-3	2.13-1	-1.88-2	-4.78-1
	.63	Y	4.63-1	1.18+1	4.63-1	1.18+1	4.50-1	1.14+1	4.50-1	1.14+1
		Z	-3.02-1	-7.67+0	-3.46-1	-8.79+0	-3.34-1	-8.48+0	-3.75-1	-9.53+0
		R	5.53-1	1.40+1	5.78-1	1.47+1	5.60-1	1.42+1	5.86-1	1.49+1
5	1.80	X	2.71-2	6.88-1	-1.67-3	-4.24-2	7.68-3	1.98-1	-2.05-2	-5.21-1
	.82	Y	4.80-1	1.22+1	4.80-1	1.22+1	4.67-1	1.19+1	4.67-1	1.19+1
		Z	-3.14-1	-7.98+0	-3.60-1	-9.14+0	-3.48-1	-8.84+0	-3.97-1	-1.01+1
		R	5.74-1	1.46+1	6.00-1	1.52+1	5.82-1	1.48+1	6.13-1	1.56+1

Table III CONT. DAMAGE TO THE RIGHT SIDE OF HELICOPTER TAIL BOOM CONSTRUCTED OF T300/5208 COMPOSITE.

Nomen-clature	Material Lost lb/kg	Direc-tion	Grid Point Displacements							
			67		70		71		72	
			in.	mm	in.	mm	in.	mm	in.	mm
6	2.34	X	2.70-2	6.86-1	-4.58-3	-1.16-1	5.39-3	1.51-1	-2.52-2	-6.40-1
	1.06	Y	5.24-1	1.33+1	5.24-1	1.33+1	5.15-1	1.31+1	5.15-1	1.31+1
		Z	-3.46-1	-8.79+0	-3.93-1	-9.98+0	-3.83-1	-9.73+0	-4.26-1	-1.08+1
		R	6.28-1	1.60+1	6.55-1	1.61+1	6.42-1	1.63+1	6.68-1	1.70+1
7	1.91	X	2.87-2	7.29-1	-3.00-5	-7.62-4	1.00-2	2.54-1	-1.39-2	-3.53-1
	.87	Y	4.46-1	1.13+1	4.46-1	1.13+1	3.89-1	9.88+0	3.89-1	9.88+0
		Z	-3.07-1	-7.80+0	-4.13-1	-1.05+1	-3.42-1	-8.69+0	-4.32-1	-1.10+1
		R	5.41-1	1.37+1	6.08-1	1.54+1	5.18-1	1.32+1	5.81-1	1.48+1
8	2.55	X	2.85-2	7.24-1	-5.14-4	-1.32-2	9.64-3	2.45-1	-1.44-2	-3.66-1
	1.16	Y	4.48-1	1.14+1	4.48-1	1.14+1	3.90-1	9.91+0	3.90-1	9.91+0
		Z	-3.10-1	-7.87+0	-4.17-1	-1.06+1	-3.45-1	-8.76+0	-4.37-1	-1.11+1
		R	5.45-1	1.38+1	6.12-1	1.55+1	5.21-1	1.32+1	5.85-1	1.49+1

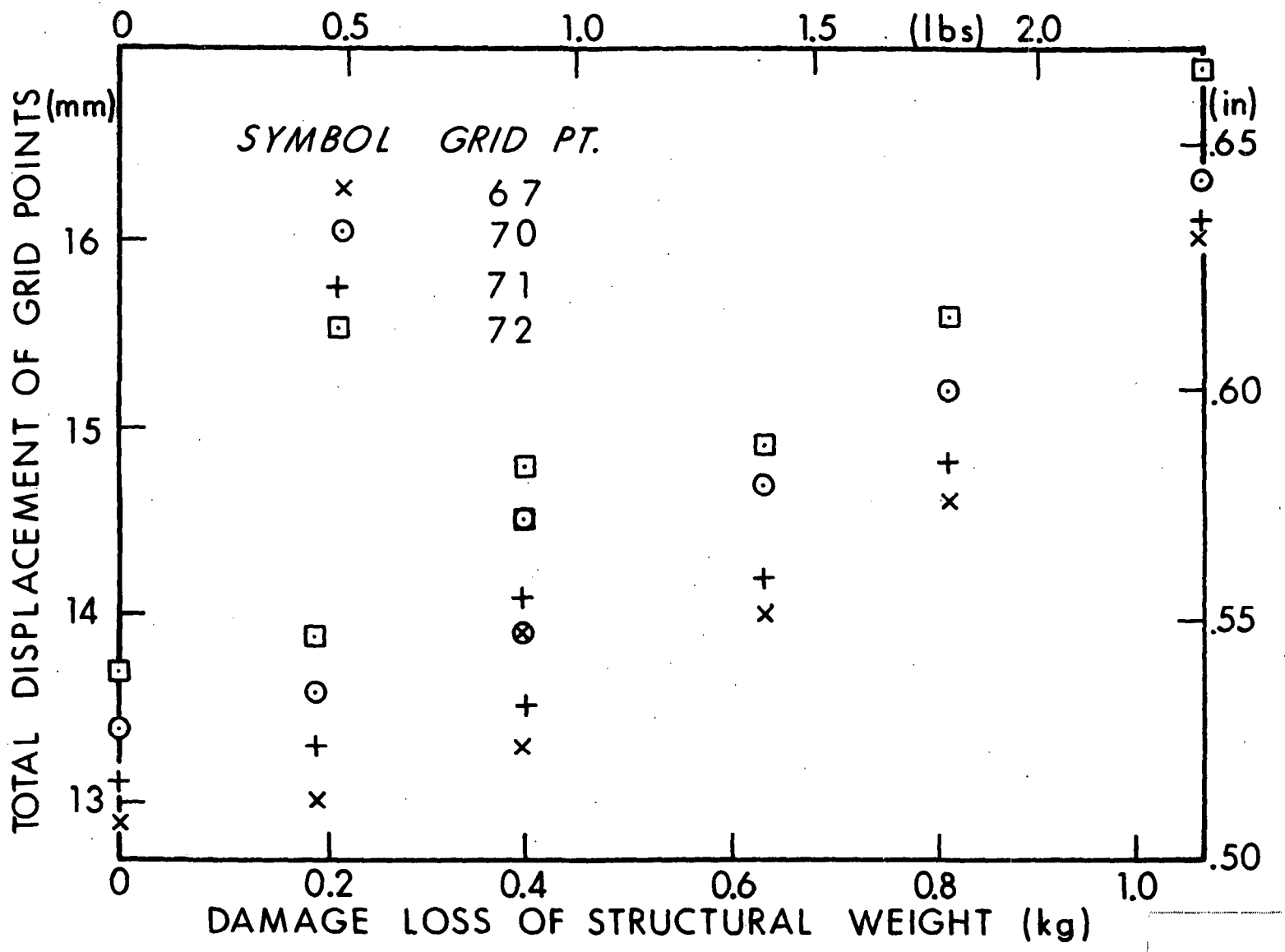


FIGURE 5. GRID POINT DISPLACEMENT VS. DAMAGE LOSS OF STRUCTURAL WEIGHT--T300/5208 COMPOSITE CONSTRUCTION

FINITE CIRCULAR PLATE ON ELASTIC FOUNDATION CENTRALLY LOADED BY RIGID SPHERICAL INDENTER

S. K. Wadhwa and P. P. Yang

IBM General Systems Division
Rochester, Minnesota

SUMMARY

This paper discusses the analytical solution of a finite circular plate on an elastic foundation centrally loaded by the rigid indenter; and the procedure to use NASTRAN as a subroutine to iteratively converge to this solution numerically.

INTRODUCTION

In contact recording applications, where a thin flexible spinning disk backed by an elastic pad is penetrated by a read/write head, it is important to understand the head penetration, the contact area, and the pressure distribution. A computer program was written to iteratively converge to the solution using MSC/NASTRAN version 60* (hitherto called NASTRAN) as a subroutine. Results were erratic. It was found that extremely fine grid was needed around the contact area to converge to the right solution.

In order to calculate the approximate contact area, the model was simplified to a finite circular plate on an elastic foundation centrally loaded by the head. The approximation is fairly valid because the flexural rigidity of the plate is extremely low and the reaction forces at the center of the spinning disk is about 2% of the total load. The effect of most of the load, thus, is to cause local deformation of the plate between the head and the pad. An analytical solution for this case was obtained. The iterative procedure to converge to this solution by using NASTRAN as a subroutine to verify the analytical solution was also defined.

* MacNeal-Schwendler Corp version of NAsa STRuctural ANalysis.

ANALYTICAL SOLUTION

Consider in figure 1 the spherical indenter with radius R_s being pressed into the circular plate with center at the origin, the plate resting on an elastic foundation with elastic spring constant k. It is of interest to determine the inside contact radius OA and the pressure distribution on the indenter over that contact. Notice, as shown in figure 1, that the outside contact radius is OB and the plate physically separates from the elastic foundation for radii larger than OB. Behavior of this "free" plate can also be modeled easily by matching boundary conditions at B, but it was not necessary in our problem. Instead we were seeking the solution to the problem "what is the outside radius R_o of the plate such that vertical displacement W=0 at R_o?"

The behavior of the plate on elastic foundation can be described (ref. 1) by equation

$$\left[\frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial \theta^2}\right]\left[\frac{\partial^2 W}{\partial r^2} + \frac{1}{r}\frac{\partial W}{\partial r} + \frac{1}{r^2}\frac{\partial^2 W}{\partial \theta^2}\right] = \frac{q-kW}{D}$$

For our case, in which the indenter presses on the center of the plate, there is a radial symmetry, and the solution is independent of angle θ, we can rewrite the above equation as:

$$\left[\frac{d^2}{dr^2} + \frac{1}{r}\frac{d}{dr}\right]\left[\frac{d^2 W}{dr^2} + \frac{1}{r}\frac{dW}{\partial r}\right] = \frac{q-kW}{D} \qquad (1)$$

Where W = vertical displacement
q = distributed lateral load
D = flexural rigidity of the plate,

$$= \frac{Eh^3}{12(1-\nu^2)}$$

h = thickness of the plate
E = Young's modules of the plate
ν = Poisson's ratio of the plate
k = spring rate/unit area of the elastic foundation.

In seeking the solution for above, we divide the plate into two sections, as shown in figure 2. The inside (section 2) portion conforms to the sphere up to contact radius R_{CI}. The outside portion of the plate must also satisfy differential equation (1) with appropriate matched boundary conditions.

OUTSIDE PORTION OF THE DISK

Consider now the outside portion of the disk (section 1 in figure 2).

If q = constant, such as gravity load, the particular solution of equation (1) is simply $\frac{q}{k}$. Therefore, we now seek the complementary solution (in absence of q) of

$$\left[\frac{d^2}{dr^2} + \frac{1}{r}\frac{d}{dr}\right]\left[\frac{d^2W}{dr^2} + \frac{1}{r}\frac{d^2W}{dr}\right] = \frac{-kW}{D} \qquad (2)$$

Let us define $\ell = \sqrt[4]{\frac{D}{k}}$

$$\xi = \frac{r}{\ell}$$

$$\eta = \frac{W R_s}{\ell^2}$$

where R_s is the radius of the indenter.

Equation 2 can then be rewritten as

$$\left[\frac{d^2}{d\xi^2} + \frac{1}{\xi}\frac{d}{d\xi}\right]\left[\frac{d^2\eta}{\partial\xi^2} + \frac{1}{\xi}\frac{\partial\eta}{\partial\xi}\right] + \eta = 0$$

This is a linear differential equation of the 4th order. The solution can be represented in terms of Bessel's function. However, it is presented below in terms of series solution for ease of computation.

The general solution is given by

$$\eta = A_1 \delta_1 + A_2 \delta_2 + A_3 \delta_3 + A_4 \delta_4 \tag{3}$$

where $A_1 \ldots A_4$ are constant,

where

$$\delta_1 = 1 - \frac{\xi^4}{2^2 4^2} + \frac{\xi^8}{2^2 4^2 6^2 8^2} \cdots\cdots \tag{4}$$

$$\delta_2 = \xi^2 - \frac{\xi^6}{4^2 6^2} + \frac{\xi^{10}}{4^2 6^2 8^2 10^2} \cdots\cdots \tag{5}$$

Coefficients in above two series can be generated by recursive relationship

$$b_n = \frac{b_{n-4}}{(n^2)(n-2)^2}$$

$$\delta_3 = \delta_1 \log \xi + \Delta_3$$

$$\delta_4 = \delta_1 \log \xi + \Delta_4$$

where

$$\Delta_3 = \frac{3}{128} \xi^4 - \frac{25}{1769472} \xi^8 \cdots\cdots \tag{6}$$

$$\Delta_4 = \frac{5}{3456} \xi^6 - \frac{1540}{442368} \xi^{10} \cdots\cdots \tag{7}$$

and coefficients in 6 and 7 can be recursively generated by

$$b_n = \frac{-1}{n^2(n-2)^2} \left[b_{n-4} + \frac{4.(n)(n-1)(n-2)(-1)^{n/4}}{2^2 4^2 6^2 \ldots n^2} \right]$$

Four constant $A_1..A_4$ can be thus evaluated by following boundary conditions.

1 At $r = R_{CI}$, $\frac{dW}{dr} = -\frac{r}{R_s}$ ... conforms to the sphere

Therefore, at $\xi = \frac{R_{CI}}{\ell}$

$$\frac{d\eta}{d\xi} = \frac{R_s}{\ell^2} \cdot \frac{dW}{d\xi}$$

$$= \frac{R_s}{\ell^2} \cdot \frac{dW}{dr} \cdot \frac{dr}{d\xi}$$

$$= -\frac{R_s}{\ell^2} \cdot \ell \cdot \frac{\ell . \xi i}{R_s}$$

$$= -\xi_i$$

2 At $r = R_{CI}$

$$\frac{d^2W}{dr^2} = -\frac{1}{R_s}$$ ... conforms to sphere

$$\frac{d^2\eta}{d\xi^2} = -1$$

3 Moment $\Big|_{r=R_o} = 0$

$$\frac{d^2W}{dr^2} + \frac{\nu}{r}\frac{dW}{dr} = 0$$

or

$$\frac{1}{R_s}\frac{d^2\eta}{d\xi^2} + \frac{\nu}{\xi_o \ell}\frac{\ell}{R_s}\cdot\frac{d\eta}{d\xi} = 0$$

$$\frac{d^2\eta}{d\xi^2} + \frac{\nu}{\xi}\cdot\frac{d\eta}{d\xi}\Big|_{\xi=\xi_o} = 0$$

4 Shear force $\Big|_{r=R_o} = 0$

$$\frac{d^3W}{dr^3} + \frac{1}{r}\frac{d^2W}{dr^2} - \frac{1}{r^2}\frac{dW}{dr}\Big|_{r=R_o} = 0$$

can be rewritten as

$$\frac{d^3\eta}{d\xi^3} + \frac{1}{\xi}\frac{d^2\eta}{d\xi^2} - \frac{1}{\xi^2}\frac{d\eta}{d\xi}\Big|_{\xi=\xi_o} = 0$$

Thus the solution of the outside portion of the disk can be obtained from equation (3).

SOLUTION OF INSIDE PORTION OF DISK

The solution for the inside portion of the disk (section 2, figure 2) that conforms to the sphere satisfies

$$\frac{d^2W}{dr^2} = -\frac{1}{R_s}$$

and, therefore,

$$W = W_i + \frac{1}{2R_s}\left(R_{CI}^2 - r^2\right) \tag{8}$$

where W_i = deflection of the disk at $r = R_{CI}$ obtained from equation (3).

PRESSURE DISTRIBUTION

Total force F is given by

$$F = \int_o^{R_o} (kW)2\pi r.dr$$

where $W = W_i + \frac{1}{2R_s}\left(R_{CI}^2 - r^2\right) \qquad o<r<R_{CI}$

$$= \frac{\ell}{R_s}^2 \left[A_1\delta_1 + A_2\delta_2 + A_3\delta_3 + A_4\delta_4\right] \qquad R_{CI}<r<R_o$$

Pressure distribution under the head is given by equation (8) multiplied by k. However, at radius R_{CI} there is additional shear force required to keep the outside portion of the disk in equilibrium, and is given by

$$Q = D\left[\frac{d^3W}{dr^3} + \frac{1}{r}\frac{d^2W}{dr^2} - \frac{1}{r^2}\frac{dW}{dr}\right]_{r=R_{CI}}$$

NORMALIZING

Solution to the equations are functions of R_s, F, ℓ, R_o. However, if the disk is very weak and outside radius R_o is taken to be that value for which W=0, then from equations (3 through 8), it can be shown that $F R_s$, R_o/ℓ , $Q R_s \ell$, $q R_s \ell^2$ are all normalized functions of R_{CI}/ℓ.

Figures 3 and 4 show some of the relationships.

NASTRAN MODEL DESCRIPTION

The application of the static analysis of MSC/NASTRAN to the problem in figure 1 is demonstrated in this section.

The elastic foundation in figure 1 is first discretized into the "gap scalar spring" of finite length corresponding to grid points on the plate. The "gap scalar spring" is defined as a linear spring that can be compressed only. An extension of this spring will produce no spring force. This spring force will then be used to generate the FORCE card for the NASTRAN program. Secondly, an isoparametric bending element with transverse shear is used to model the plate between the indenter and the elastic foundation. As shown in figure 5, to save computer time, only a section of this circular plate was modeled in this work. A symmetric condition is used along the edge boundaries. Since the spherical indenter can be treated as a rigid body in this problem, its contour is computed and stored in [C]. This [C] is used as an enforced displacement to the plate model within the contact area.

The radius R_{CI} of the contact area between the indenter and the plate is essentially the solution we need. In the following part, an iterative technique was developed to compute the R_{CI}, contact radius, and the load distribution over the plate.

Final solution to the problem must meet the following conditions:

1 There should be no geometric interference between the deformed plate and the contour of the indenter. The interference example is as shown in figure 6, where the plate deformation pattern $[W]_1$ interferes with the indenter contour [C]. The smallest R_{CI} without a geometric interference is the solution to this problem.

2 The summation of the distributed load, or the "gap scalar spring" force, over the plate be equal to the total force F.

$$\Sigma\ [q]\ [\text{area}] = F \qquad (9)$$

In order to satisfy the above two conditions, an iterative type of algorithm was developed to solve the problem by utilizing the CALL NASTRAN technique. In this case, the whole NASTRAN program becomes a subroutine that can be implemented by any user-developed program. The NASTRAN program can thus be invoked by using the CALL statement in the user's program just like using any other conventional subroutines.

To start this iteration process, an initial guess of R_{CI} is required as an input to the program. With this assumed R_{CI}, the static contact area between the indenter and the plate can be defined and set equal to the indenter contour by using the enforced displacement card, SPC. Initial plate deflections are obtained by using NASTRAN. With these deflections and the elastic foundation stiffness k, forces on the grid can be calculated satisfying condition (2) above. This [q], as the FORCE cards, along with the previously defined SPC cards, are then put into the NASTRAN program to compute the plate displacement [W]. This plate displacement is not supposed to interfere with the indenter contour [C]. If this geometric constraint is not satisfied, a larger R_{CI} will be assumed next and repeat the above steps until a R_{CI} can be found that satisfies this geometric constraint. This final R_{CI} is the solution we are trying to obtain. Thus, the static contact area and the final distributed load can be computed accordingly without any difficulty.

Figure 7 shows the simplified flow diagram of this iteration process. Some additional explanations and programming considerations are made below.

1 Since the NASTRAN program is being called and invoked many times, NASTRAN CHECKPOINT and RESTART features have been used.

2 The CALL NASTRAN statement actually includes the following functions:

   a Generate the NASTRAN executive control deck, including CHECKPOINT dictionary for RESTART use, Case Control Decks and the Bulk Data Card.

   b Invoke the NASTRAN program by using a CALL statement. The NASTRAN program is invoked throughout the entry point of its load module. This entry point name is "NASTRAN" in our case.

c Recover the displacement vector computed by NASTRAN as the output from this CALL NASTRAN operation. The module OUTPUT4 was used to modify the DMAP sequence to get the desired output from NASTRAN.

RESULTS

The iteration scheme defined in the NASTRAN model gave results quite close to those obtained analytically. The graphs are shown in figures 3 and 4. Given force F and the radius R_s of the indenter, contact radius and the pressure distribution can be obtained from figures 3 and 4 respectively.

REFERENCE

1 Timoshenko, S., Woinowsky-Krieger, S., "Theory of Plates and Shells", M'Graw Hill Bork Company, New York.

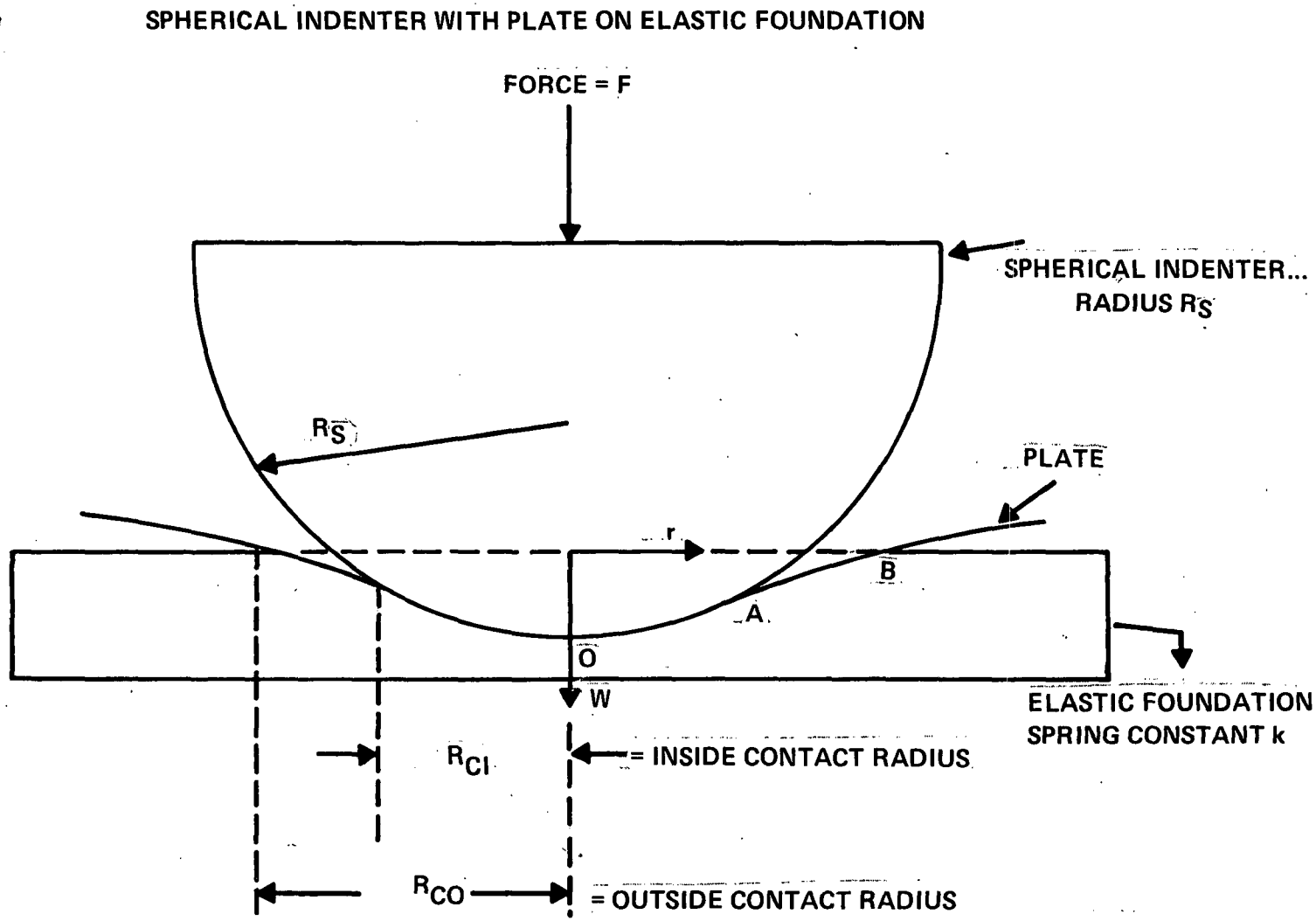


Figure 1. Spherical indenter with plate on elastic foundation

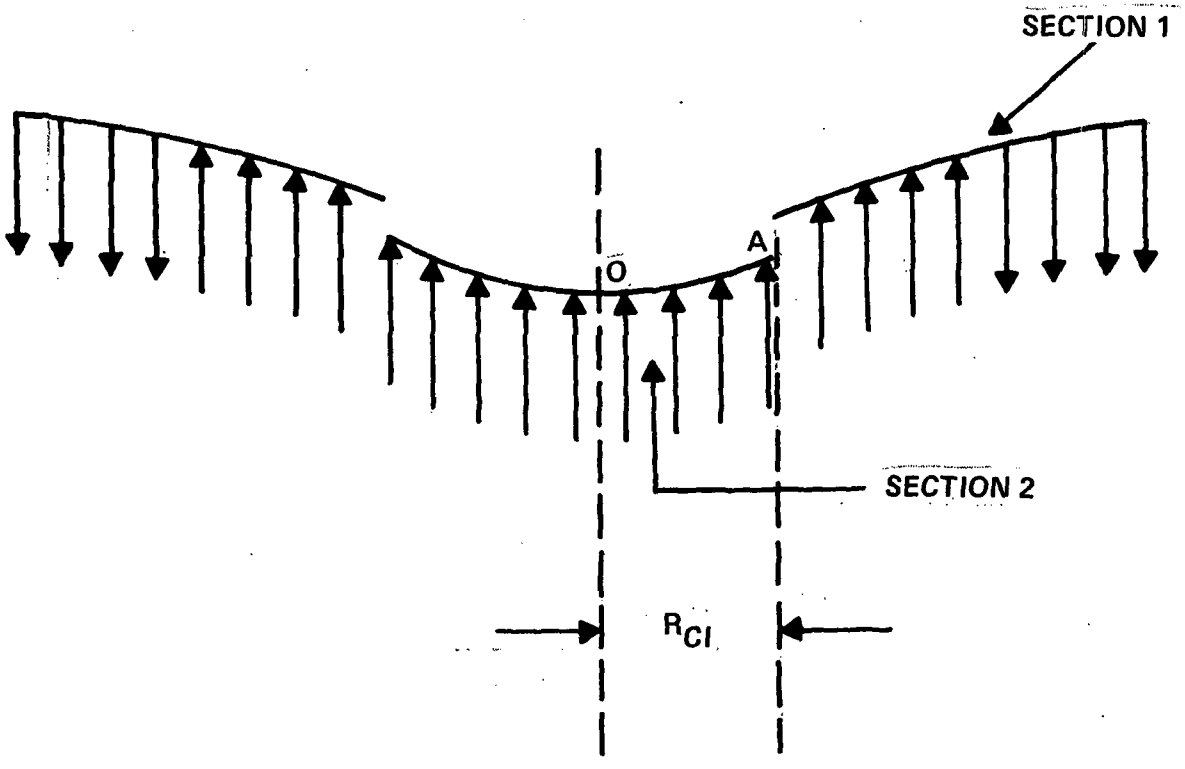


Figure 2. Plate shown in two sections. Inside portion, O-A, conforms to the sphere with radius R_s.

