

OLD DOMINION UNIVERSITY RESEARCH FOUNDATION

SCHOOL OF ENGINEERING
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA

Technical Report 75-M3

STUDIES IN FINITE ELEMENT ANALYSIS OF COMPOSITE
MATERIAL STRUCTURES

By

Dale O. Douglas

Donna E. Holzmacher

Zoa C. Lane

and

Earl A. Thornton

PRICES SUBJECT TO CHANGE

Final Technical Report

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia

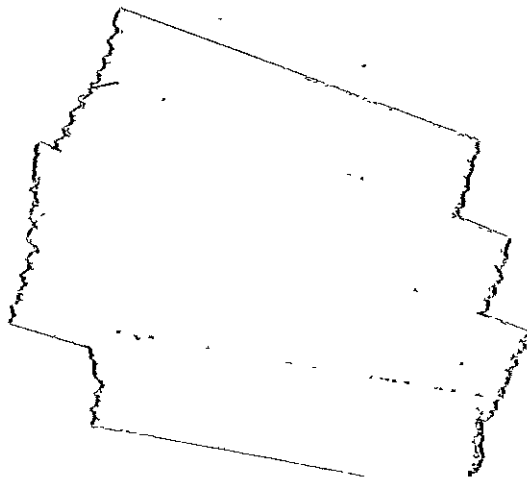
Under
Grant NSG 1043
June 1, 1974 - May 31, 1975

Reproduced by
**NATIONAL TECHNICAL
INFORMATION SERVICE**
US Department of Commerce
Springfield, VA. 22151

September 1975

(NASA-CR-119144) STUDIES OF FINITE ELEMENT
ANALYSIS OF COMPOSITE MATERIAL STRUCTURES
(Old Dominion Univ., Norfolk, Va.) 135 P HC
\$5.75 CSCI 11D
G3/39 42246 Unclas

N75-32508



N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM THE BEST COPY FURNISHED US BY THE SPONSORING AGENCY. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE.

SCHOOL OF ENGINEERING
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA

Technical Report 75-M3

STUDIES IN FINITE ELEMENT ANALYSIS OF COMPOSITE
MATERIAL STRUCTURES

By

Dale O. Douglas

Donna E. Holzmacher

Zoa C. Lane

and

Earl A. Thornton

Final Technical Report

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665

Under
Grant NSG 1043
June 1, 1974 - May 31, 1975
Dr. J.G. Davis, Jr., Technical Monitor
Materials Division
Composites Section



Submitted by the
Old Dominion University Research Foundation
Norfolk, Virginia

September 1975

FOREWORD

This report presents four papers resulting from research conducted under a grant from NASA to the Old Dominion University Research Foundation entitled: "A Research Participation Program for Minority Engineering Students". The three undergraduate engineering students, Dale O. Douglas, Donna E. Holzmacher, and Zoa C. Lane, worked under the direction of Dr. Earl A. Thornton, Associate Professor of Mechanical Engineering and Mechanics.

The student-faculty team began their research in analysis of composite materials at Langley Research Center during a ten-week period in the summer of 1974. The work was continued during the academic year 1974-1975 at Old Dominion University.

Dr. John G. Davis, Jr., of the Composites Section, Materials Application Branch of the Materials Division served as technical monitor for the program. For his cooperation, encouragement, and counsel the authors express their deepest appreciation.

CONTENTS

	<u>Page</u>
 FINITE ELEMENT ANALYSIS OF A PICTURE FRAME SHEAR TEST	
Dale O. Douglas	
Introduction	2
Description of Test	2
Test Specimen	3
Analysis of Shear Test	3
Results and Discussion	6
Concluding Remarks	7
References	9
Figures	10
 BANSAP, A BANDWIDTH REDUCTION PROGRAM FOR SAP IV	
Donna E. Holzmacher	
Introduction	26
Collins Bandwidth Reduction Algorithm	28
SAP IV Preprocessing Program	29
Applications of BANSAP	30
Concluding Remarks	31
Appendix A: User Instructions	32
Appendix B: BANSAP Source Listing	35
Appendix C: Sample BANSAP Output	47
References	53
Tables	54
Figures	57
 FEMESH, A FINITE ELEMENT MESH GENERATION PROGRAM BASED ON ISOPARAMETRIC ZONES	
Zoa C. Lane	
Introduction	65
Interpolation Function Techniques for Finite Element Generation	66
Program FEMESH	67
Applications	68
Concluding Remarks	69
Appendix A: User Instructions for FEMESH	71
Appendix B: FORTRAN Listing of Mesh Generation Program, FEMESH	77
References	91
Tables	92
Figures	99

(cont'd.)

CONTENTS - concluded

Page

FINITE ELEMENT ANALYSIS OF A COMPOSITE BOLTED JOINT
SPECIMEN

Earl A. Thornton

Introduction	109
Bolted Joint Specimen	110
Analytical Procedures	110
Finite Element Model	110
Bolt Loads	112
Results and Discussion	113
Concluding Remarks	115
References	117
Table	118
Figures	119

FINITE ELEMENT ANALYSIS OF A PICTURE FRAME SHEAR TEST

By.

Dale O. Douglas

FINITE ELEMENT ANALYSIS OF A PICTURE FRAME SHEAR TEST

By

Dale O. Douglas

INTRODUCTION

Shear testing of composite materials is generally concerned with two principal areas of interest: (1) to determine the in-plane shear properties, or (2) to determine the interlaminar or normal shear properties. In-plane shear properties of a laminate are among the most difficult to determine because of problems in applying a state of uniform shearing stress. Concepts for determining in-plane shear properties include torsion tube tests, rail shear tests, and picture frame shear tests

The most direct method of applying pure shear is by torsion of a tube. This test method has proven to be a reliable means of determining in-plane shear properties (ref. 1). However fabrication techniques for high quality $\pm 45^\circ$ metal matrix composite tubes have not yet been established. The difficulty of fabricating high quality tubes has stimulated research in other methods of shear testing.

Another type of shear test is the rail shear test. It uses a thin laminate, loaded along its length by two pairs of rails, leaving an unsupported central test section.

In the present study an analysis of a picture frame shear test performed at Langley Research Center is presented. The purposes of the study were to determine the stress distributions in the picture frame shear test specimen and to determine the effect of local reinforcements on the stress distributions.

DESCRIPTION OF TEST

The experimental setup for a picture frame shear test is shown in figure 1