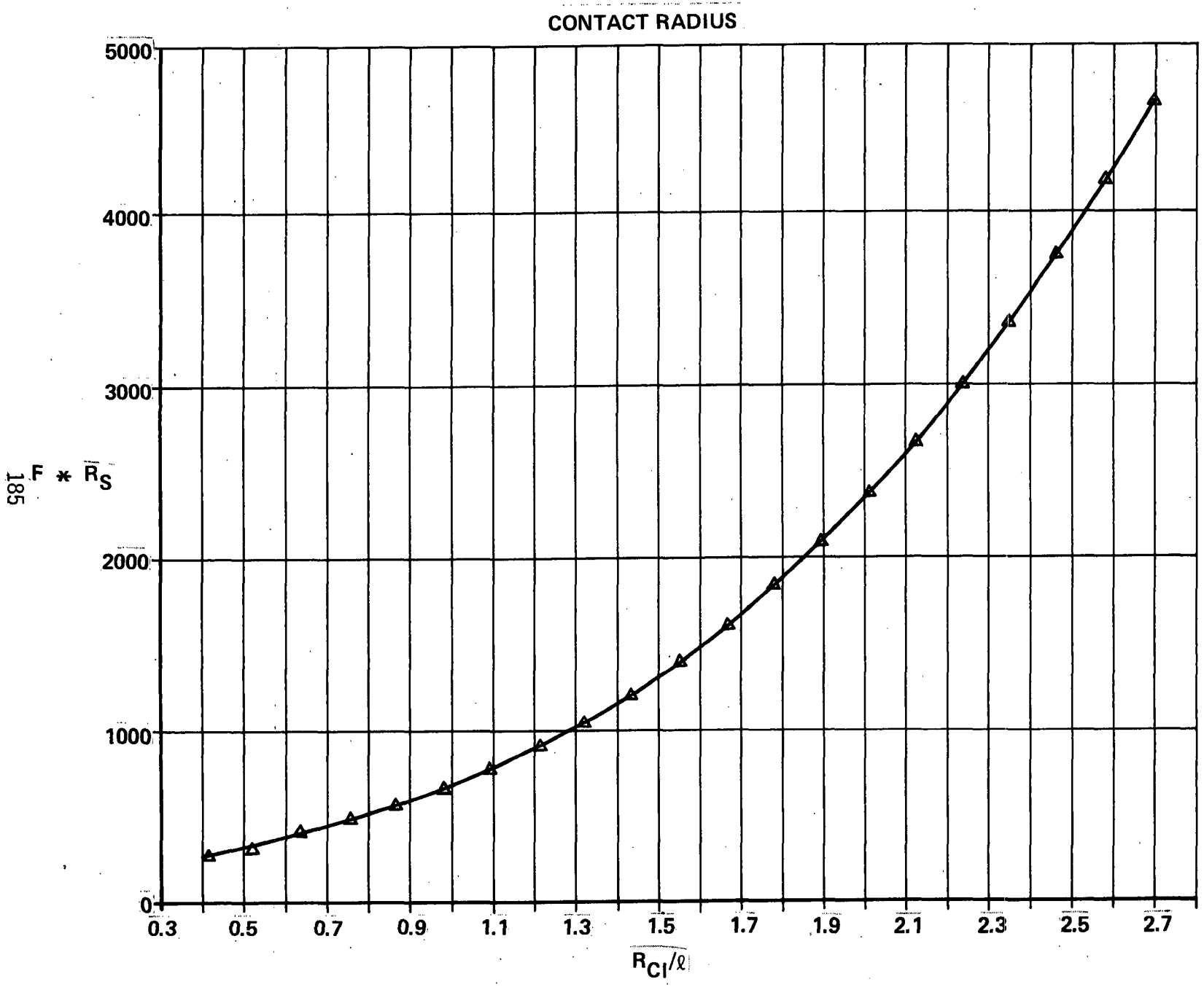


Figure 3. Contact radius

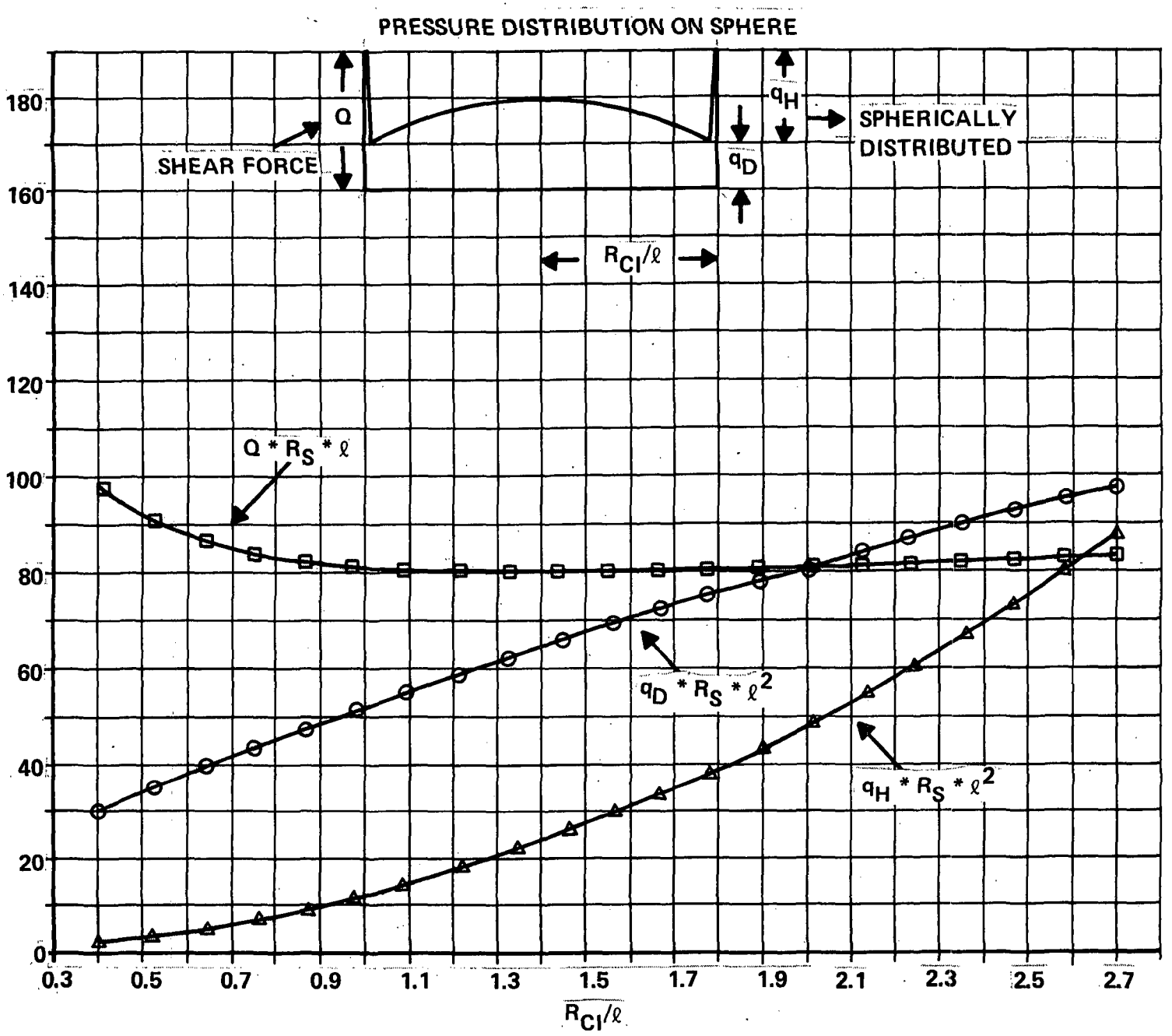


Figure 4. Pressure distribution on sphere

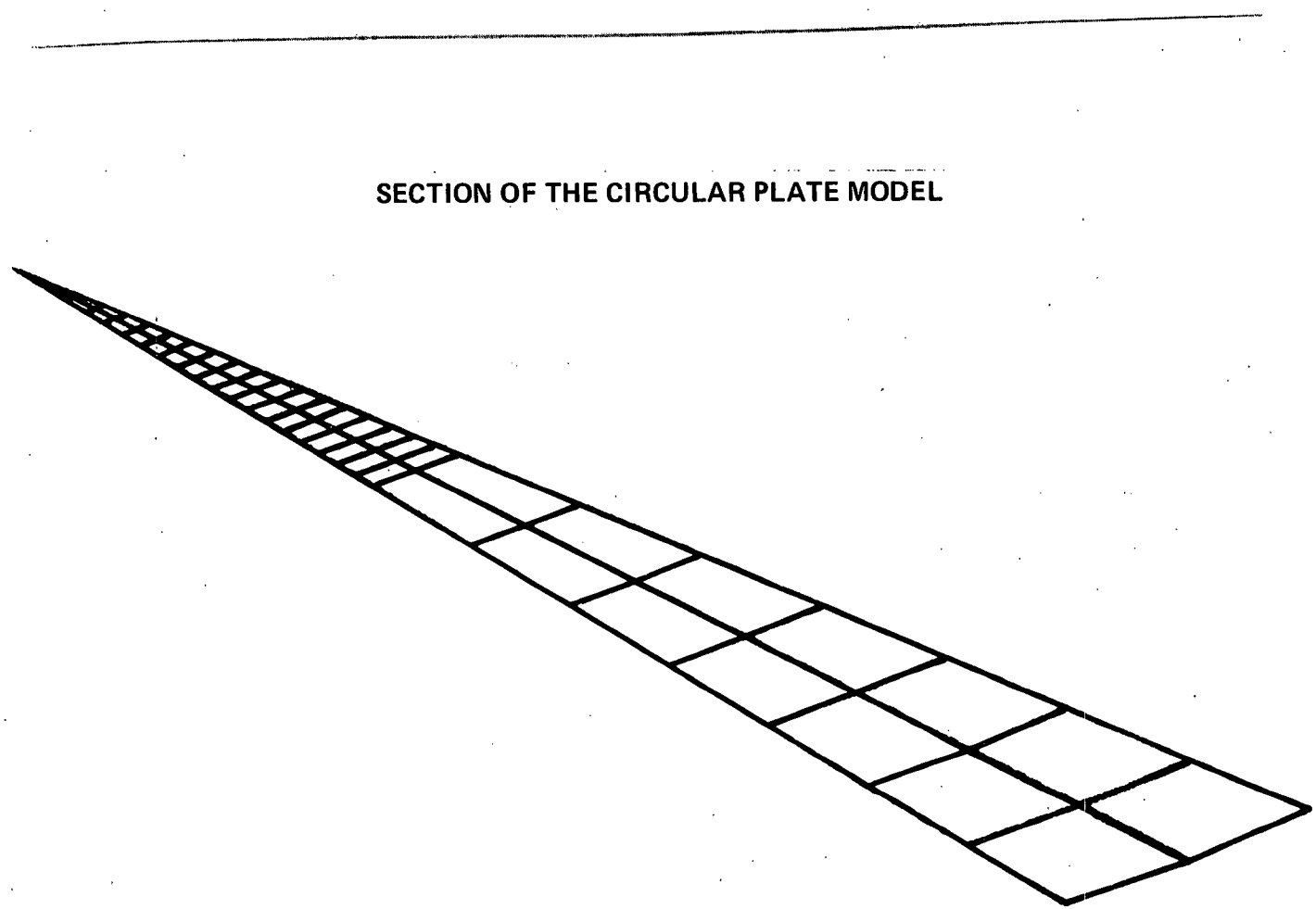


Figure 5. A section of the circular plate model

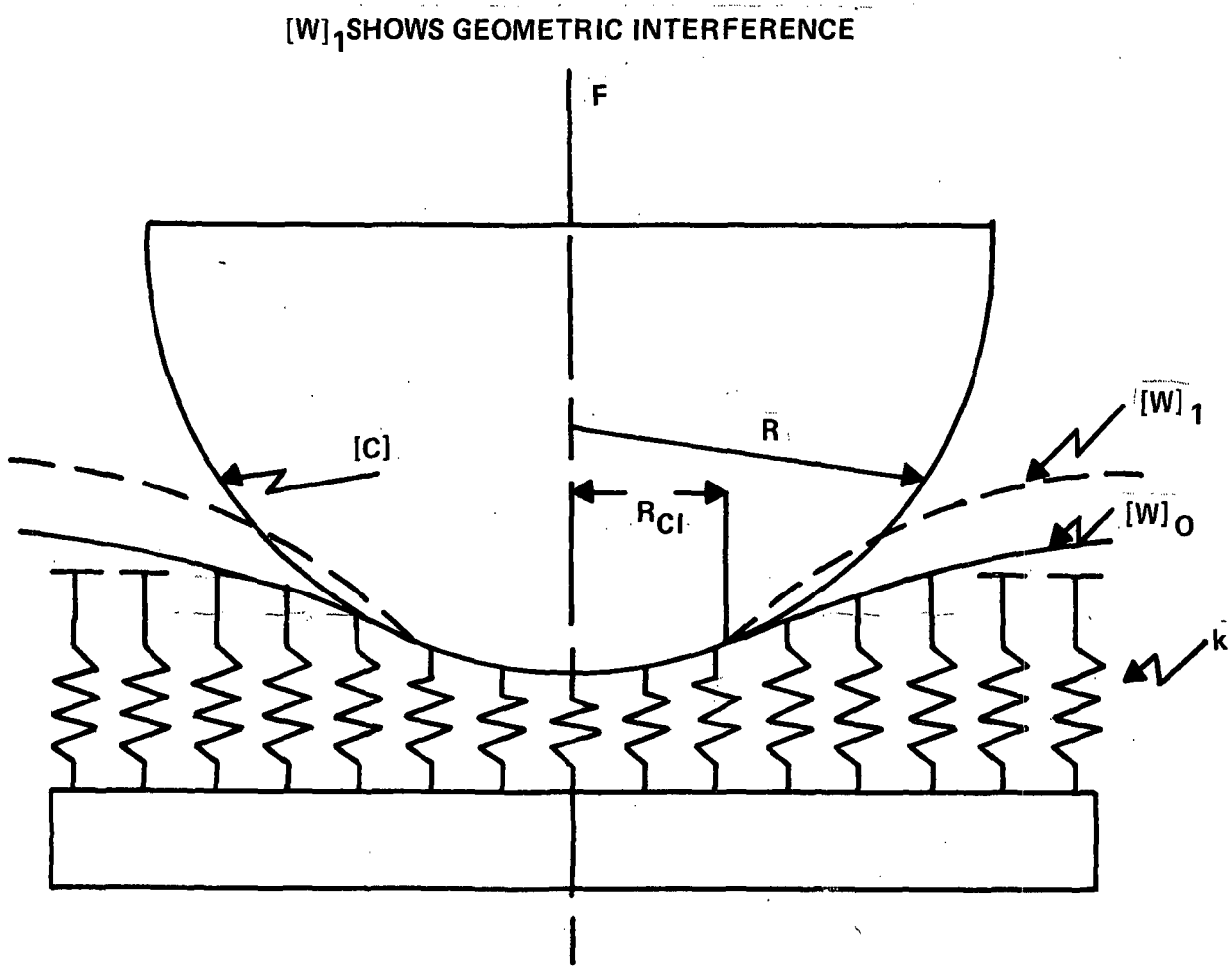


Figure 6. $[W]_1$ shows geometric interference

SIMPLIFIED FLOW DIAGRAM OF THE ITERATION PROCESS

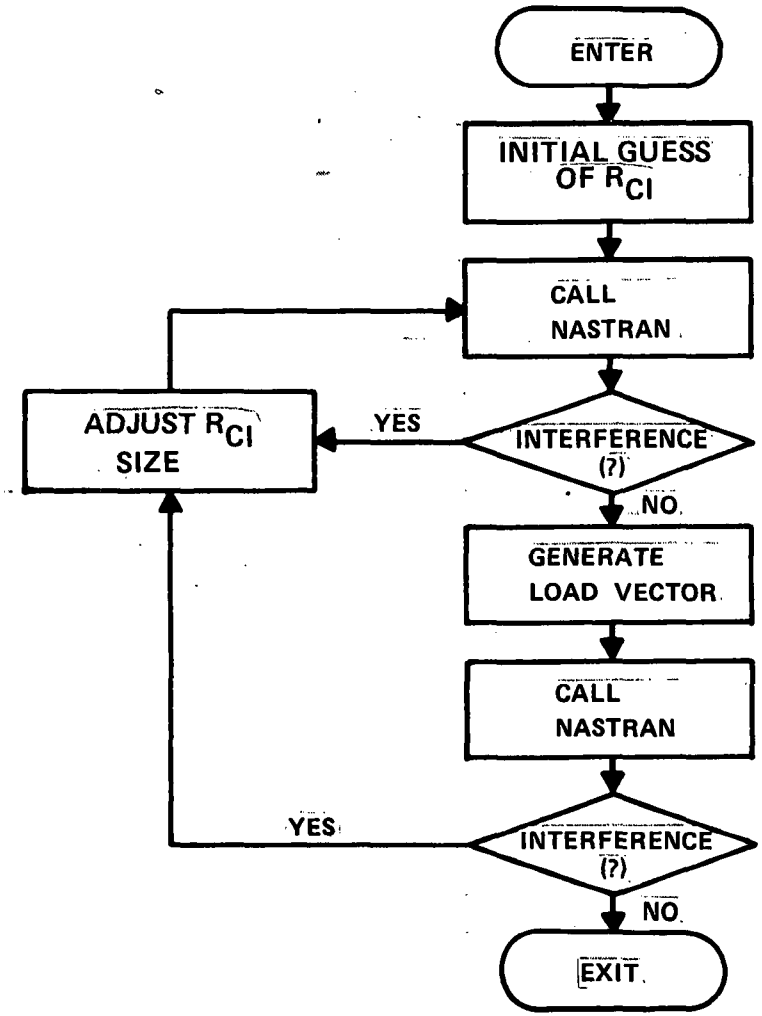


Figure 7. Simplified flow diagram of the iteration process

ELASTIC-PLASTIC ANALYSIS USING A TRIANGULAR RING ELEMENT IN NASTRAN

P. C. T. Chen
U.S. Army Armament Research and Development Command
Benet Weapons Laboratory, LCWSL
Watervliet, NY 12189

SUMMARY

An elastic-plastic triangular ring element is implemented in NASTRAN computer program. The plane-strain problem of partially-plastic thick-walled cylinder under internal pressure is solved and compared with the earlier finite-difference solution. A very good agreement has been reached. In order to demonstrate its application to more general problems, an overloaded thread problem for the British Standard Buttress is examined. The maximum axial and principal stresses are located and their values are determined as functions of loadings.

INTRODUCTION

The piecewise linear analysis option of the NASTRAN program provides an algorithm for solving nonlinear problems in material plasticity (ref. 1). However, the usefulness of this option is quite limited because only a few elements have been implemented. These include rod, tube, bar elements for one-dimensional problems and plate elements for two-dimensional plane stress problems. In a recent paper (ref. 2), the implementation of a trapezoidal ring element in NASTRAN for elastic-plastic analysis was described and two test problems of rotational symmetry were solved. The first is an infinitely long tube under uniform internal pressure. The NASTRAN results are in excellent agreement with an exact solution based on the finite-difference approach (ref. 3). The second problem is a thick-walled cylinder of finite length loaded over part of its inner surface. The NASTRAN results are in good agreement with another finite-element solution (ref. 4).

In the present paper, two elastic-plastic problems of rotational symmetry are solved using triangular ring elements for the finite element models. The implementation of a triangular ring element in NASTRAN for elastic-plastic analysis follows the same procedures in reference 2 for a trapezoidal ring element. In order to test the accuracy of the implementation, the plane-strain problem of a partially-plastic thick-walled cylinder under internal pressure is solved again and compared with the finite difference solution (ref. 3). The second demonstrative problem in using NASTRAN triangular ring elements is an overloaded thread problem for the British Standard Buttress. A detailed discussion for this problem in the elastic range of loading was reported in reference 5. In the present paper,

uniform pressure distribution on the primary bearing surface of the thread is assumed and the load is applied in increments beyond the elastic limit. The maximum axial and principal stresses are located and their values are determined as functions of loadings.

TRIANGULAR RING ELEMENTS

The incremental displacement field employed for the triangular ring element are

$$\Delta u(r,z) = \beta_1 + \beta_2 r + \beta_3 Z \ , \tag{1}$$

$$\Delta w(r,z) = \beta_4 + \beta_5 r + \beta_6 Z \ . \tag{2}$$

The transformation from grid point coordinates to generalized coordinates is

$$\{\beta\} = [\Gamma_{\beta q}] \ \{\Delta q\} \tag{3}$$

where

$$\{q\}^T = \llcorner \Delta u_1, \ \Delta w_1, \ \Delta u_2, \ \Delta w_2, \ \Delta u_3, \ \Delta w_3 \lrcorner , \tag{4}$$

$$\{\beta\}^T = \llcorner \beta_1, \ \beta_2, \ \beta_3, \ \beta_4, \ \beta_5, \ \beta_6 \lrcorner , \tag{5}$$

and the elements of the inverse of the transformation matrix $[\Gamma_{\beta q}]^{-1}$ are the coefficients of the β's in equations (1) and (2), evaluated at the corners of the element.

The stiffness matrix is formed in the same manner as that for the anisotropic elastic element. The final form referred to grid coordinates is

$$[K] = [\Gamma_{\beta q}]^T \ [\bar{K}] \ [\Gamma_{\beta q}] \ , \tag{6}$$

where

$$[\bar{K}] = 2\pi \int r [B]^T \ [D] \ [B] \ dzdr \ . \tag{7}$$

The $[B]$ matrix is the same as the elastic case, but now it expresses the incremental strains in terms of generalized coordinates

$$\{\Delta\varepsilon\} = [B] \ \{\beta\} \ . \tag{8}$$

The $[D]$ matrix which relates the incremental stresses to the incremental strains, i.e.,

$$\{\Delta\sigma\} = [D] \ \{\Delta\varepsilon\} \ , \tag{9}$$

is the same as that for a trapezoidal ring element presented in reference 2. In developing NASTRAN program for strain-hardening materials, we calculate

$[D]^{-1}$ and obtain its inverse $[D]$ numerically. For ideally plastic materials, this procedure fails and we should calculate $[D]$ directly using the closed form.

NASTRAN IMPLEMENTATION

The implementation of a triangular ring element in NASTRAN for elastic-plastic analysis follows the same procedures for a trapezoidal ring element (ref. 2). Changes were required in the functional modules PLA1, PLA3, and PLA4, which included the writing of seven new subroutines. The changes in PLA1 allows this module to identify the new element as a member of the piecewise linear element set and properly initialize the nonlinear Element Summary and Element Connection Property Tables. Three element stress recovery subroutines were added to PLA3: PSIARG, a driver for stress data recovery; PSRIR1 ans PSRIR2, phase I and II stress recovery routines. Element stiffness calculations in PLA4 require four new subroutines: PKIARG, a driver for nonlinear triangular ring elements in PLA4, PKRIR1 and PKRIR2, stress recovery routines which generate stresses for the computation of the nonlinear material matrix; and PKRIRS, the stiffness matrix generation routine for nonlinear triangular ring elements.

PROGRAM EVALUATION

In order to evaluate the accuracy of the computed code, the plane-strain problem of an elastic-plastic thick-walled cylinder under uniform internal pressure is solved again and compared with the finite-difference solution (ref. 3). The tube of outside radius 2" and inside radius 1" has been divided into 10 equal intervals and each interval consists of 2 triangular ring elements. The material constants are $E = 30 \times 10^6$ psi, $\nu = 0.3$, $\sigma_o = 1.5 \times 10^5$ psi and the effective stress-strain curve is represented by three line segments connecting the four points in the $(\bar{\varepsilon},\bar{\sigma})$ plane, $(\bar{\varepsilon},\bar{\sigma}/\sigma_o) = (0.0,0.0)$, $(0.005,1.0)$, $(0.055,1.5)$, $(0.1,1.5)$. Twenty-three load factors are chosen: $P/\sigma_o = 0.4323$, 0.4738, 0.5125, 0.5484, 0.5818, 0.6128, 0.6415, 0.6681, 0.6925, 0.7150, 0.7356, 0.7545, 0.7716, 0.7871, 0.8011, 0.8135, 0.8245, 0.8341, 0.8423, 0.8493, 0.8550, 0.863, 0.87. The numerical results based on the NASTRAN program have been obtained. For this problem, exact solution based on a new finite-difference approach (ref. 3) can be used to assess the accuracy of the NASTRAN code. Some of the results for the displacements and stresses are presented graphically in Figures 1 and 2. The radial displacements at the inside as well as outside surface are shown in Figure 1 as functions of internal pressure. Figure 2 shows the distributions of radial, tangential and axial stress components in a partially-plastic tube when the pressure is $p = 0.7356\ \sigma_o$. As demonstrated in Figures 1 and 2, the NASTRAN results are in very good agreement with those based on the finite-difference approach (ref. 3).

THREAD PROBLEM

As an application of using NASTRAN triangular ring elements for two-dimensional problems, an overloaded thread problem for the British Standard Buttress will be examined. A finite element model is shown in Figure 3. A detailed discussion for this problem in the elastic range of loading was reported in reference 5. In the present paper, uniform pressure distribution (P) on the primary bearing surface of the thread is assumed and the load is applied in ten unequal increments beyond the elastic limit. The elastic constants are E = 25000 Ksi, ν = 0.25, and the effective stress-strain curve is represented by five line segments connecting the six points in the $(\bar{\varepsilon}-\bar{\sigma})$ plane, $(\bar{\varepsilon}-\bar{\sigma}$ in Ksi) = (0.0, 0.0), (0.0048,120), (0.0122,180), (0.0167,200), (0.0918,210), (0.25,210). The sides A-B and C-D are constrained in the axial direction and the side B-C, in the axial and radial directions.

The elastic problem is solved first and the upper limit of the elastic loading (p*) is found to be 54.65 Ksi. Eleven load factors are chosen as follows: p = 54.65, 60.59, 66.22, 71.54, 76.54, 81.23, 85.61, 89.68, 93.43, 96.87, 100.0 Ksi. The numerical results based on the NASTRAN code have been obtained and the total CPU time is 59 minutes on IBM 360 model 44. The plastic zone at the maximum load is indicated by the shaded area in Figure 3. Some of the stress results are shown graphically in Figures 4 to 6. Of particular interest is the region along side D-E-F. The axial stress (σ_z) and major principal stress (σ_1) in the boundary elements along side D-E-F are shown in Figures 4 and 5, for the first and last load factor, respectively. The maximum fillet stress (σ_1) occurs near E while the maximum axial stress always occurs at D. Finally, the maximum fillet stress and axial stress are plotted as functions of contact pressure in Figure 6.

CONCLUSION

An elastic-plastic triangular ring element has been implemented in NASTRAN computer program. Its accuracy has been evaluated by solving a simpler problem for which exact solution is available. Its application to more general problem has also been demonstrated by solving an overloaded thread problem.

REFERENCES

1. "NASTRAN Theoretical Manual," NASA SP-221(01), 1972.

2. Chen, P. C. T. and O'Hara, G. P., "Implementation of a Trapeziodal Ring Element in NASTRAN for Elastic-Plastic Analysis," NASA CP-2131, Proceedings of the Eighth NASTRAN User's Colloquium, pp. 101-112, October 1979.

3. Chen, P. C. T., "A Finite-Difference Approach to Axisymmetric Plane-Strain Problems Beyond the Elastic Limit," Transactions of the Twenty-Fifth Conference of Army Mathematicians, pp. 455-465, January 1980.

4. Chen, P. C. T., "Numerical Solution of Gun Tube Problems in the Elastic-Plastic Range," Proceedings of the 1977 Army Numerical Analysis and Computer Conference, pp. 423-439, November 1977.

5. O'Hara, G. P., "Stress Concentrations in Screw Threads," NASA CP-2131, Proceedings of the Eighth NASTRAN Users' Colloquium, pp. 65-77, October 1979.

6. "NASTRAN Programmer's Manual," NASA SP-223(01), 1972.

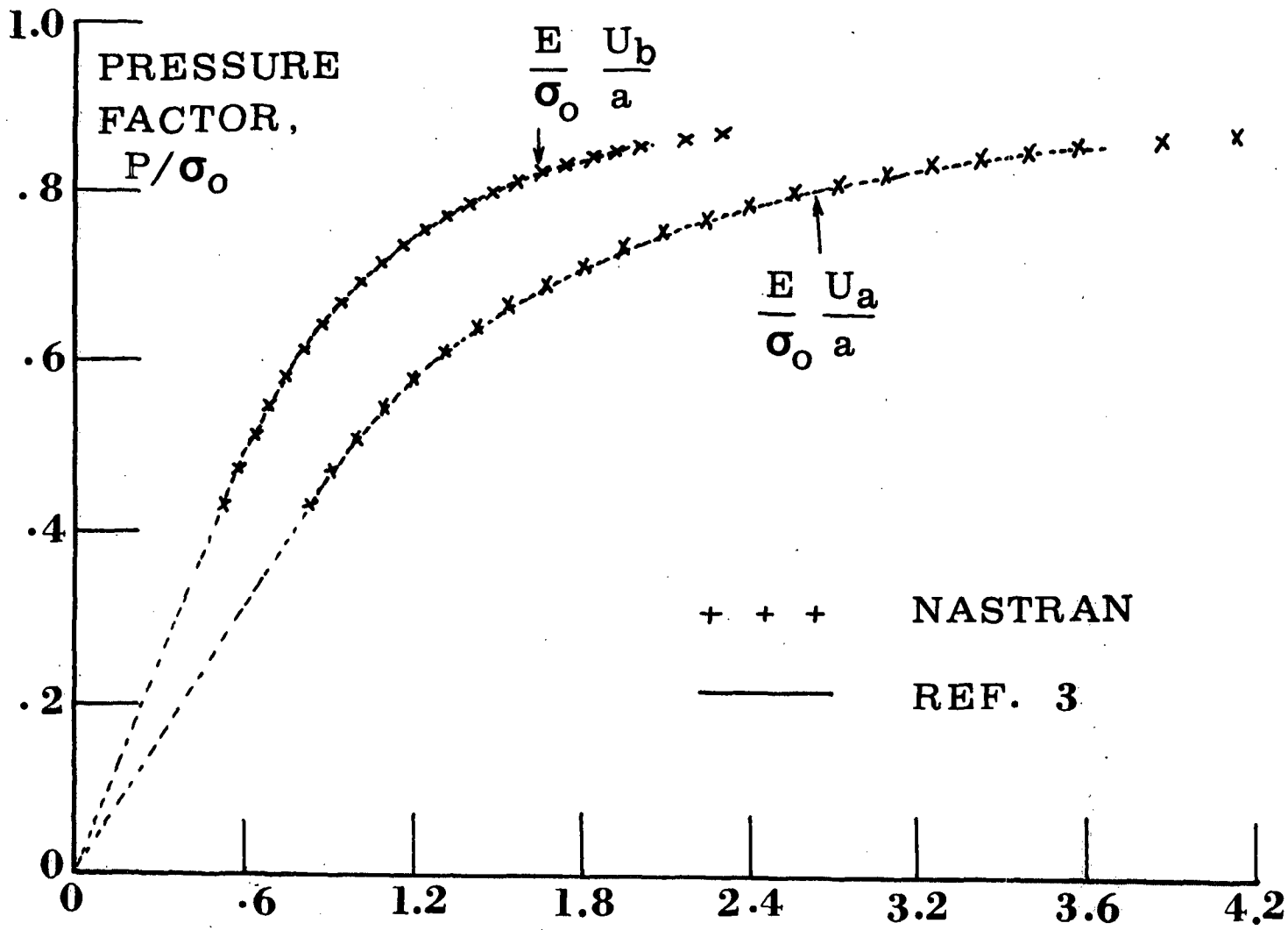


Figure 1. Radial displacement in a pressurized tube.

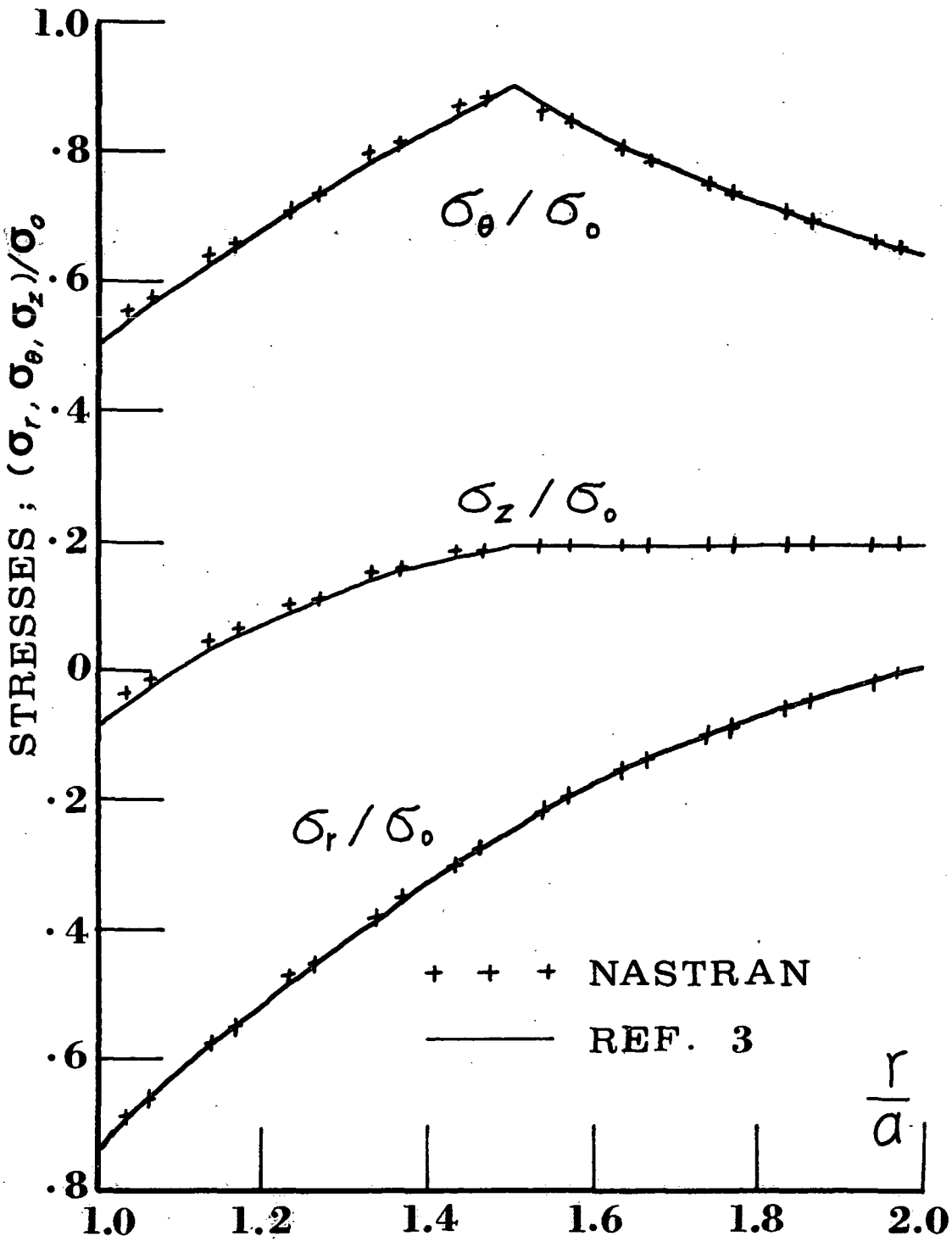


Figure 2. Stresses in a pressurized tube.

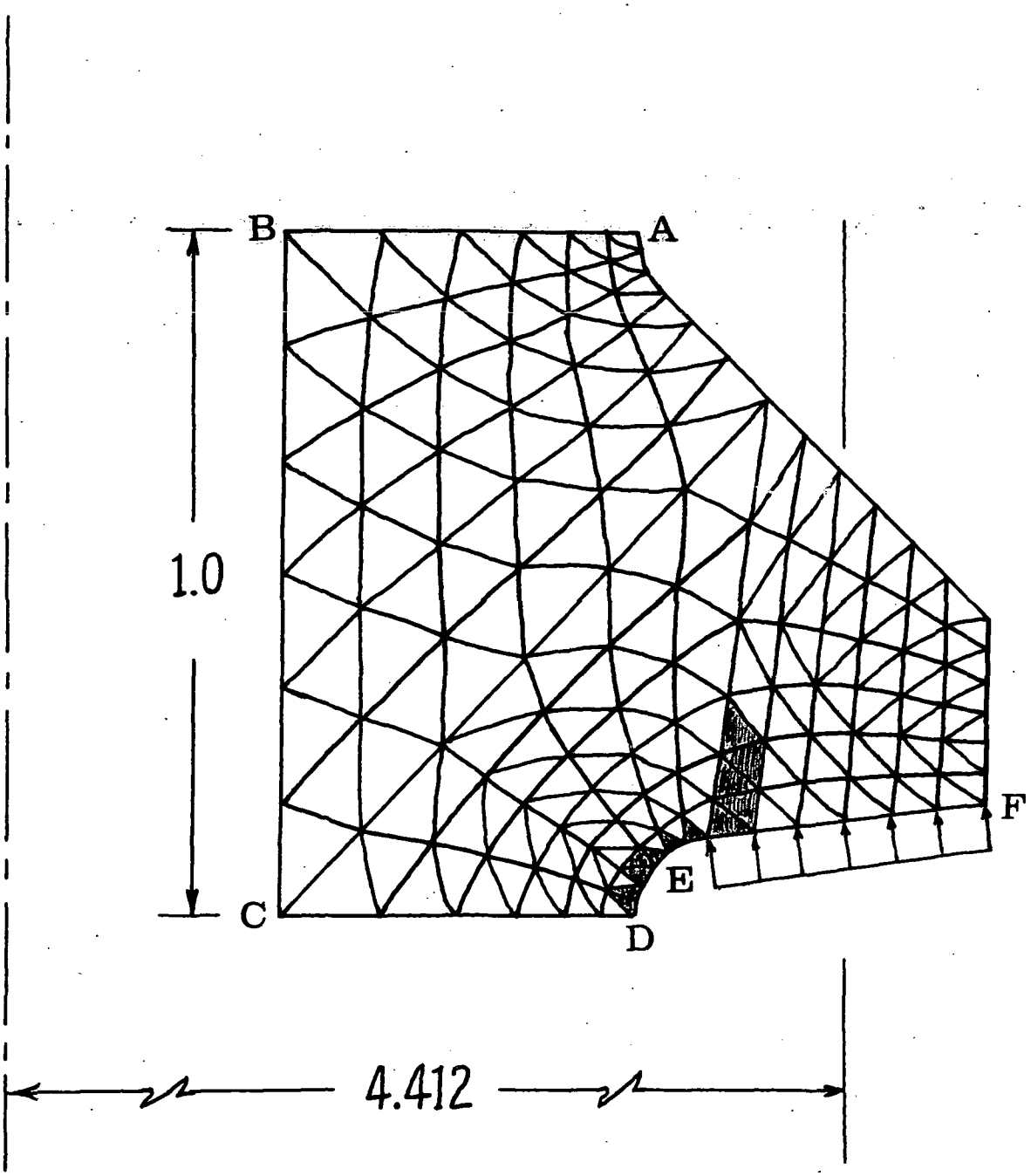


Figure 3. Finite element model for the thread problem.

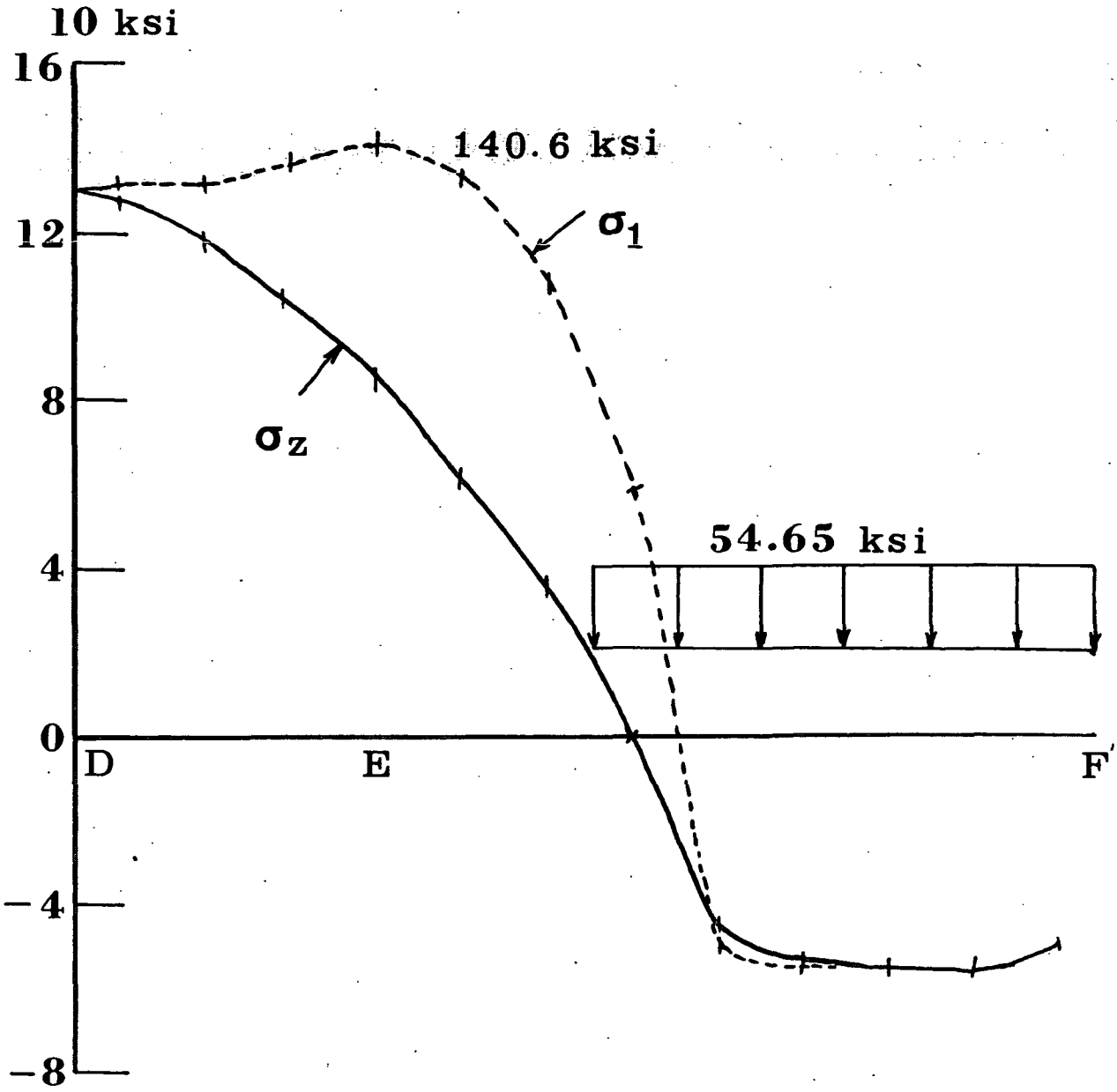


Figure 4. Stresses in boundary elements (side D-E-F) at p = 54.65 Ksi.

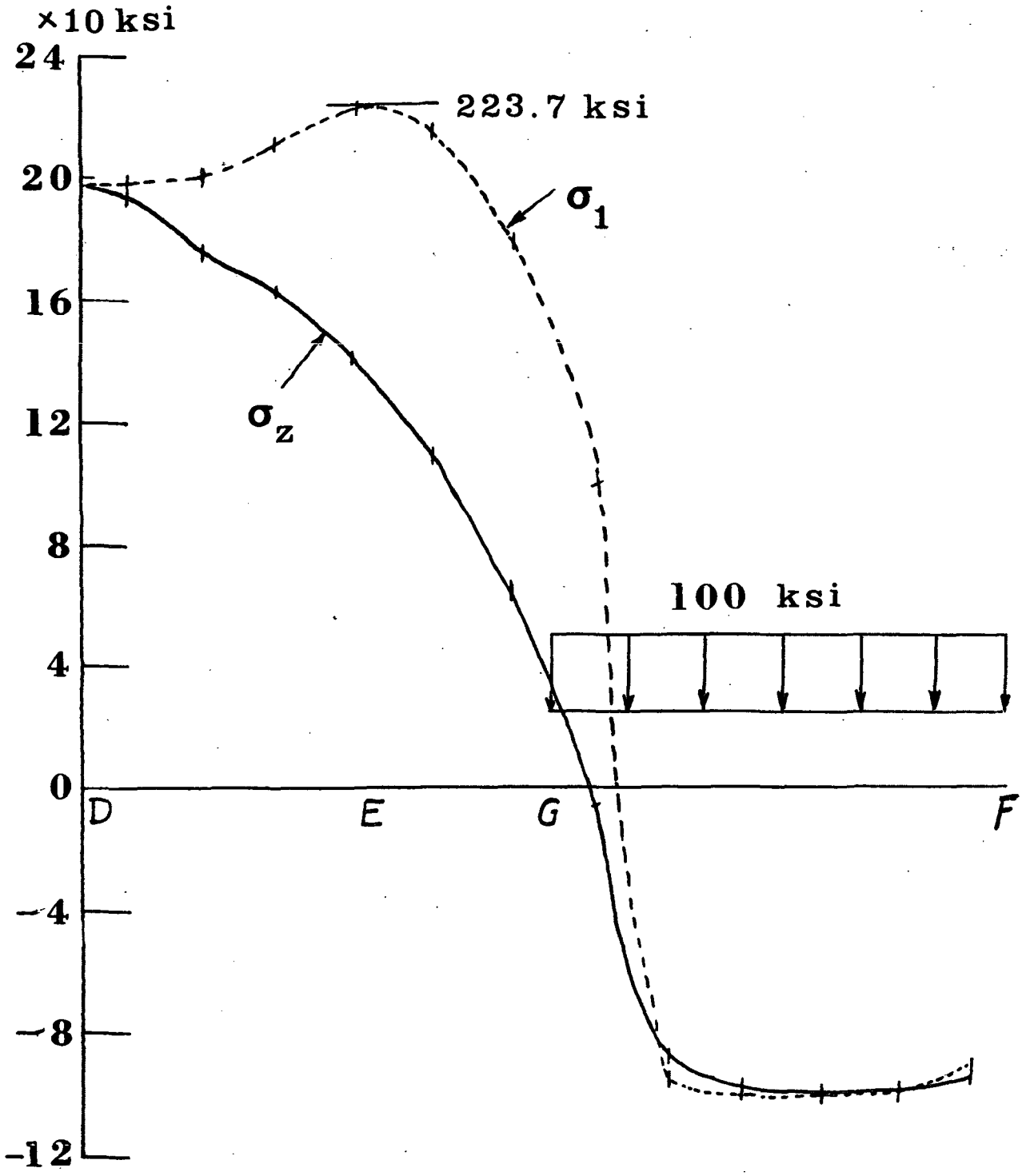


Figure 5. Stresses in boundary elements (side D-E-F) at p = 100 Ksi.

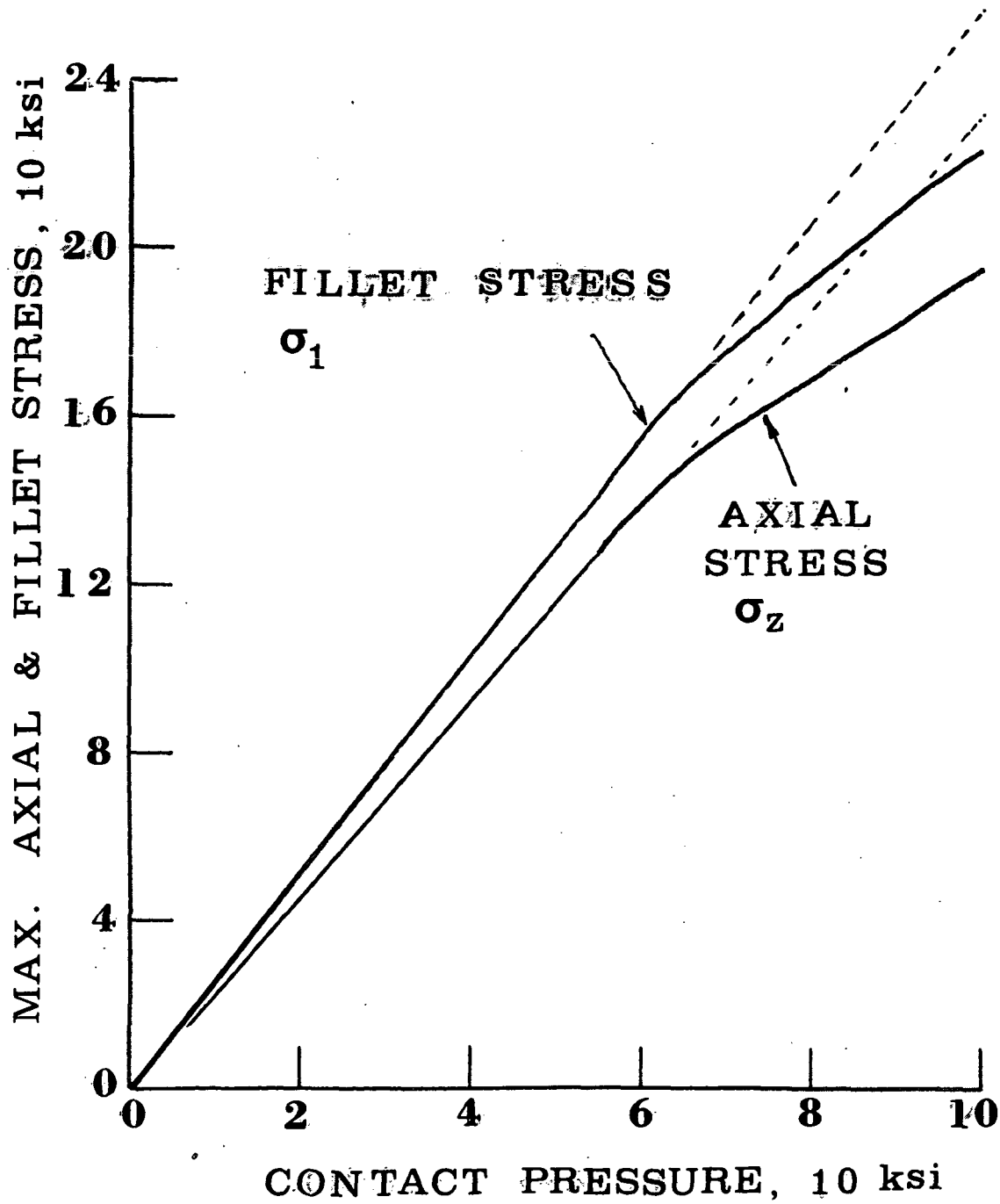


Figure 6. Maximum axial and fillet stresses as functions of contact pressure.

DEVELOPMENT AND ANALYSIS OF THE LEARJET 54/55 FUSELAGE NASTRAN MODEL USING SUBSTRUCTURE TECHNIQUES

BY

ROBERT R. BOROUGHS, SIVAM PARAMASIVAM, AND JOANNA WERNER

SUMMARY

Development and analysis of the Learjet 54/55 fuselage Nastran model presented a real challenge considering the size of the task and the resources that were available at the beginning of this project. Consequently, this structure was broken down into several substructures to make the modeling and analysis effort more manageable. Since the geometry was fairly complex in some areas and in order to provide flexibility for future configuration studies, a series of local coordinate systems were used to describe the model. This work was accomplished using several different computer systems, the more recent of which were connected by a high speed data communications lines to perform the tasks of model generation and Nastran analysis.

INTRODUCTION

The Learjet Model 50 series aircraft has become the latest and largest member of the Learjet product line (see Figure 1). This airplane was an almost completely different aircraft from earlier Learjet models. The wing has been extended six feet from the original Learjet wing configuration, and the tip tanks have been removed and replaced by winglets. A paper describing the Nastran finite element analysis of this wing which was also used on the 28/29 airplanes can be found in another NASA document (Ref. 1). Discussion here will be primarily directed toward the Nastran analysis of the fuselage and vertical fin.

The fuselage on the Model 54/55 aircraft has been increased in diameter as well as length over previous models. Construction in the fuselage was the typical skin stringer arrangement with frames located at given increments to provide ridity to this shell structure. The windshield was similar to other Learjet windshields with the large unsegmented stretched acrylic panels, except that these panels were bigger and two small side windows were installed on the aft perimeter of the windshield. Attachment of the wing to the fuselage was accomplished through eight fittings, four on each side of the fuselage, which provided fail safe capability in this area. The fuselage structure was cut out at the wing-fuselage juncture to permit wing continuity through the fuselage. A keel beam was installed beneath the wing to provide a more uniform internal load path in this region. This keel beam structure was also continuous forward and aft of the wing attachment.

Pressure bulkhead construction consisted of either sheet metal webs reinforced with stringers and support beams or honeycomb panels reinforced with structural beams. The aircraft was powered by two Garrett AiResearch TFE 731-3 turbofan engines mounted on each side of the aft fuselage section. The engine support structure consisted of a forward and aft box beam which were also integral to two partial bulkheads. Configuration of the vertical tail included five spars covered by aluminum sheet and stiffened by formed sheet metal ribs.

BACKGROUND

Initial studies on the 50 series aircraft started in 1976. Nastran was used as preliminary design tool to study various structural arrangements and determine which configuration was more advantageous. As these investigations matured and more parameters became defined, a transition from a conceptual finite element fuselage model to a final configuration model was necessary. The final configuration fuselage model proposed was to have a fine enough mesh to accurately define the internal loads distribution throughout the structure for all load cases, yet not be overly complex so that the lead times and costs would be excessive. With these guidelines in mind a model of approximately 20,000 degrees of freedom was recommended. Development of this size model still presented real challenges considering the computer resources that were available at the beginning of this effort. There was a good deal of concern about the logistics of model generation, management of the resulting data base, and the method of analysis. The computer system initially considered to carry out these functions was an in-house IBM 370-145, and the method of analysis proposed was static substructuring. If the Nastran analysis were to become too large for the IBM 370-145 to handle effectively, an alternate course of action was planned where the Nastran model would be run on a Cyber computer at a service bureau.

SUBSTRUCTURE DEFINITION

Based on previous runs on the IBM 370-145 computer, a maximum substructure size of 3500 degrees of freedom was established. Each fuselage node point was considered to have the capability of transferring all 6 degrees of freedom. Grid points on the outer surface were to be located at the intersection of the frame and stringer members, and grid points in the fuselage interior were to be defined by the intersection of primary structural beams and stiffeners. These constraints and the basic characteristics of the structure resulted in dividing the Nastran model into seven substructures (see Figure 2). Before the modeling ever began a node and element numbering system was established for the entire structure. Element and grid point numbers were assigned to each substructure in a progressive manner so that one substructure could later be easily merged or subdivided with adjacent substructures. Node numbers were assigned in a sequence that attempted to minimize the matrix bandwith. Element numbering was keyed to the grid point numbering so that both the grid points

and connection members had a similar numbering sequence.

After studying the pros and cons of various coordinate systems, a geometric definition using a series of local cylinderical coordinate systems in conjunction with the basic rectangular coordinate system was selected. In this system the grid points at each frame location on the outer surface were described by a local cylinderical coordinate system unique to that frame. Grid points interior to the fuselage model were defined in a rectangular coordinate system. There were several advantages in using this type of approach. First of all, this made the modeling of some of the more complex geometry much easier, and also simplified the determination of offsets to be used with the BAR elements which were to define the frame members. Consideration was also made for the incorporation of a fuselage stretch and other possible modifications that would impact future modeling and analysis. Thus, the use of a local cylinderical coordinate system for each frame would permit a fuselage stretch by inserting a fuselage plug and redefining the affected coordinate cards with little or no changes to existing grid or element cards.

COMPUTERS USED FOR MODEL GENERATION

Several in-house computers were used in different phases of the model generation. The first phase of the modeling was the generation of the connectivity cards. Simple routines were written for CONROD and CQDMEM2 elements (Ref. 2) which given the bay ID numbers generated each connection element. These routines were originally run on an IBM 1130 but all these programs were eventually converted to run on an IBM 370-145 or on a PDP 11/70. A similar situation occurred with some frames which were constant in section around the circumference. A program was used that generated all four cards necessary for each BAR element, complete with connectivity, cross-section area, X and Y moments of inertia, and stress recovery coefficients, but with all offsets equal to zero. These values had to be added manually, but even after adding these values by hand, much time was saved using these routines.

The second phase of model generation was the section property calculation. This was done with two different routines. The first routine used ran on the IBM 370-145 and was primarily designed to calculate the section properties for a channel section made of bent-up sheet metal. This routine could also be used to calculate the properties for an angle section. A second routine that was used calculated the properties for any extruded section used such as channels, I-sections, T-sections, and angles. Special provisions for a general section built up of rectangles and fillets were also incorporated in this program. This routine ran interactively on the PDP 11/70 in-house computer.

In some cases where a frame section was comprised of two or more bent up sheet sections which were fastened together, the combined section properties had to be computed. Using the appropriate section property program to find the properties of each component, these properties were then input to a transformation program which converted all the properties with respect to the composite centroid and combined these values. This program was set up to run

on the in-house IBM 370-145.

FORWARD SECTION

The forward section began at frame 1 and extended to frame 16 just behind the cabin door (see Figure 2). This portion of the model was divided into two substructures which were identified as the nose substructure and crew substructure. The nose substructure extended from frame 1 to frame 6 (see Figure 3) and the crew substructure extended from frame 6 through frame 16 (see Figure 4). The components of the forward fuselage consisted of circumferential frames oriented in a fuselage station reference system, stringers located approximately normal to the frames, and the skins which covered this framework. Along the bottom centerline of the fuselage was the keelbeam which runs almost the complete length of the fuselage to give additional longitudinal stiffness and to provide a more continuous load path around the wing. Other components of the forward section included a forward pressure bulkhead, a windshield, and a cabin door.

The grid points defining the outside contour for these substructures as well as the other substructures were located at the frame stringer intersections. These grid points were defined in a local cylinderical coordinate system coincident with each frame. The local coordinate systems were defined with reference to a basic rectangular system where the X axis runs longitudinally aft, the Y axis left hand outboard, and the Z axis down. The local systems were oriented with R radially outward from the centerline, θ counterclockwise from the basic Y coordinate, and positive Z oriented forward. The grid points were numbered with even values on the left hand side and odd values on the right hand side. These numbers were keyed off the local system identification numbers to give unique ID's for each grid point. Grid points in the fuselage interior utilized a similar technique.

Stringer members in the forward section were modeled using CONRODS, since these elements had small cross-sectional areas that acted primarily in axial load transfer. In a few cases a torsional stiffness value was included to allow some torsional loading. The CONROD IDs directly keyed off the grid point ID's. The frames were modeled as BARS (Ref. 2) to carry axial and bending loads. The BAR orientation was defined by a $\bar{v}$ vector at the element origin pointing radially outward. QDMEM2 elements were used to model the skin, since these panels have small thicknesses and generally do not carry significant local bending loads. The keelbeam was modeled using CONRODS along the four corners with SHEAR elements (Ref. 2) on vertical sides. The bottom was coincident with the outer skin and consequently was already defined by QDMEM2 panels.

Stretched acrylic material was used for windshield and cabin windows. These members were generally quite thick and were modeled using QUAD2 elements (Ref. 2) which had bending capability. The forward end of the pressure vessel was located at frame six in the form of a pressure bulkhead. This bulkhead was comprised of a thin web which was supported by vertical stiffeners on the forward side and horizontal stiffeners on the aft side. The thin web was modeled with

QDMEM2 elements, and the stiffeners were modeled with BAR elements.

The cabin door structure consisted of a framework of intersecting frames and stringers with skin and doublers on both the inner and outer sides of this framework. A double set of grid points was used to define the perimeter of the door. The first set of grid points defined the door frame or cutout in the fuselage while the second set of grid points defined the edges of the door. Once again the frames were modeled with BARS, the stringers were modeled with CONRODS or BARS depending on the depth of the member, and the skin panels and doublers were modeled using QDMEM2 elements. The entire door was connected to the fuselage by the use of RIGID (Ref. 2) elements which transferred the appropriate degrees of freedom.

MID SECTION

The fuselage mid section began at frame 16 and continued through frame 31 just aft of the wing trailing edge (see Figure 2). This section was divided into two substructures at frame 24 which was just in front of the wing leading edge. The substructure between frames 16 and 24 was identified as the cabin substructure (see Figure 5), and the substructure between frames 24 and 31 was called the center substructure (see Figure 6). The shell structure for these sections consisted of frame-stringer-skin type construction and contained the keel beam, the escape/baggage door, passenger windows, the frame 24 partial bulkhead, a pressurized baggage floor, the aft pressure bulkhead, and a portion of the fuselage fuel cell bay.

The grid point locations and modeling of the fuselage shell structure for the mid section was accomplished in basically the same manner as described in forward section modeling discussion. The keel beam which runs almost the entire length of the fuselage was an open box type structure forward of frame 24, but aft of frame 24 and the wing leading edge region the keel beam has become a completely closed box. Modeling of the keel beam in this region was accomplished using two dimensional Nastran elements. CONROD members were used to model both the upper caps, lower caps, and vertical stiffeners. SHEARS were used to model the vertical webs while QDMEM2 elements were used to model the horizontal webs.

Window cutouts were reinforced with doublers as well as frame members in certain locations. The doubler panels were modeled using QDMEM2 elements and were connected to the same grid points as the outer skin panels. The windows were made of the same stretched acrylic material as the windshield but only thinner, and these panels were also modeled with QDMEM2 elements.

Between frames 22 and 25 and stringers 6 through 15 on the right side of the fuselage was located the escape/baggage door. A double set of grid points was used to define the perimeter of the escape/baggage door similar to the modeling performed on the main cabin door. Apart from the two close out frames at each end of the door, there were also two inner frames which were adjacent to the window in the door. The door cutout in the fuselage was

reinforced with doublers and additional frame members and intercostals. The door frame and intercostal members were modeled using BAR elements, and the door skin was modeled using QDMEM2 elements. Connection of the escape/baggage door to the fuselage was achieved by using RIGID (Ref. 2) elements between the appropriate degrees of freedom at the door hinge and pinned attachments.

The partial bulkhead at frame 24 served as a close out for the wing cutout in the fuselage. This structure consisted of a thin web supported by horizontal and vertical stiffeners. QDMEM2 elements were used to model the web, and BAR elements were used to model the stiffeners. The top edge of this bulkhead provided the forward support point for the pressurized baggage floor which was made of sandwiched honeycomb plate. Due to the thickness and bending characteristics of the honeycomb, this structure was represented in the Nastran model by QUAD1 elements (Ref. 2). The baggage floor was attached on both sides to a longeron which ran the length of the wing cutout in the fuselage. These longerons also helped to close out the wing cutout and provided a redistribution path for the internal loads. Since the longeron was a large member, BAR elements were used to similate this structure in the model instead of CONRODS.

Attachment of the wing to the fuselage was accomplished through four fittings on each side of the fuselage in the wing cutout region. The frames located through this section were all double frames. This was done to provide increased stiffness and an adequate load path for the wing reactions. These double frames were all modeled with a single ring of BAR elements to conserve degrees of freedom. Since the wing structure was not simulated in this model, the model was constrained at each of these wing attachment points. The frame 28 structure also provided the support point for the aft pressure bulkhead. This bulkhead also served as the forward retainer for the fuselage fuel cell, and was constructed of sandwiched honeycomb plate. QUAD1 elements were used to model this bulkhead as was previously done for the pressurized baggage floor.

AFT SECTION

The aft fuselage section began just behind the trailing edge of the wing and was divided into two substructures (see Figure 2). These substructures were referred to as the fuel cell (see Figure 7) and tailcone substructures (see Figure 8) and had a mutual boundary at frame 39 which was between the two aft baggage doors. Some of the basic assemblies of the fuel cell section included the fuel cell support structure and aft fuel cell bulkhead, the engine support structure, and the aft keel beam structure. Major features in the tailcone substructure consisted of the aft baggage compartment and the vertical tail support structure. Frames in both substructures were generally oriented in a vertical position with the exception of the frames that attached to the vertical fin which were oriented parallel to the spars in the vertical fin.

Nastran elements used to model the skin, stringer, and frame elements were the same as those used in the other substructures. There was one full

bulkhead in the aft section and several partial bulkheads. The full bulkhead served as the rear boundary of the fuselage fuel bay and was located just in the front of the forward engine beam. This bulkhead was constructed of flat plate reinforced by structural beams. QUAD2 elements (Ref. 2) were used to model the plate structure, and BAR elements were used to model the structural reinforcement beams. One partial bulkhead was located at the aft end of the wing fuselage cutout at frame 31 and served as a close out as well as a fuel retainer for the fuselage fuel bay. Construction of this partial bulkhead was also a flat panel with structural support beams, but the reinforcement beams were generally smaller and there were more of these members than in the aft fuel bulkhead. The webs for this partial bulkhead were modeled with QUAD2 elements, and the support beams were modeled with BARS.

Since grid points in the bulkhead mesh did not always match the grid points on the fuselage outside contour in an even manner, the interface of cross support beams on the bulkheads with the fuselage frames was not always easy to simulate in the Nastran model. MPC equations were used initially to relate the displacement of the grid points in the bulkhead which were adjacent to the outside fuselage contour to the displacement of the grid points on the outside contour. The results of the first Nastran debugging runs with these MPC equations revealed the reactions did not satisfy equilibrium conditions. However, when the MPC equations were removed from the model, then equilibrium was satisfied. This problem was reported to COSMIC, and a sample Nastran run with the MPC equations and faulty equilibrium conditions was sent to the COSMIC staff. After reviewing the Nastran run, COSMIC indicated that the problem appeared to be related to writing an MPC equation for grid points which were in different coordinate systems whose degrees of freedom were not on parallel reference axes. The coordinate system used for the grid points on the fuselage outside contour was a local cylinderical coordinate system with the origin at the maximum breadth line of the fuselage while the coordinate system used for the grid points on the fuselage interior was a rectangular coordinate system. Consequently, MPC equations were discarded for this application and tailored BAR elements were used instead.

Two of the other partial bulkheads were located at the forward and aft engine beam supports. These engine beams were box beams that were continuous through the fuselage from right to left. Both beams were curved so that the outboard ends, or engine attach points, were higher than that portion of the engine beam on the aircraft center line. A double frame was also installed at each of these locations to provide greater support for the engine loads. In order to simplify the modeling and conserve degrees of freedom, the combined section properties of the double frame were calculated and one ring of BAR elements was used to model both frames. The partial bulkheads in this region extended from the engine beam on the bottom to the double frame on the sides and on the top. BAR elements were used to model the engine beams, and QUAD2 members were used to model the web panels in the partial bulkhead. There were only a few reinforcement members in the partial bulkhead, and these elements were modeled using BAR elements.

The intersection of the engine beams and the stringer members on the fuselage contour were such that the grid points for these elements did not

coincide at the true intersection point. Since there was internal load transfer capability of all six degrees of freedom at this point, a method of relating this interaction was necessary. RIGID elements (Ref. 2) were used to connect these degrees of freedom, and the results appeared to be satisfactory.

VERTICAL FIN

The seventh substructure in the fuselage model was the vertical fin installation which also included a model of the rudder (See Figure 9). A model of the horizontal stabilizer was constructed, but was not included in this data base in order to minimize the degrees of freedom and cut down on computer run costs. Horizontal stabilizer loads were applied to the vertical fin at the attach points where the horizontal stabilizer and vertical fin joined. The vertical fin consisted of five spars and eight ribs covered with aluminum sheet while the rudder contained one spar and eleven ribs with an aluminum covering. The vertical fin was joined to the tailcone at each of the five spars as well as through attachments on the skin. Spar connections were made at frames 43, 45, 46, 47 and 48 which were canted to accommodate this interface.

A separate local rectangular coordinate system was adopted for the grid locations for both the vertical fin and rudder to facilitate the modeling. The spar caps in the region of the fuselage frames were modeled using BAR elements while the remainder of the spar caps and rib caps were modeled using CONROD elements. Skin panels were modeled using the QDMEM2 member, and SHEAR elements were used to represent the spar and rib webs. Attachment of the rudder to the vertical fin was accomplished through two hinges which were modeled using CONRODS. The torque load in the rudder was restrained by a torque tube attached to the bottom of the rudder which in turn attached to fuselage frame 48 and was modeled using BAR elements. The spar caps and rib caps were again modeled using CONROD elements while the skin was modeled using QDMEM2 panels. CSHEAR elements were used to model the rudder leading edge ribs while TRMEM elements were used for the ribs between the hinge line and trailing edge of the rudder.

GRAPHICS SYSTEMS

Plots for each of the completed substructures were obtained using an in-house plot package developed for Nastran on an in-house PDP 11/70 mini computer. Since this program was originally written for plots with grid points in the basic coordinate system, an expansion to the software was made to incorporate multiple local coordinate systems. The user could access this routine through a Tektronix CRT, and the substructure would be displayed in a three-dimensional view on the screen. Different views could be selected by specifying the appropriate rotational angles about each of the three principal axes. Hard copy plots of very fine resolution then could be obtained by spooling the desired view to a Versatec plotter. This approach allowed the user to correct any noticeable geometry or connectivity errors before going

to the initial Nastran finite element analysis.

LOADS

Several different load conditions and combinations of load conditions were applied to the model. The basic types of loads used during this analysis were pressure loads, down bending loads, up bending loads, and side bending loads. An ultimate and limit internal pressure was applied separately as well as in conjunction with the ultimate down bending and ultimate up bending load cases. Side bending loads were applied to the vertical fin and the rudder.

COMPUTERS USED FOR ANALYSIS

Further error correction was achieved by running the substructures with the Nastran software on the IBM 370-145. However, after the Nastran debugging process was completed, the first few substructures ran so long that even over night turn around became a problem. This difficulty appeared to be significantly influenced by increased usage of the computer not only by structures personnel but also by other departments. One of the alternatives considered as a solution for this situation was to break the model into smaller substructures and increase the number of substructures from seven to eleven substructures. A second alternative was to run the model on a more powerful outside computer. At that point in time the second of the two alternatives seemed to be more satisfactory in order to meet schedule requirements. Several different outside sources were examined, and the selection process finally narrowed down to Control Data's Cybernet System running Nastran on a Cyber 175 computer.

The first complete computer runs for the 54/55 Nastran fuselage analysis were made successfully on the Cybernet System using Nastran Superelement analysis. In order to cut down on the transmission of data over the phone lines, a data base was set up for the fuselage model at the computer site. This data base permitted the running of several different load cases at reduced cost by transmitting only the JCL and loads data between the Learjet terminal and the Cyber 175.

EXPERIMENTAL RESULTS

Only a portion of the static test program had been completed, and the correlation of experimental data with analytical results had been under way a short time when this paper had to be submitted to the publisher. Consequently, just a few figures were available showing the comparison between the Nastran results and the static test data. A sample of two load conditions have been shown on the following pages. These two load conditions were the vertical fin side bending load case and the limit pressure loading. Plots for the side

bending load case have been shown in figures 10 and 11, and the correlation between the Nastran values and experimental data for the limit pressure load case have been shown in figure 12.

CONCLUDING REMARKS

Modeling of the Learjet 54/55 fuselage was based upon use of a multiple local coordinate system which proved to have significant advantages and worked very satisfactorily. The fuselage was analyzed using Superelement analysis where the structure was divided into seven superelements. This approach made the analysis of this size structure much more manageable and easier to perform.

REFERENCES

1. Boroughs, Robert R.: Development of The Learjet 28/29 Wing Using Nastran Analysis, pp. 11-32, EIGHTH NASTRAN USER'S COLLOQUIUM, NASA CP 2131, OCTOBER 1979.

2. THE NASTRAN USER'S MANUAL (Level 17.0), NASA SP-222 (04), Washington, D.C., December, 1977.

FIGURE 1 – LEARJET 54/55 AIRCRAFT

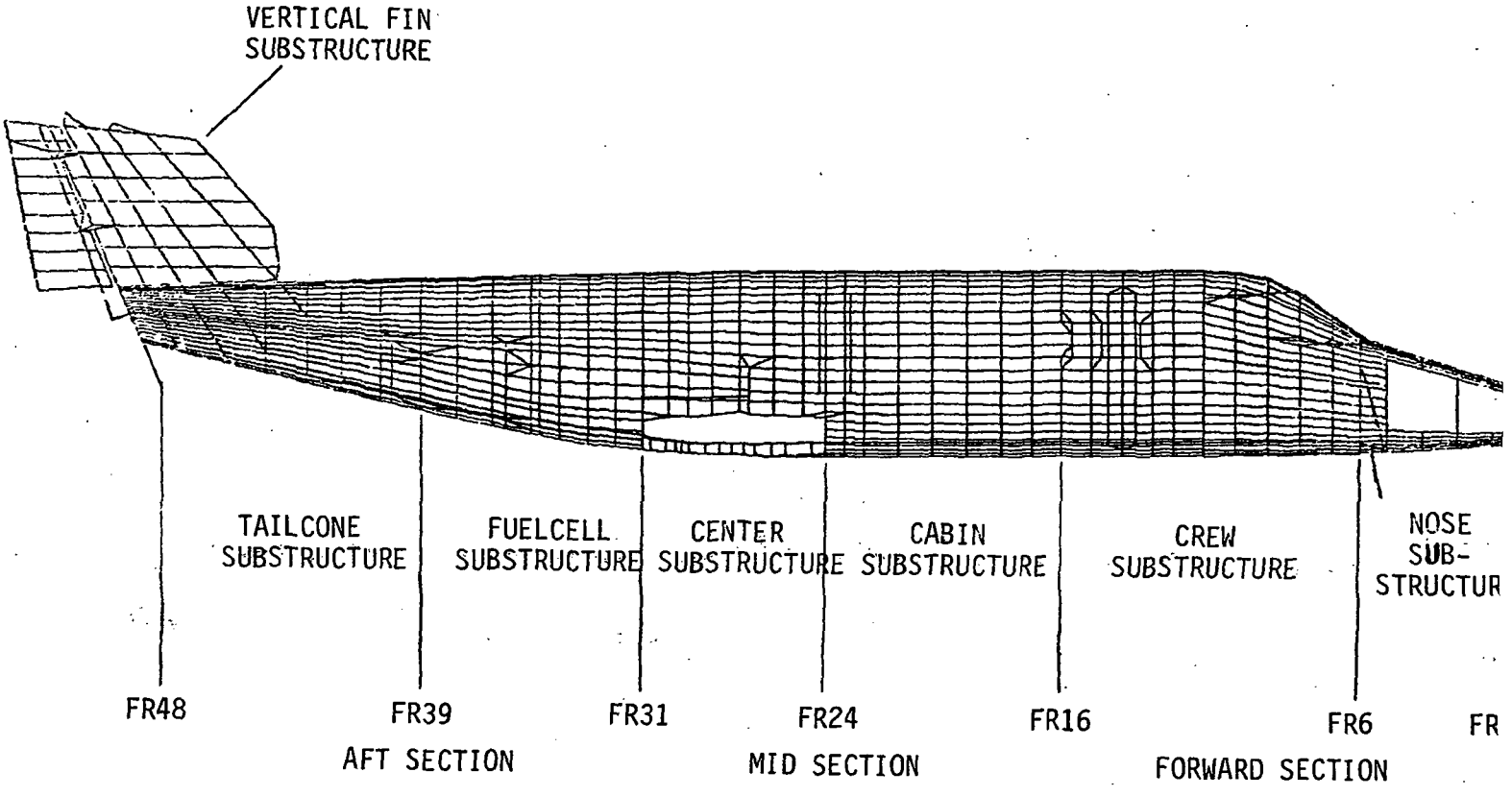


FIGURE 2 - NASTRAN 54/55 FUSELAGE MODEL

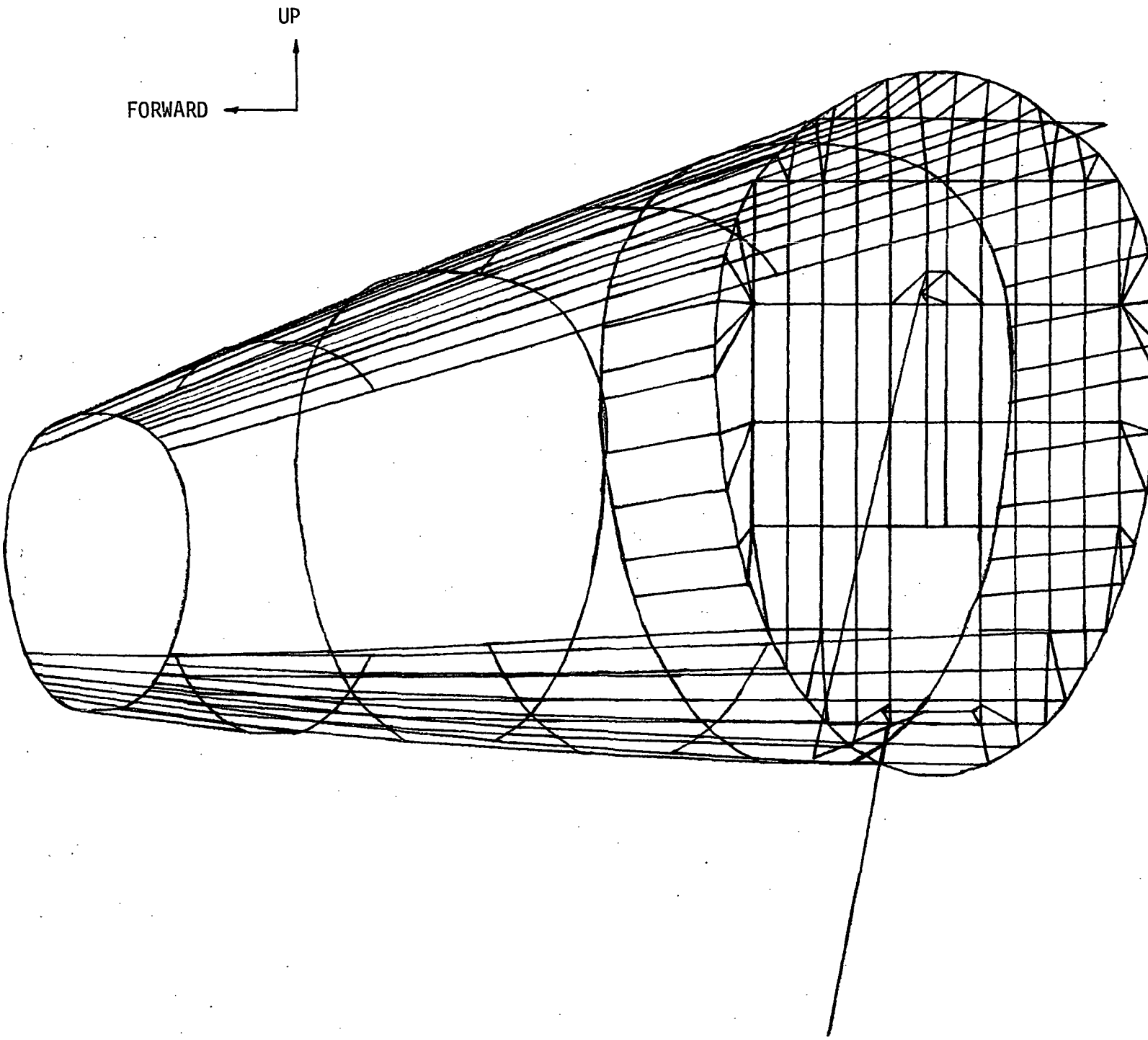


FIGURE 3 - NOSE SUBSTRUCTURE

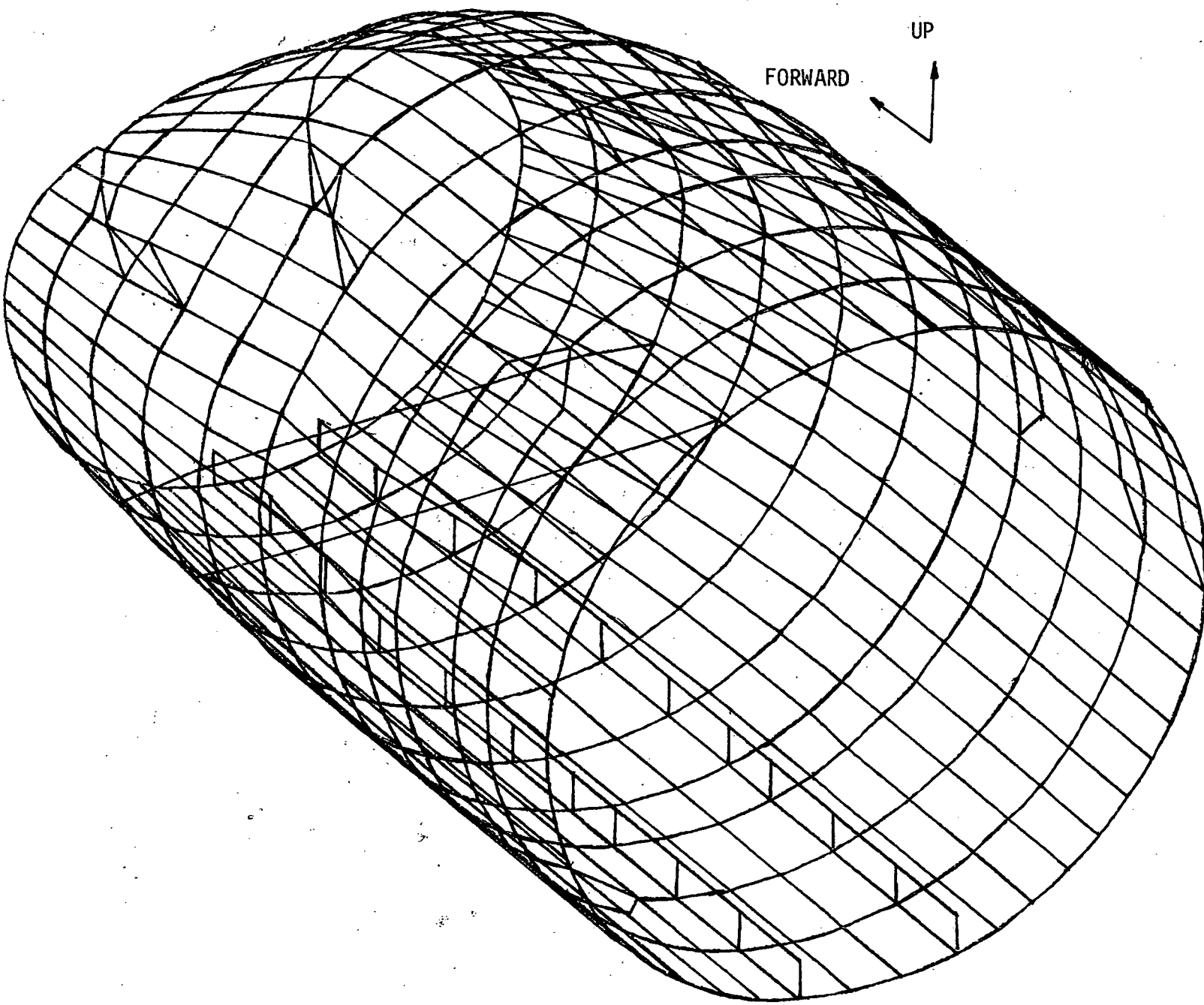


FIGURE 4 - CREW SUBSTRUCTURE

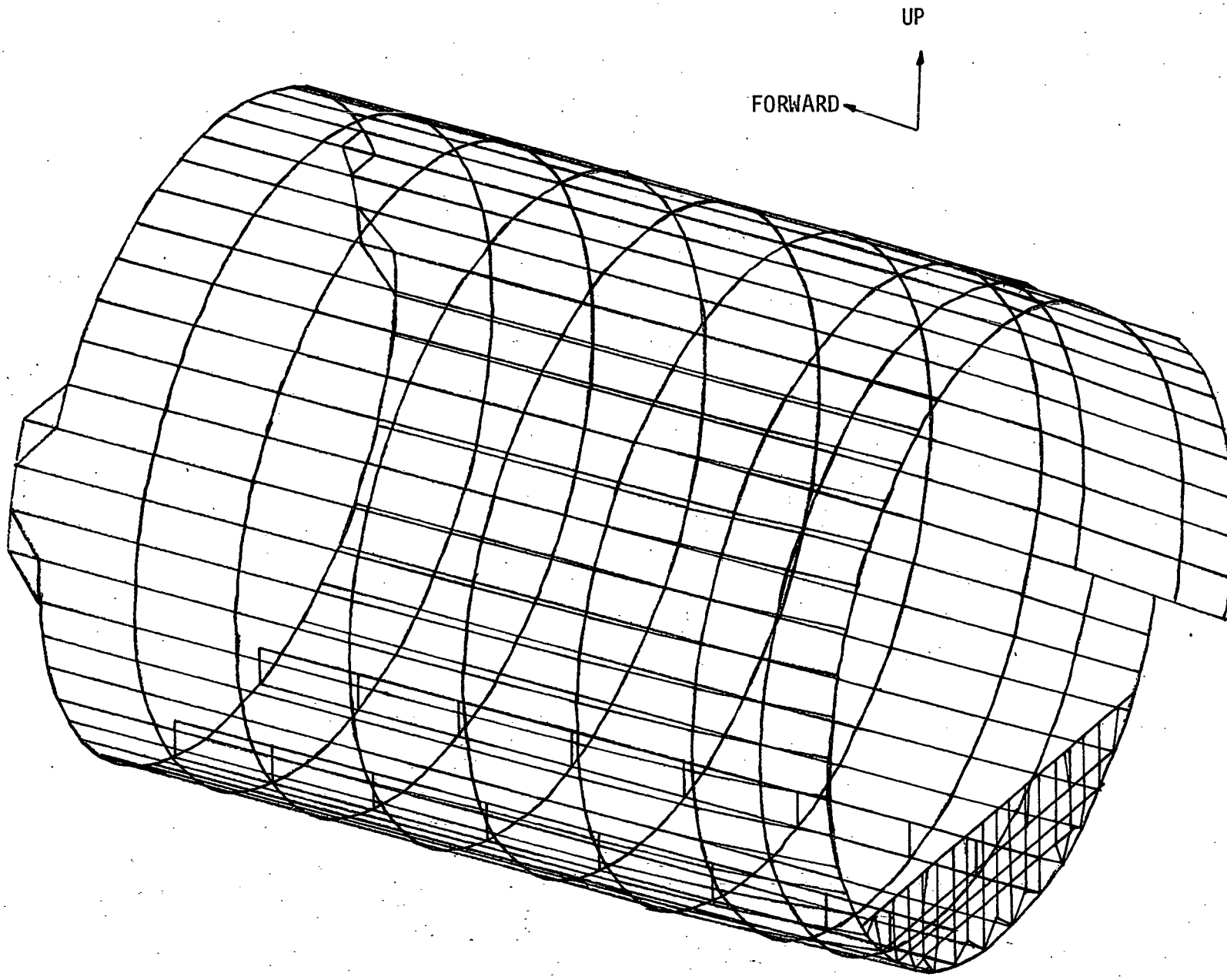


FIGURE 5 - CABIN SUBSTRUCTURE

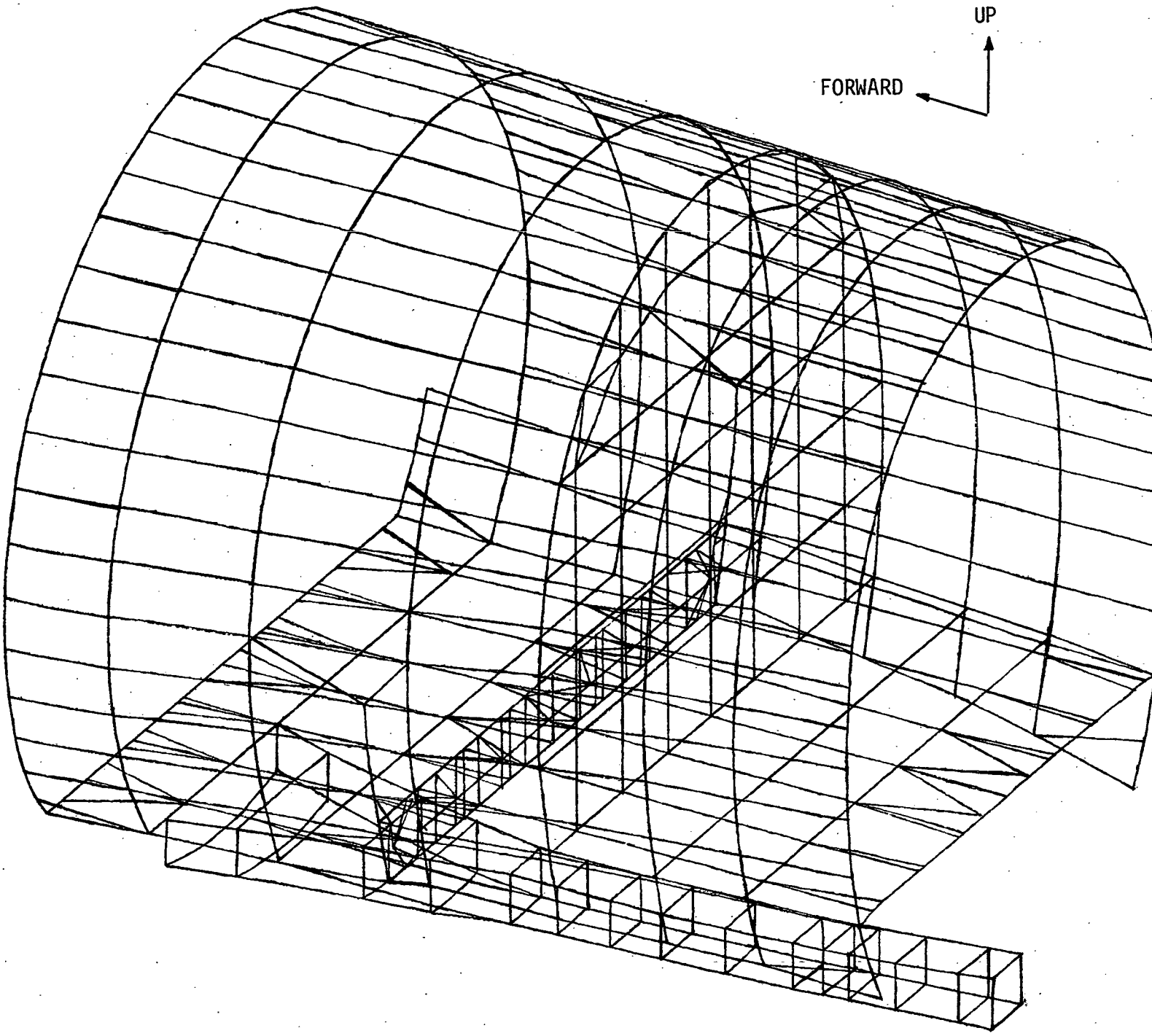


FIGURE 6 - CENTER SUBSTRUCTURE

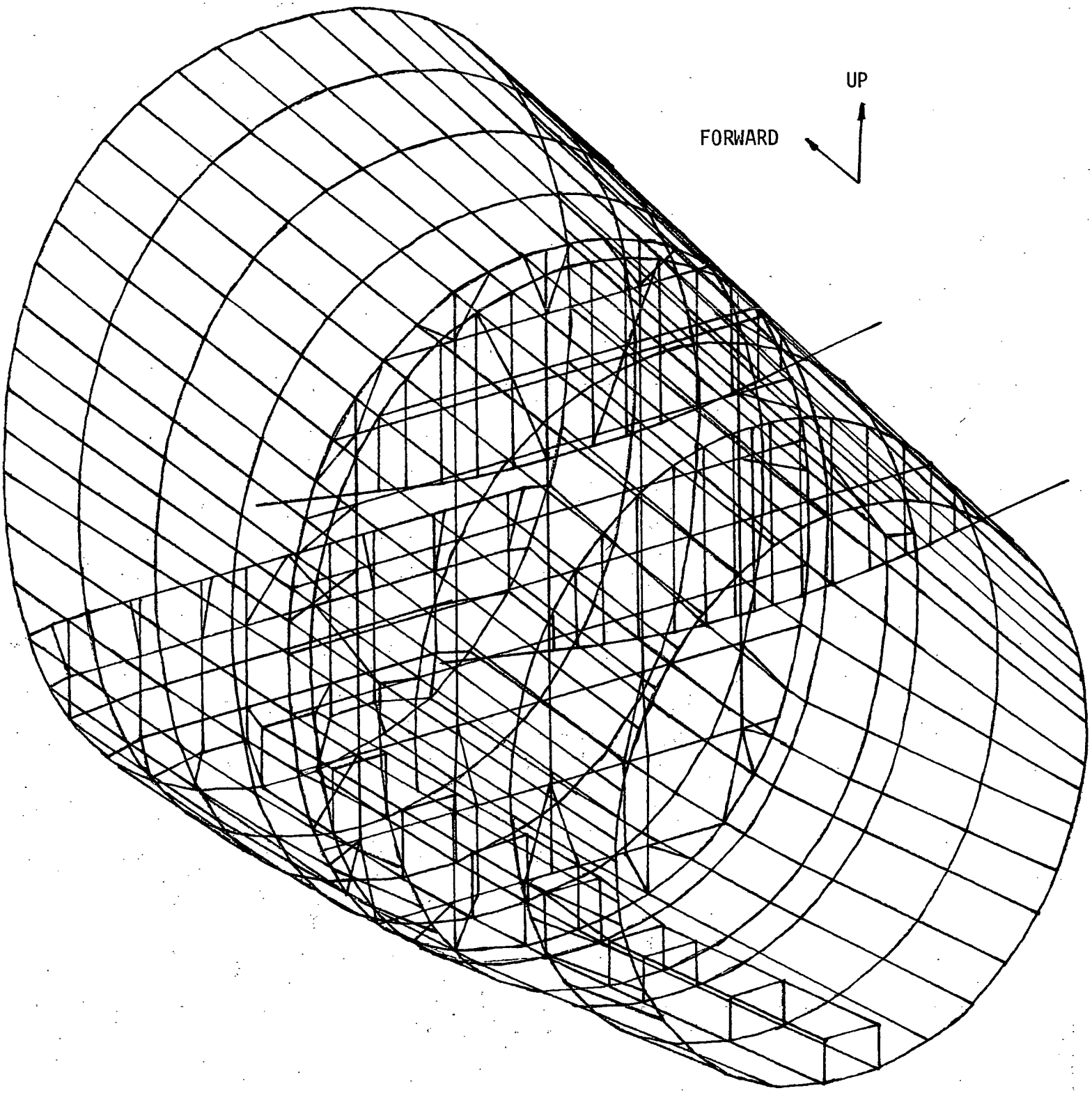


FIGURE 7 - FUELCELL SUBSTRUCTURE

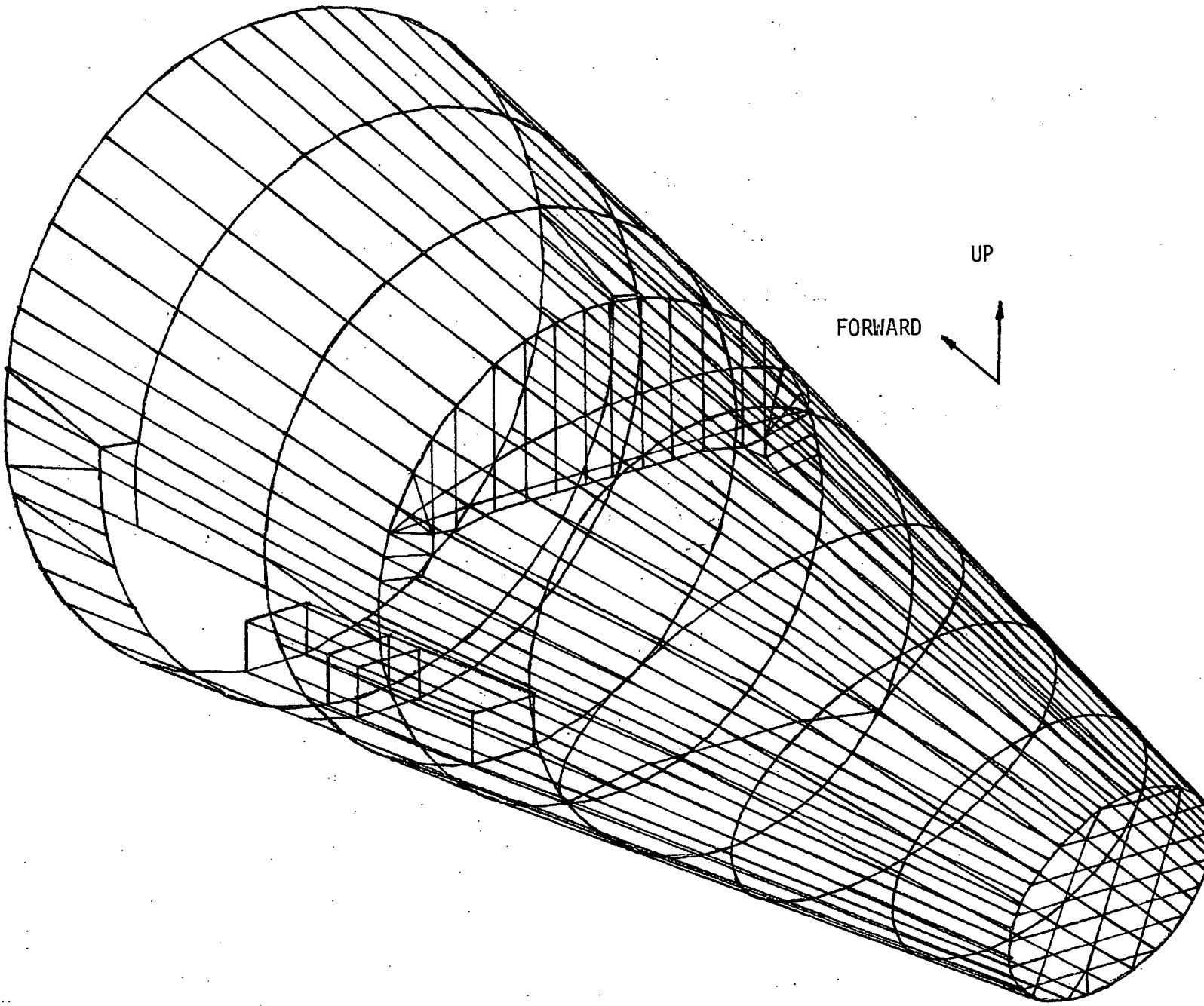


FIGURE 8 - TAILCONE SUBSTRUCTURE

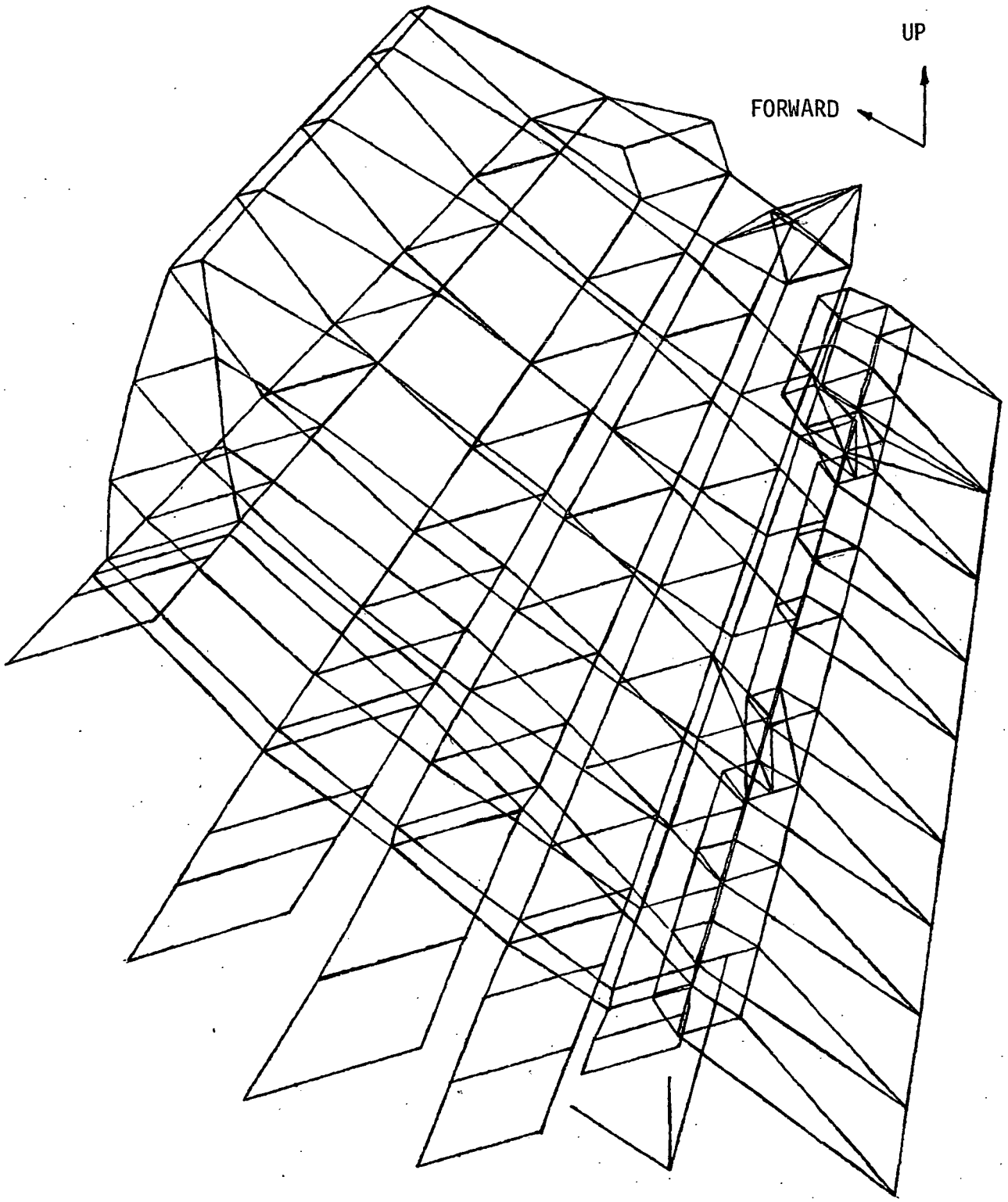


FIGURE 9 - VERTICAL FIN SUBSTRUCTURE

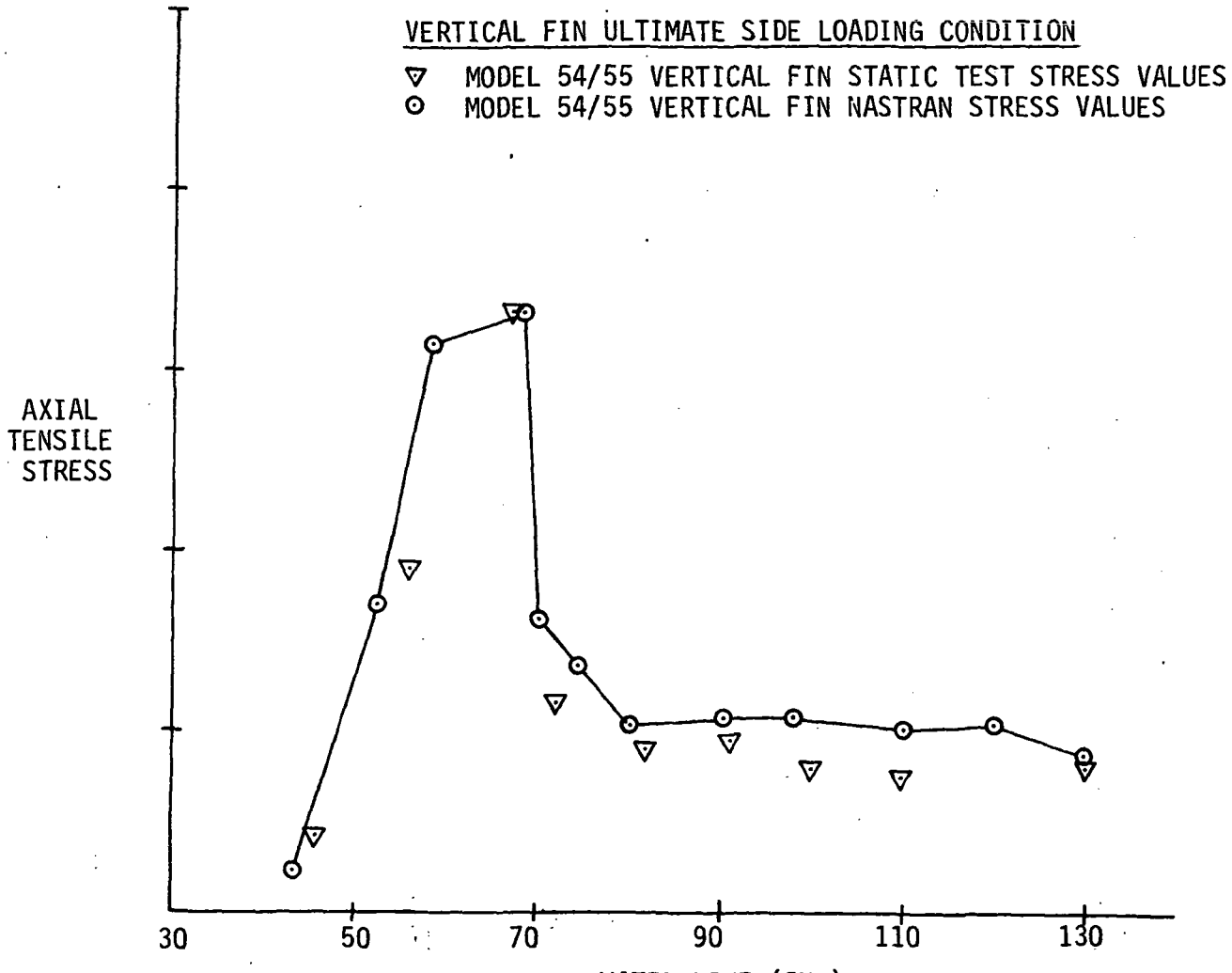


FIGURE 10 - AXIAL TENSILE STRESS IN SPAR 2 (L.H.)

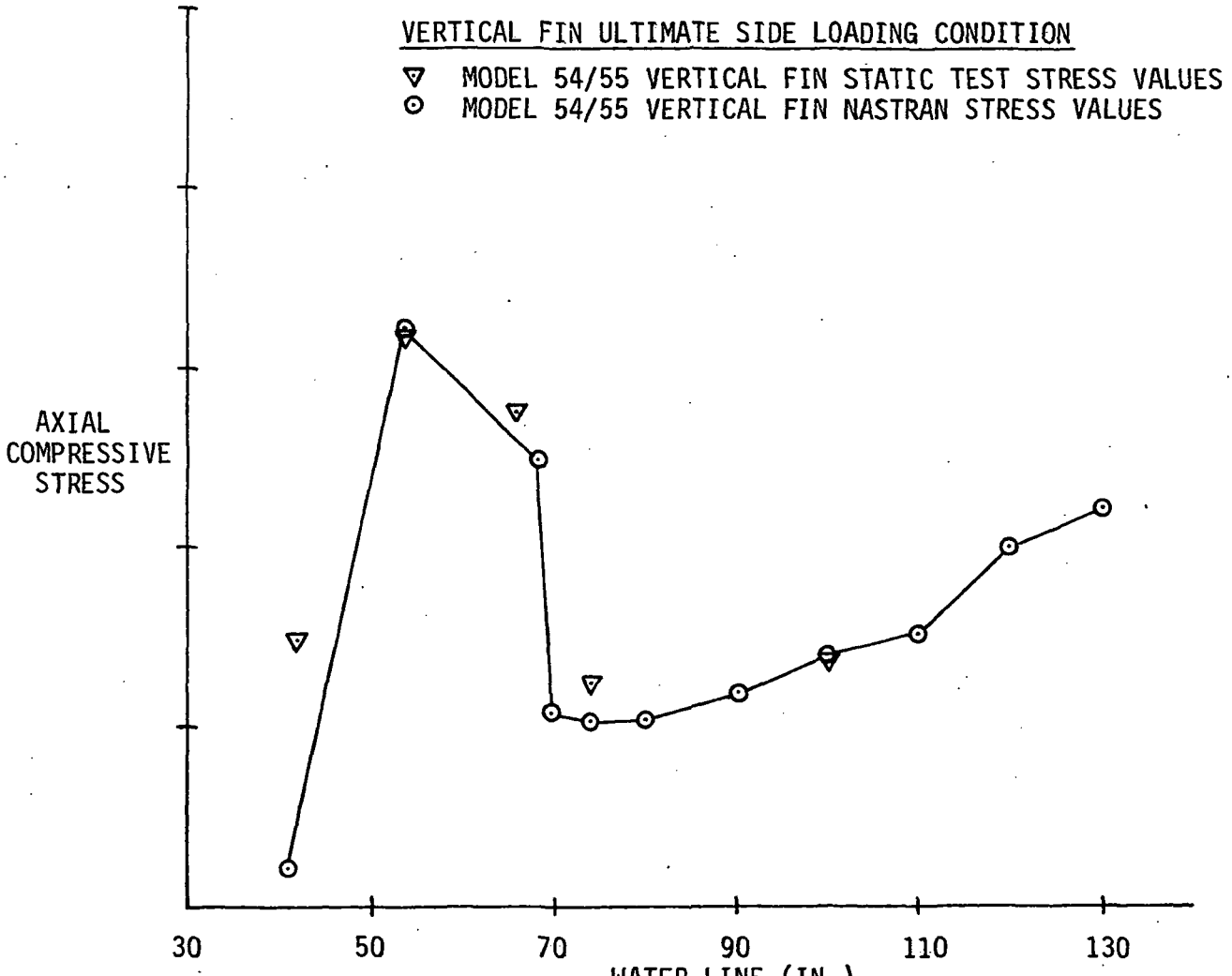


FIGURE 11 - AXIAL COMPRESSIVE STRESS IN SPAR 3 (R.H.)

XXX = NASTRAN STRESS (KSI)
YYY EXPERIMENTAL STRESS (KSI)

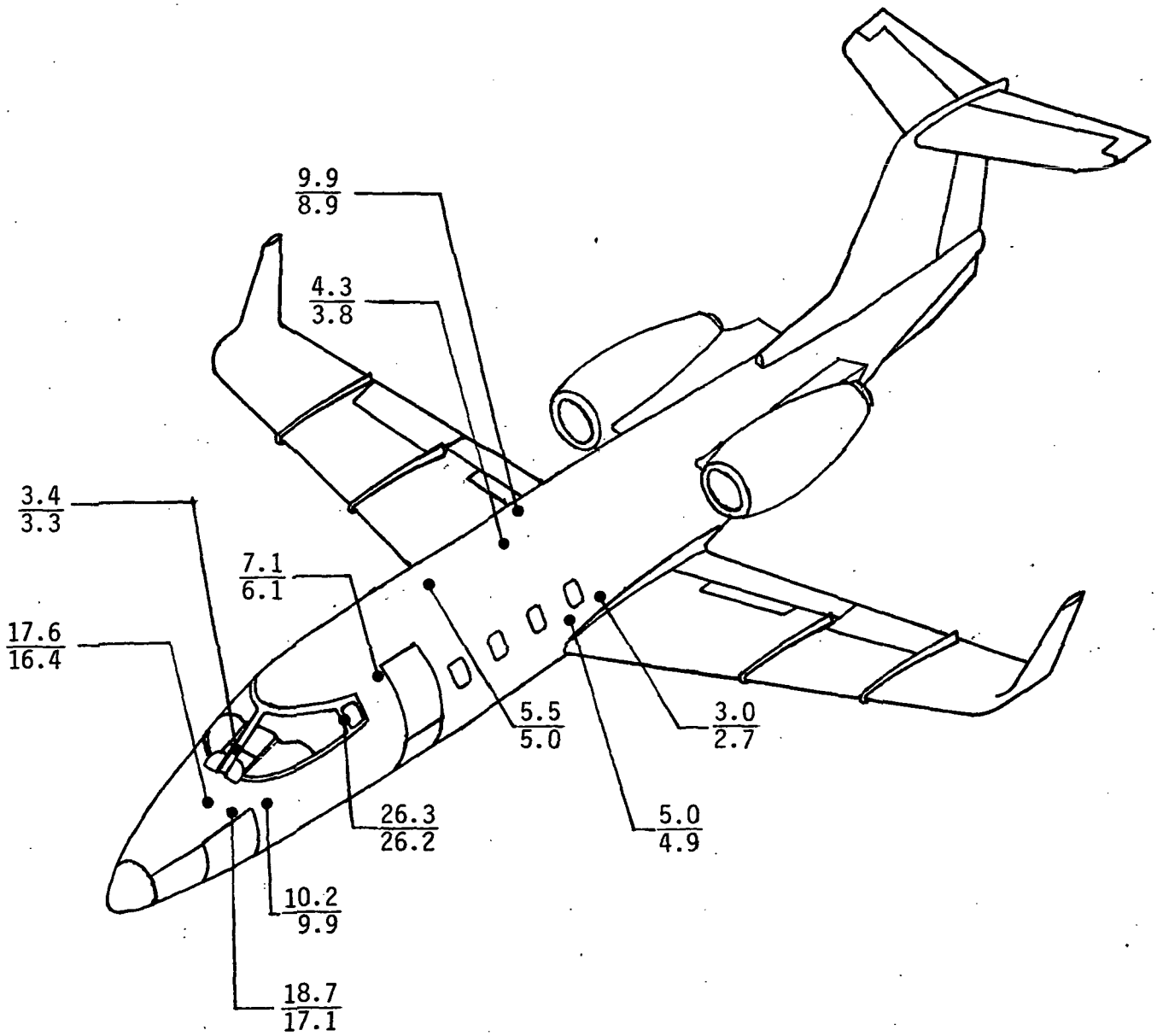


FIGURE 12 - LIMIT PRESSURE LOAD STATIC TEST VALUES AND NASTRAN VALUES

COMPARISON OF FINITE ELEMENT ANALYSES OF A PIPING TEE USING NASTRAN AND CORTES/SA

Antonio J. Quezon and Gordon C. Everstine
David W. Taylor Naval Ship Research and Development Center

SUMMARY

A comparison of finite element analyses of a 24" x 24" x 10" piping tee was made using NASTRAN and CORTES/SA, a modified version of SAP3 having a special purpose input processor for generating geometries for a wide variety of tee joints. Four finite element models were subjected to force, moment, and pressure loadings. Flexibility factors and principal stresses were computed for each model and compared with results obtained experimentally by Combustion Engineering, Inc. Of the four models generated, the first was generated from actual measured geometry using GPRIME, a geometric and finite element modeling system developed at DTNSRDC. The other three models were generated from an idealized tee using the data generator contained in CORTES/SA.

The generation of an idealized tee proved to be very easy and inexpensive compared to generation from actual geometry, and, when analyzed by NASTRAN, proved adequate. Results from the NASTRAN analyses were in good agreement with experimental results for all loadings except internal pressure. The CORTES/SA analyses gave good results for the internal pressure loading, but poorer results for out-of-plane bending moments or forces resulting in out-of-plane bending. Two of the basic load cases in CORTES/SA were found to contain errors that could not be easily corrected. A cost comparison of NASTRAN and CORTES/SA showed NASTRAN to be less expensive to run than CORTES/SA for identical meshes. Overall, considering modeling effort, cost, and accuracy, it is concluded that tees can be easily and accurately analyzed by NASTRAN using an idealized mesh generated by CORTES/SA.

BACKGROUND

The designer of a piping system requires a knowledge of the deflections and stresses caused throughout the system by anticipated service loads. Of particular interest are critical components such as elbows and tees.

The purpose of this paper is to summarize the results of a study (ref. 1) made recently to assess the effectiveness of the finite element method (FEM) in predicting flexibility factors and stresses in piping tees subjected to force, moment, and pressure loadings. A similar study (ref. 2), performed for piping elbows, indicated that very good agreement could be expected between FEM

analysis and experiment. Tees, although conceptually no more difficult to analyze than elbows, are considerably more complicated geometrically. A reducing tee, for example, has in the crotch region a fillet with a variable radius of curvature as well as variable thickness. Moreover, the adjacent straight sections may not be cylindrical. Thus, geometrical idealizations of tees, although plausible, may be incorrect.

The finite element analyses described here involve idealized models as well as a model based on actual measured geometry. Two computer programs were used for the analyses: NASTRAN, a widely-used general purpose finite element structural analysis program, and CORTES/SA (ref. 3), a special purpose finite element tee analysis program based on SAP3 and written at the University of California at Berkeley under the sponsorship of the Oak Ridge National Laboratory.* This series of analyses was designed to provide information on the sensitivity of the results to various mesh densities as well as on the adequacy of the assumed idealizations.

In this paper, the program CORTES/SA will be referred to by the abbreviated name "CORTES".

STATEMENT OF THE PROBLEM

Combustion Engineering, Inc., performed an experimental stress analysis (ref. 4) on an ANSI B16.9 carbon steel tee designated T-12. Pipe extensions were welded to the branch and run ends of the tee, and the resulting assembly was placed in a load frame. One of the run ends was built in to represent a fixed end, and the other run end and the branch end were used to apply six orthogonal moments and five orthogonal forces. Internal pressure was also applied. Table 1 and Figure 1 summarize the applied loads. Load case 4 (F3X) was not tested because of strength limitations of the load frame. Stress data for all twelve load cases were gathered from strain gages fixed on specific rows on the tee (Figure 2) and were plotted against normalized surface distance.

The tee analyzed was a reducing tee with a 24-inch-diameter run end and a 10.75-inch-diameter branch. Loads to the run were applied at the free end of the run pipe extension, 173 inches from the branch-run intersection (Figure 1). Loads to the branch were applied at the end of the branch pipe extension, 77 inches from the branch-run intersection. The run pipe extension consisted of 24-inch-diameter schedule 40 (0.687-inch nominal wall thickness) carbon steel piping. The branch pipe extension consisted of 10.75-inch-diameter schedule 40 (0.365-inch nominal wall thickness) carbon steel piping.

The finite element analyses of the tee simulated these loading conditions so that stresses at selected locations could be compared to the experimental

---

* The CORTES package of computer programs is distributed as program number 759 by the National Energy Software Center (NESC), Argonne National Laboratory, 9700 S. Cass Avenue, Argonne, Illinois 60439.

results. For most load cases, the strain gage rows (Figure 2) selected for comparison were those on which the peak stresses occurred.

ANALYSES PERFORMED

NASTRAN analyses were performed for the first three models generated, and CORTES analyses were performed for the third and fourth models. These five finite element analyses are summarized in Table 2. In the abbreviations N1, N2, N3, C3, and C4 used to identify the analyses, the first character (N or C) indicates the analysis program used (NASTRAN or CORTES), and the second character indicates the mesh used. A typical mesh generated by CORTES is shown in Figure 3. In general, a higher mesh number corresponds to a finer mesh, either overall or in selected key regions of the tee.

The NASTRAN analysis of Mesh 1 was the only analysis performed for a model generated from actual measured geometry. The remaining analyses were performed either by NASTRAN or by CORTES on meshes generated by CORTES assuming an idealized geometry. In all cases only one-fourth of the actual tee was modeled due to symmetry.

For the NASTRAN analyses, the tee, including pipe extensions, was modeled with plate (NASTRAN QUAD2) elements. Flexible beam (BAR) elements were arranged in a spoke formation radiating from an imaginary point in the center of the cross section at the ends of the tee branch and run to facilitate the calculation of the average rotation of these cross sections. Rigid (RIGD1) elements were defined at the ends of the pipe extensions for use in load application. The loads were applied to a point in the center of the rigid cross section at the ends of the pipe extensions.

In the CORTES analyses, the tee and pipe extensions were modeled using an 8-node hexahedral element. This element, designated ZIB8R9, is a modification of the standard Zienkiewicz-Irons isoparametric element and, according to Gantayat and Powell (ref. 3), has bending properties superior to those of the unmodified isoparametric element.

Mesh 1 was modeled from actual geometry as specified in the Combustion Engineering, Inc., report (ref. 4) which tabulated coordinates of points on the outer surface of the tee and thicknesses at these points. From these digitized data, a general B-spline surface was fitted through the supplied points using the geometric and finite element modeling processor GPRIME (refs. 5 and 6). Once this geometric model was defined, GPRIME was used to generate a finite element mesh which included the effects of variable thicknesses.

Meshes 2 through 4 were modeled as idealized tee joints using the automatic mesh generation routine in CORTES. The tee joint is idealized by shallow cones representing the branch and run portions of the tee, connected to each other through an analytically defined transition fillet (ref. 3).

STRESS RESULTS

The results of primary interest are normalized principal stress values for elements in particular locations on the tee. The peak normalized principal stress was plotted against surface distance ratio for each load case and compared to the experimental results obtained by Combustion Engineering, Inc. (ref. 4).

The tee analyzed by Combustion Engineering, Inc., was heavily instrumented, both internally and externally, with strain gages in two of the four quadrants. The gages in each quadrant were arranged in six rows as shown in Figure 2. Since the peak stresses for most load cases occurred on row 1 or row 6, analytical and experimental results were compared for these rows only.

For each load case, the analytical results for principal stresses were normalized by a stress calculated from beam theory, as indicated in Table 1. The normalized principal stresses were then plotted against the surface distance ratios of the elements lying on row 1 and row 6.

Stress plots for several typical load cases are shown in Figures 4 through 8. (Ref. 1 contains plots for all load cases.) All finite element curves are smoothed slightly by fitting B-spline curves (refs. 7 and 8) through the computed values, which are located at element centroids for the NASTRAN results and at grid points for the CORTES results.

FLEXIBILITY FACTORS

Two ambiguities were encountered in comparing computed flexibility factors with experimental results obtained by Combustion Engineering, Inc. These ambiguities involved the definition of flexibility factors and the way in which the rotation of branch or run end cross sections was measured. Combustion Engineering, Inc., defined the flexibility factors as

$$k = \frac{\theta_{meas} - \theta_{corr}}{\theta_{nom}} \tag{1}$$

where

θ_{meas} = measured rotation at an intermediate location on the pipe extension

θ_{corr} = rotation correction computed by simple beam theory for the length of pipe between the tee weld line and the location at which the rotation is actually measured

θ_{nom} = nominal rotation computed by simple beam theory for the distance between the tee weld lines where

$$\theta_{nom} = \frac{ML}{EI} \quad \text{(for bending moments)} \tag{2}$$

$$\theta_{nom} = \frac{TL}{JG} \quad \text{(for torsional moments)} \tag{3}$$

$$\theta_{nom} = \frac{PL^2}{2EI} \quad \text{(for point loads)} \tag{4}$$

Since Combustion Engineering, Inc., could not measure the actual rotations at the branch and run end cross sections, measurements were made at other locations on the pipe extensions and then corrected to the branch and run ends using simple beam theory. On the other hand, the NASTRAN analyses used very flexible beam elements radiating from an imaginary point in the center of the branch and run end cross sections to the points on the circumference of the branch and run ends. This modeling technique allowed an approximate average rotation for the cross sections to be easily obtained for the imaginary center point. However, because plane sections do not, under loading, remain plane, there is no single rotation for a section, so that different methods for computing rotations will yield different results.

For the computation of flexibility factors from the NASTRAN results, the relation

$$k = \frac{\theta_{ab}}{\theta_{nom}} \tag{5}$$

was used, where

θ_{ab} = computed relative rotation of end "a" with respect to end "b"

θ_{nom} = nominal rotation computed by beam theory for the rotation of end "a" with respect to end "b"

Flexibility factors were computed for the free branch and run ends with respect to the fixed run end for each load case except for F2X (an axial load on the run) and internal pressure, neither of which causes any significant rotation. For example, the flexibility factor for a rotation about the X-axis of the branch end with respect to the fixed run end is denoted by k_{X31}, where the X in the subscript represents the axis of rotation, the 3 represents the branch end, and the 1 represents the fixed run end. For each load case, flexibility factors for each cross section were computed.

Table 3 compares the flexibility factors computed from the three NASTRAN analyses to the experimental values. The computed flexibility factors compare reasonably well for most load cases, an exception being k_{Z21} for load case 5 (F3Y) of N2. Combustion Engineering, Inc., did not compute flexibility factors for this load case because the stresses and deflections were considered too small to give reliable answers. The displacements computed in the three NASTRAN analyses for load case 5, however, did not appear to be significantly smaller than those of the other load cases, although the run end of the tee did

warp severely in all three analyses. Since the distortions in all three analyses were similar, it appears to have been due to chance that the flexibility factors for N1 and N3 were not also negative for this load case. This implies that any method used to compute a single rotation of the run end is inadequate for severely distorted cross sections. Moreover, the usefulness of a flexibility factor when severe cross-sectional distortion occurs is questionable.

In general, a negative flexibility factor, whether arising from experiment or analysis, is physically impossible, since such a factor implies a rotation in a direction opposite to that of the applied moment. Negative values can arise experimentally whenever rotations measured at one location have to be "corrected" (using beam theory) to yield rotations elsewhere. Negative values can result from a finite element analysis whenever severe cross-sectional distortion occurs, in which case the usefulness of an "average" rotation of the cross section is in doubt.

DISCUSSION OF RESULTS AND CONCLUSIONS

The three NASTRAN analyses of the tee joint were generally in very good agreement with the experimental results and accurately predicted peak stresses for most loadings except load cases 3 (M3Z) and 13 (pressure). Also, as expected, the agreement with the experimental results improved with finer meshes. In general, the two CORTES analyses were slightly less accurate than the NASTRAN analyses except for load case 13 (pressure). We are unable to explain this behavior. While the CORTES results for pressure loading were significantly better than NASTRAN's, the results for load cases 1 (M3X), 8 (M2Y), 10 (F2X), and 12 (F2Z) were worse. Note that most of these load cases involve either out-of-plane bending moments or forces resulting in out-of-plane bending. The CORTES analyses of load cases 6 (F3Z) and 7 (M2X) were also found to contain errors in formulation and coding which could not be easily corrected.

The preparation of the NASTRAN model of mesh 1 (called N1) was the most time-consuming and expensive of all the models, since this mesh was generated from actual geometry. Although the N1 calculations for all load cases except pressure (Figure 8) are in very good agreement with the experimental results, they are not significantly better than those obtained from the other analyses, so the extra effort is not justified.

In a comparison of NASTRAN and CORTES analyses of an identical mesh (Mesh 3), the NASTRAN results (N3) were more consistent and predicted peak stresses more accurately than CORTES for ten of the twelve load cases. Only for M3Z and pressure (Fig. 8) did C3 do better than N3. Although N3 was less expensive than C3 in computer costs, it required slightly more time for input preparation.

Since N3 was in generally better agreement with experimental results than C3, a coarser mesh (Mesh 2) was also analyzed by NASTRAN (N2) and compared

with C3. In all but three of the load cases, M3Z, F3Y, and pressure, N2 was again in better agreement with experimental results than C3. Computer costs from N2 were significantly less than those for C3, as indicated in Table 2.

In an effort to obtain better results from CORTES, a much finer mesh (Mesh 4) was generated and analyzed, so that the results could be compared with N3. This time, overall performance was about equal for the two analyses, although C4 achieved better results than N3 for M3Z, F3Y, M2Z, F2Y, and pressure.

In conclusion, it is apparent that GPRIME, although well-suited in general to the generation of tee meshes based on actual geometry, is more difficult and time-consuming to use than the special purpose idealized tee generator contained in CORTES. Models based on actual geometry also require geometric data that would probably not be generally available to the analyst. For these reasons, CORTES generation of a finite element model based on idealized geometry appears to be acceptable. However, if an analyst is interested in an F3Z or an M2X loading, CORTES should not be used as the analyzer because the program currently contains errors in the coding of these two load cases. Also, as shown by the comparison of N3 with C4, CORTES requires a mesh about 20% finer to obtain results as accurate as NASTRAN.

Overall, considering modeling effort, cost, and accuracy, it is concluded that tees can be easily and accurately analyzed by NASTRAN using an idealized mesh generated by CORTES/SA.

REFERENCES

1. Quezon, A.J.; Everstine, G.C.; and Golden, M.E.: Finite Element Analysis of Piping Tees. Report DTNSRDC/CMLD-80/11, David W. Taylor Naval Ship Research and Development Center, Bethesda, Maryland, June 1980.

2. Marcus, M.S.; and Everstine, G.C.: Finite Element Analysis of Pipe Elbows. Report DTNSRDC/CMLD-79/15, David W. Taylor Naval Ship Research and Development Center, Bethesda, Maryland, Feb. 1980.

3. Gantayat, A.N.; and Powell, G.H.: Stress Analysis of Tee Joints by the Finite Element Method. Report No. US SESM 73-6, Structural Engineering Laboratory, Univ. of California, Berkeley, California, Feb. 1973.

4. Henley, D.R.: Test Report on Experimental Stress Analysis of a 24" Diameter Tee (ORNL T-12). Report CENC 1237, ORNL-Sub-3310-4, Combustion Engineering, Inc., Chattanooga, Tennessee, Apr. 1975.

5. Golden, M.E.: Geometric Structural Modelling: A Promising Basis for Finite Element Analysis. Trends in Computerized Structural Analysis and Synthesis, ed. by A.K. Noor and H.B. McComb, Jr., Pergamon Press, Oxford, England, May 1978, pp. 347-350.

6. Golden, M.E.: The Role of a Geometry Processor in Structural Analysis. New Techniques in Structural Analysis by Computer, ed. by R. Melosh and M. Salama, Preprint 3601, ASCE Convention and Exposition (2-6 Apr 1979), American Society of Civil Engineers, Boston, MA, pp. F1-F17.

7. McKee, J.M.; and Kazden, R.J.: G-Prime B-Spline Manipulation Package--Basic Mathematical Subroutines. DTNSRDC Report 77-0036, David W. Taylor Naval Ship Research and Development Center, Bethesda, Maryland, Apr. 1977.

8. McKee, J.M.: Updates to the G-Prime B-Spline Manipulation Package--26 October 1977. Periodic updates available from the author at the David W. Taylor Naval Ship Research and Development Center, Bethesda, Maryland 20084.

TABLE 1 - SUMMARY OF APPLIED AND NORMALIZED LOADS

Load Case	Applied Load	Nominal Stress	Normalized Load
1. M3X	4.29E5 in-lb	$\frac{M3X}{Z_b}$	29.91
2. M3Y	-6.03E5 in-lb	$\frac{M3Y}{Z_b}$	29.91
3. M3Z	5.98E5 in-lb	$\frac{M3Z}{Z_b}$	29.91
5. F3Y	4.0E4 lb	$\frac{F3Y}{A_b}$	11.91
6. F3Z	5.58E3 lb	$\frac{77F3Z}{Z_b}$	3.884E-1
7. M2X	4.9E6 in-lb	$\frac{M2X}{Z_r}$	285.0
8. M2Y	-7.54E6 in-lb	$\frac{M2Y}{Z_r}$	285.0
9. M2Z	3.40E6 in-lb	$\frac{M2Z}{Z_r}$	285.0
10. F2X	-6.28E5 lb	$\frac{F2X}{A_r}$	50.3
11. F2Y	2.01E4 lb	$\frac{173F2Y}{Z_r}$	1.6474
12. F2Z	2.46E4 lb	$\frac{173F2Z}{Z_r}$	1.6474
13. P	600 psi	$\left(\frac{P D_0}{2t}\right)_r$	5.725E-2

Notes:

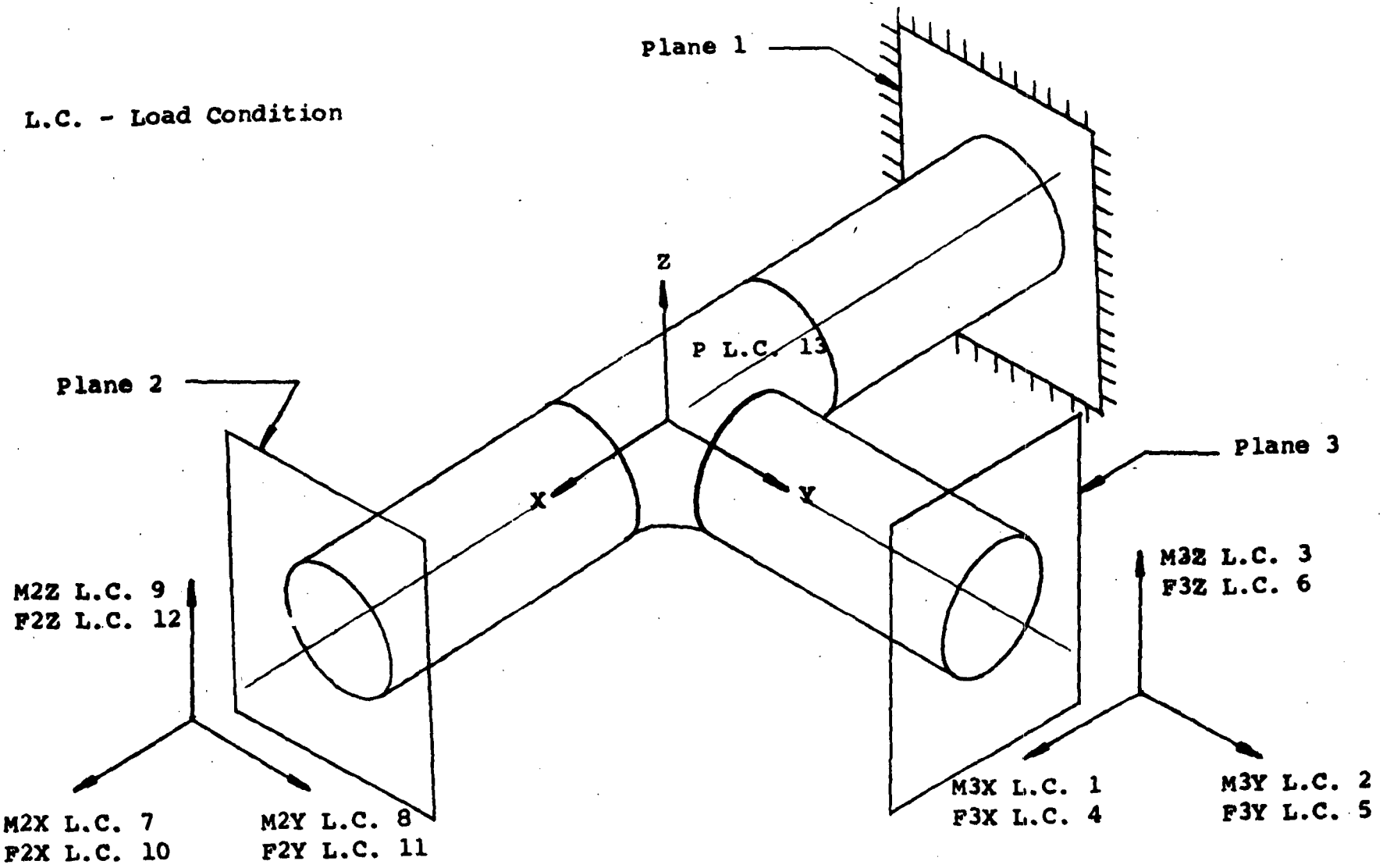
1. Load case 4 (F3X) was not tested.
2. The "normalized load" is computed by dividing the experimentally applied load (column 2) by the nominal stress (column 3).
3. Subscripts "r" and "b" above denote "run" and "branch", respectively.

TABLE 2 - COMPARISON OF FINITE ELEMENT ANALYSES:
NASTRAN vs. CORTES

	N1	N2	N3	C3	C4
NASTRAN or CORTES Analysis	NASTRAN	NASTRAN	NASTRAN	CORTES	CORTES
Idealized or Actual Geometry	Actual	Idealized	Idealized	Idealized	Idealized
Number of Elements	432	420	525	549	626
Number of Nodes	484	473	583	609	689
Number of Degrees of Freedom	2525	2462	3047	3458	3958
Total CP Seconds (CDC 6400)	2213	2200	3135	3310	4748
Cost	$228	$226	$335	$421	$605

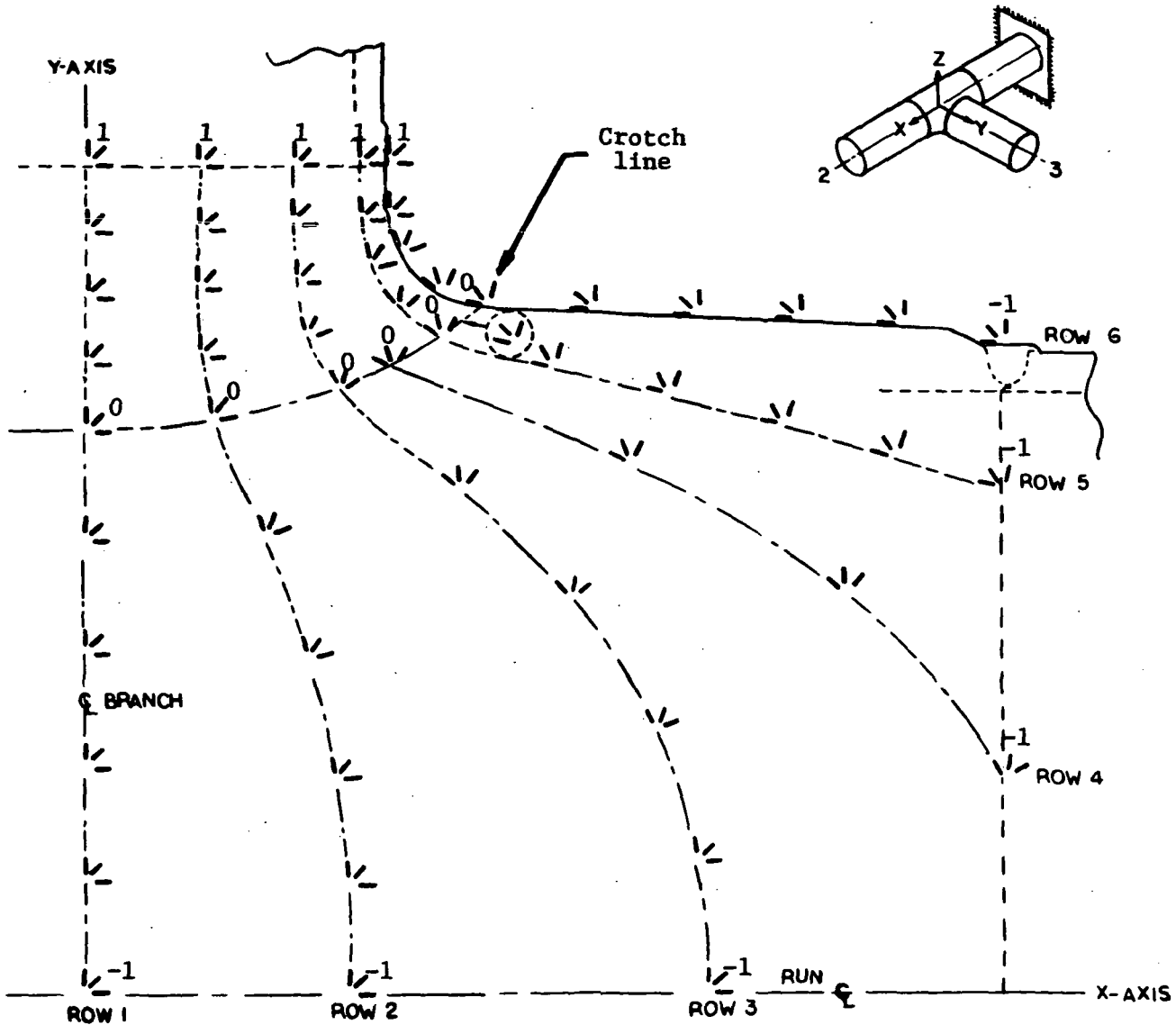
TABLE 3 - COMPARISON OF FLEXIBILITY FACTORS OF EXPERIMENTAL RESULTS TO NASTRAN RESULTS

Load Case	k Subscript	Experiment	N1	N2	N3
1. M3X	X21	-0.8	0.82	1.00	0.85
	X31	1.8	2.40	4.00	2.73
2. M3Y	Y21	0.5	0.72	0.76	0.77
	Y31	-0.3	0.32	0.33	0.32
3. M3Z	Z21	0.5	0.73	1.03	0.93
	Z31	0.9	0.90	0.85	0.84
5. F3Y	Z21		1.22	-1.35	1.53
	Z31		1.97	0.51	2.08
6. F3Z	X21		0.85	1.09	0.88
	X31	1.8	2.97	4.94	3.40
7. M2X	X21	-0.4	0.82	1.00	0.85
	X31	-0.5	0.82	1.00	0.85
8. M2Y	Y21	0.7	0.72	0.76	0.77
	Y31	0.6	0.72	0.76	0.77
9. M2Z	Z21	0.9	0.73	1.01	0.93
	Z31	0.8	0.73	1.01	0.93
11. F2Y	Z21	0.8	0.73	1.01	0.93
	Z31	0.8	0.83	1.08	1.01
12. F2Z	Y21	0.7	0.72	0.76	0.77
	Y31	0.7	0.91	0.89	0.94



Source: Figure 6, Ref. 4

Figure 1 - Schematic View of Applied Loads for Test Tee



Source: Figure 4, Ref. 4

Figure 2 - Location of Strain Gage Rows on Test Tee

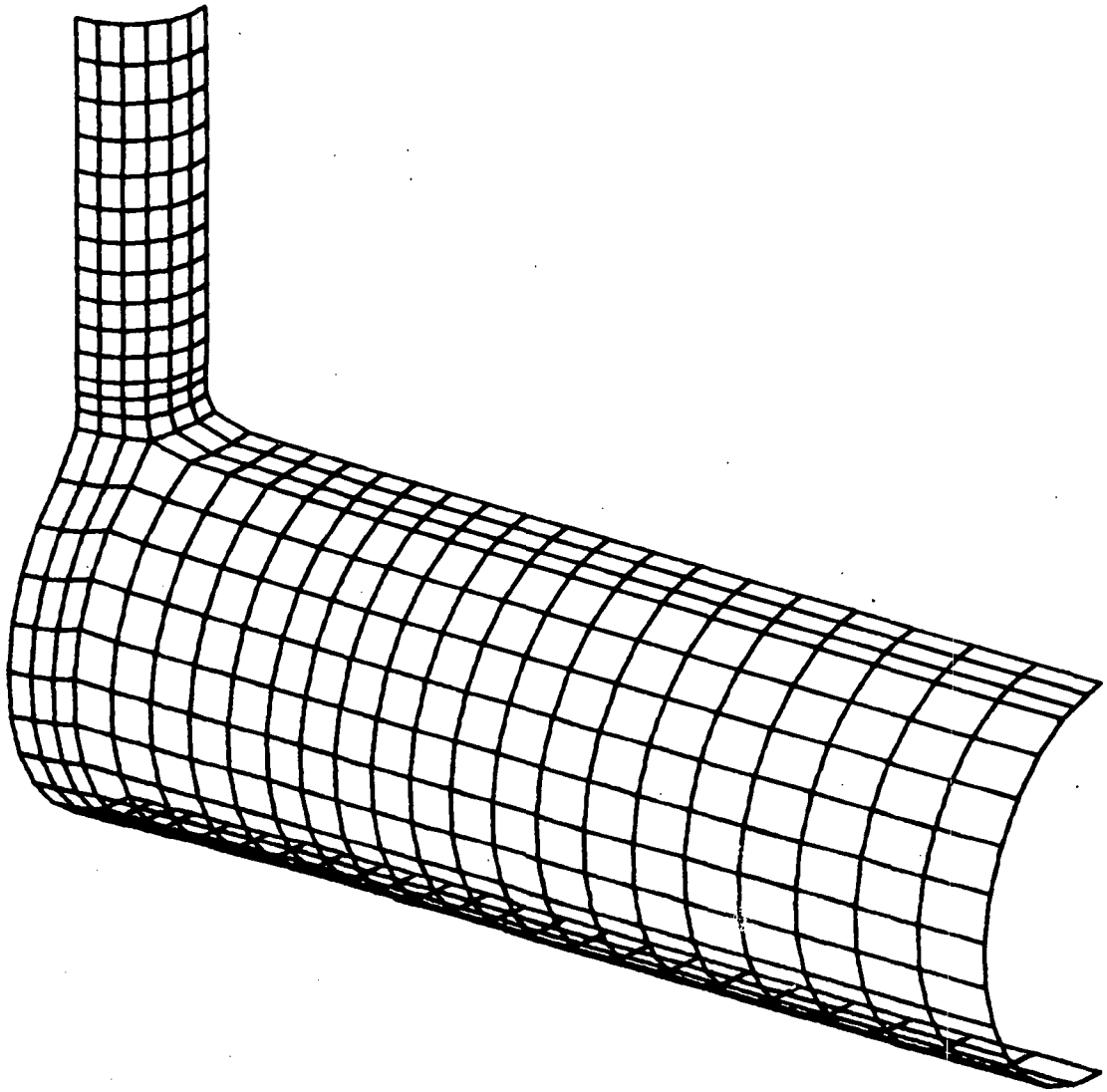


Figure 3 - Finite Element Model of Tee Using Idealized Geometry; Mesh 2

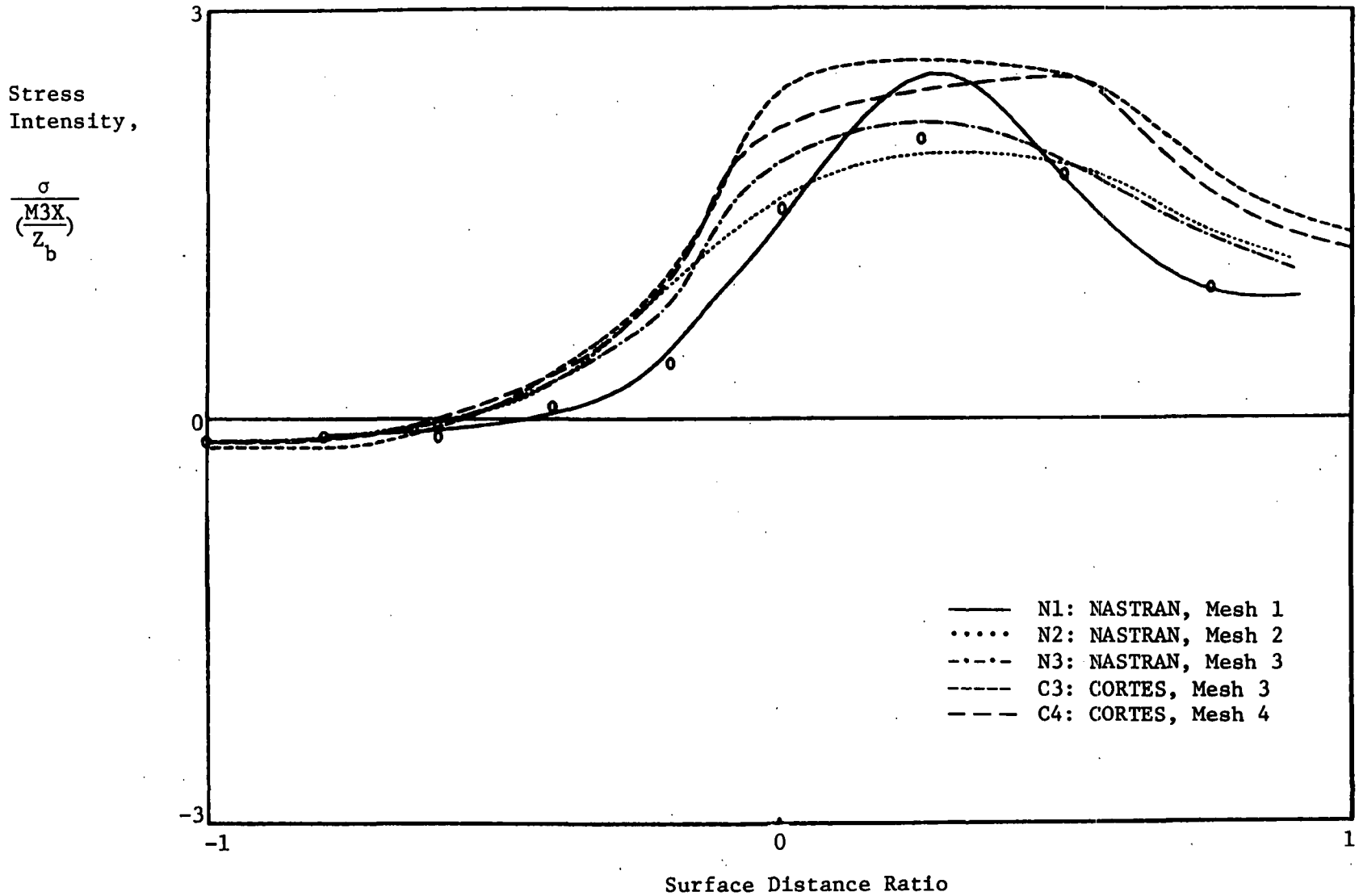


Figure 4 - Normalized Stress Intensity for Load Case 1 (M3X), Row 1, Major Principal Stress on Outer Surface

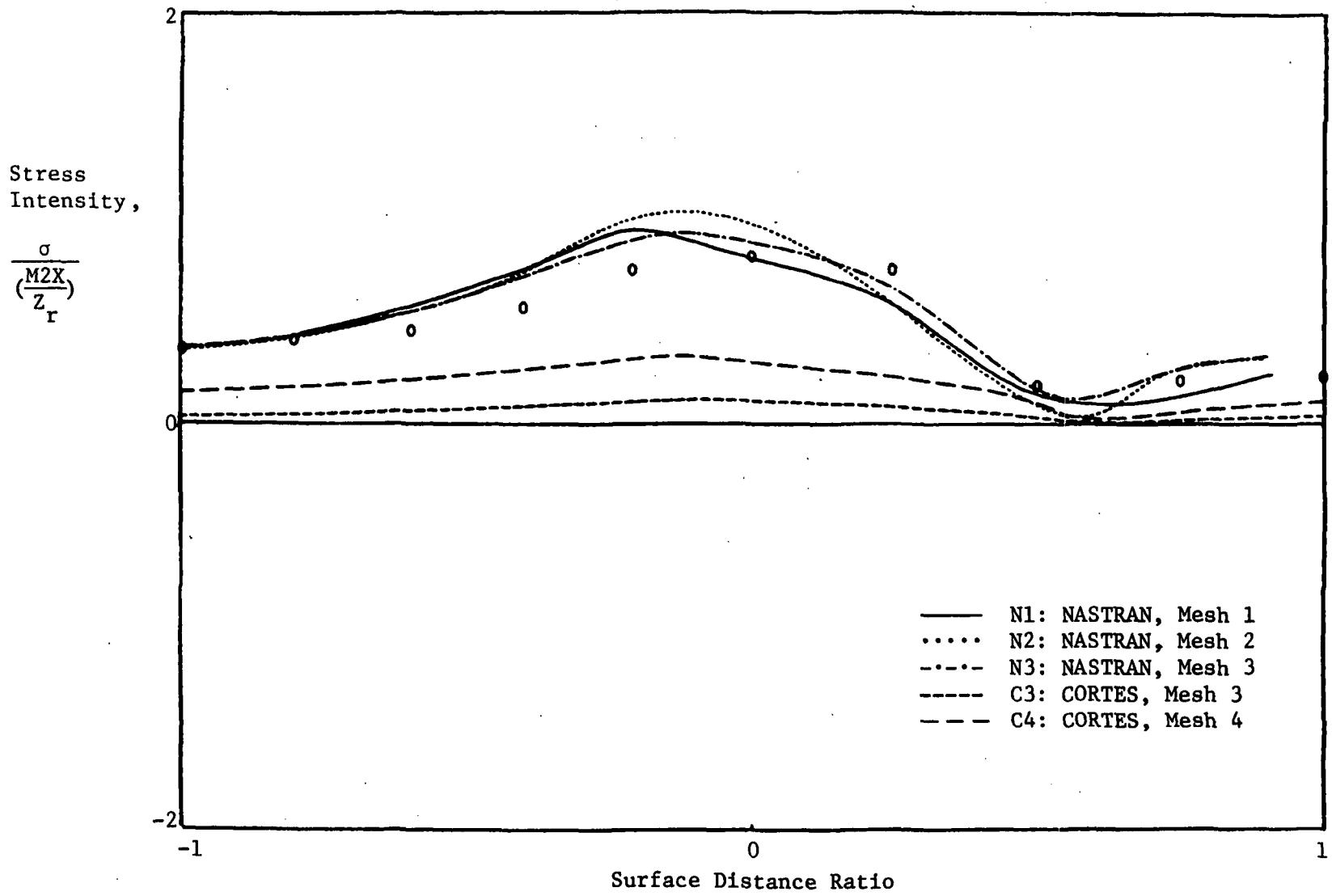


Figure 5 - Normalized Stress Intensity for Load Case 7 (M2X), Row 1, Major Principal Stress on Outer Surface

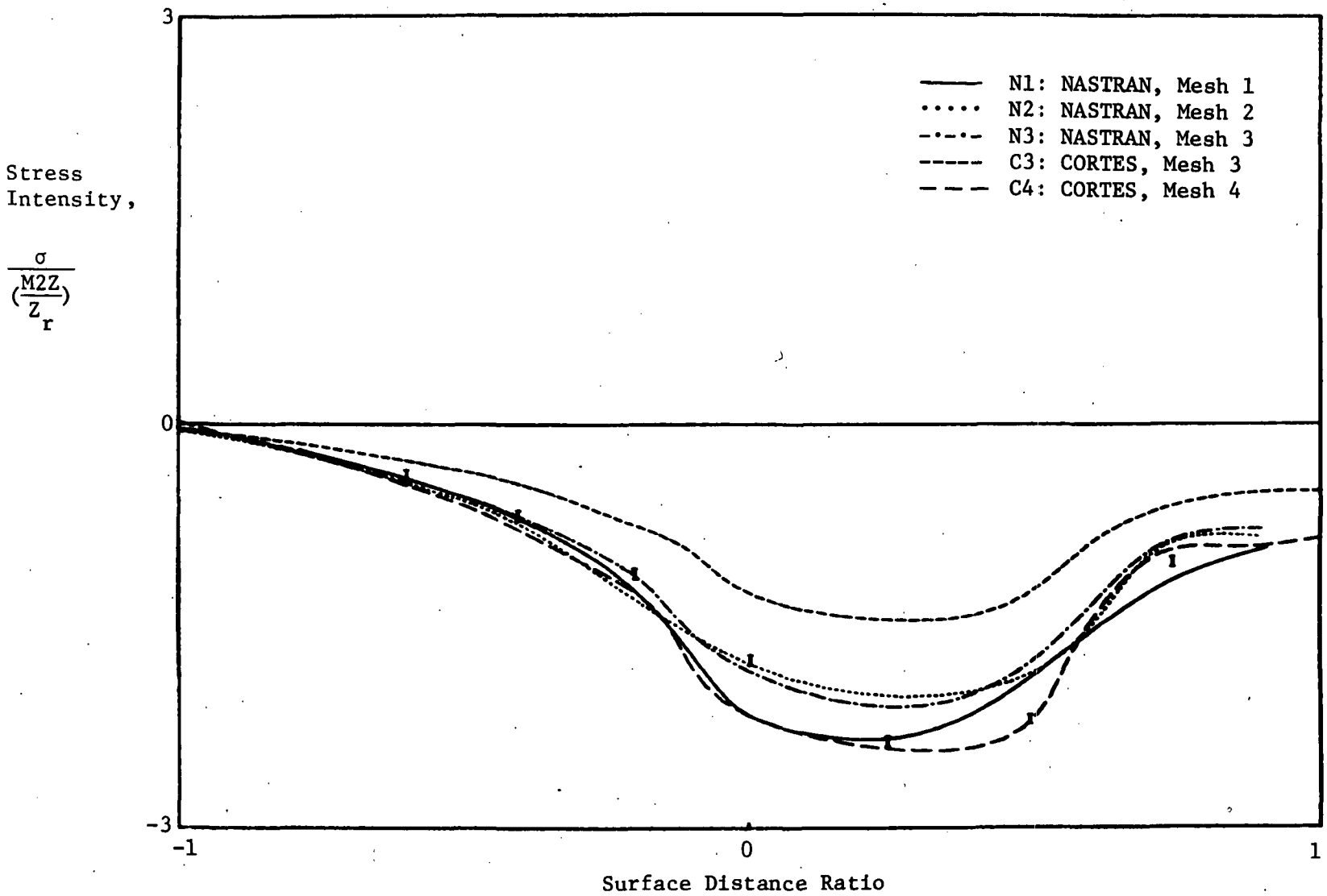


Figure 6 - Normalized Stress Intensity for Load Case 9 (M2Z), Row 1, Minor Principal Stress on Inner Surface

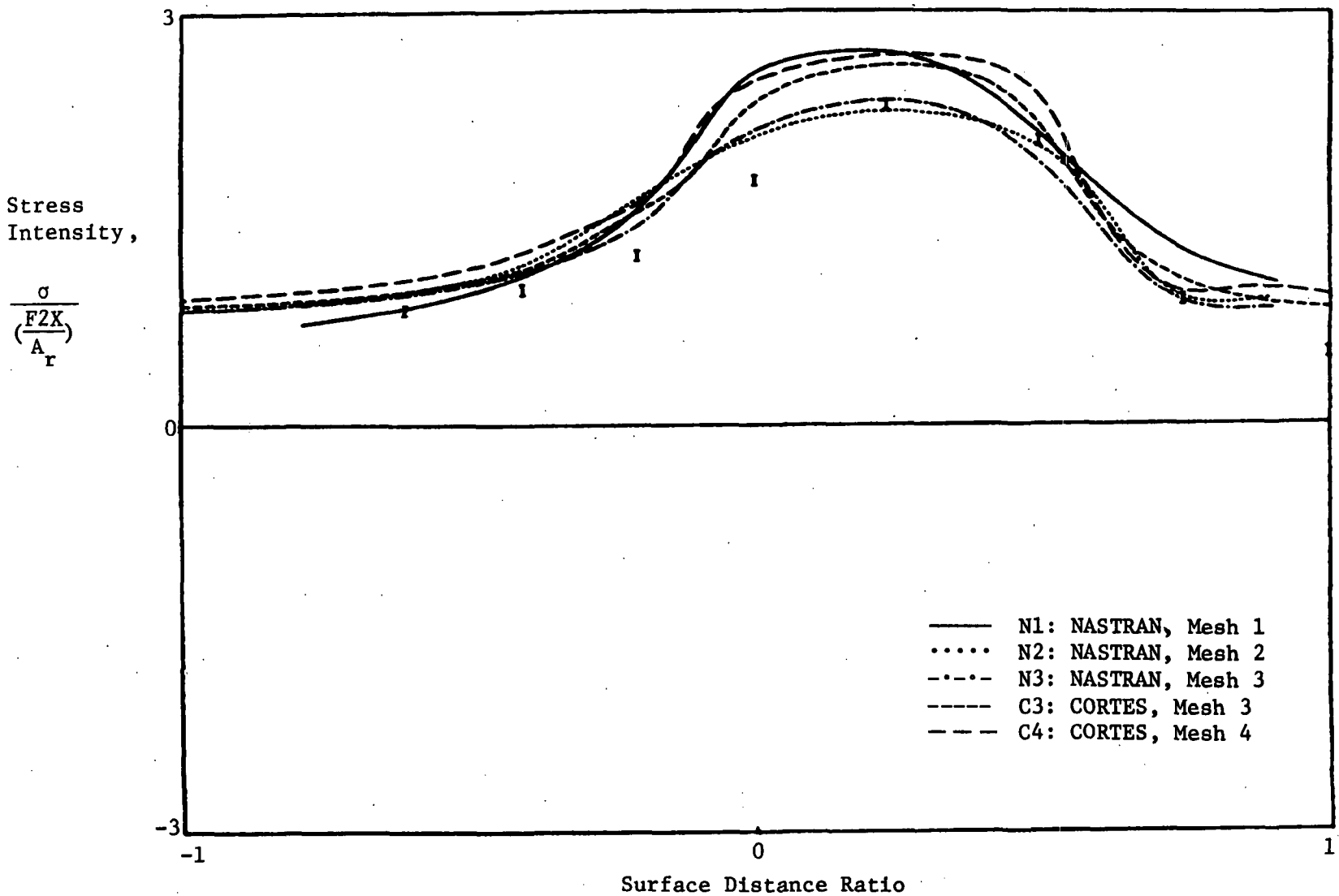


Figure 7 - Normalized Stress Intensity for Load Case 10 (F2X), Row 1, Major Principal Stress on Inner Surface

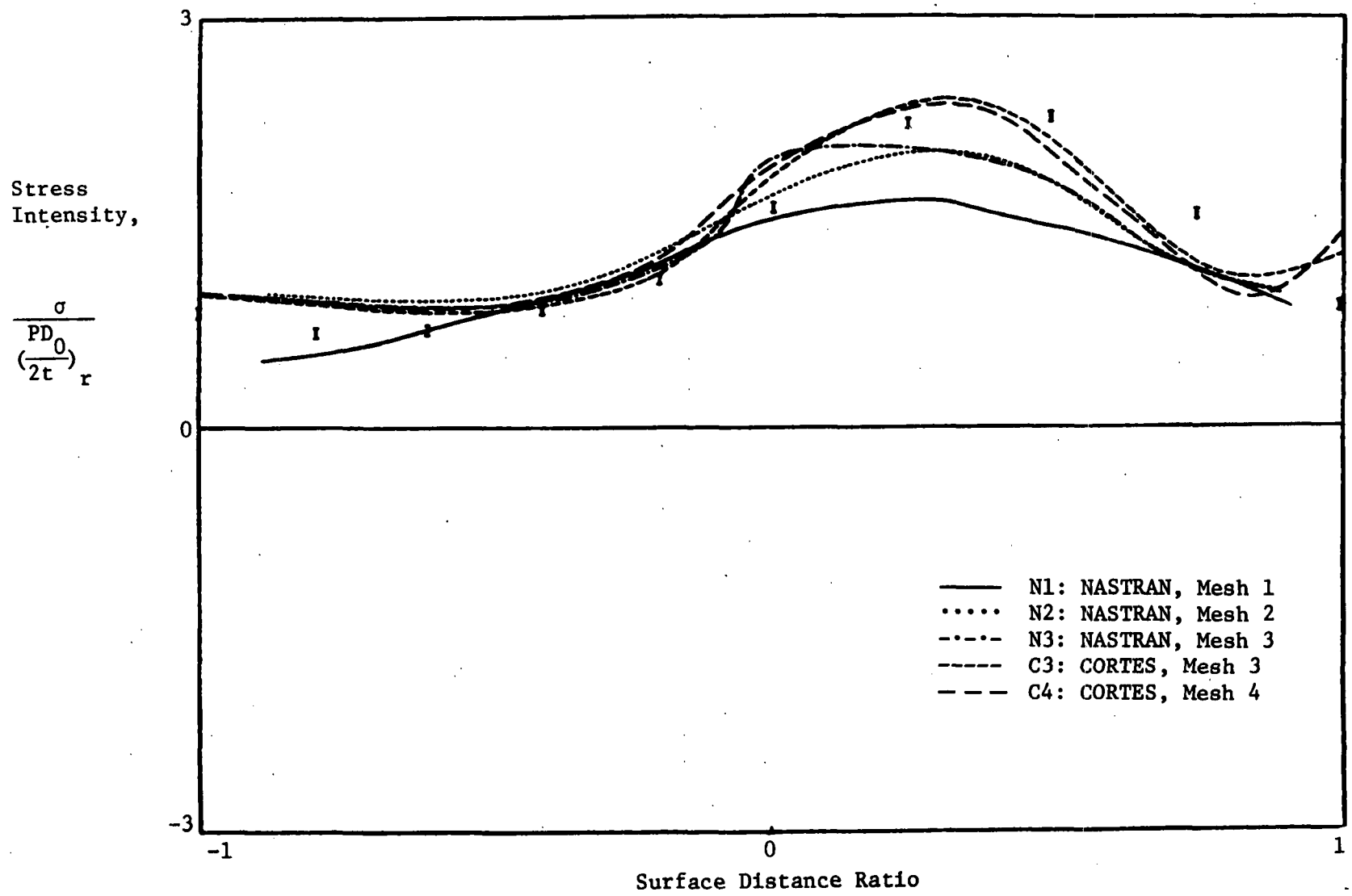


Figure 8 - Normalized Stress Intensity for Load Case 13 (P), Row 6, Major Principal Stress on Inner Surface

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO.	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
NASA CP-2151		

4. TITLE AND SUBTITLE	5. REPORT DATE
Ninth NASTRAN® Users' Colloquium	October 1980
	6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S)	8. PERFORMING ORGANIZATION REPORT #
9. PERFORMING ORGANIZATION NAME AND ADDRESS John F. Kennedy Space Center Kennedy Space Center, Fla. 32899	10. WORK UNIT NO.
	11. CONTRACT OR GRANT NO.
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration Washington, D.C. 20546	13. TYPE OF REPORT & PERIOD COVERED Conference Publication
	14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

16. ABSTRACT

This publication is comprised of papers presented at the Ninth NASTRAN Users' Colloquium held at the NASA John F. Kennedy Space Center October 22-23, 1980. The authors discuss the general application of finite element methodology and the specific application of NASTRAN to a wide variety of static and dynamic structural problems.

17. KEY WORDS	18. DISTRIBUTION STATEMENT
NASTRAN, structures Finite element analysis Substructuring Colloquium	Unclassified — Unlimited Subject Category 39

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	242	A11

For sale by the National Technical Information Service, Springfield, Virginia 22161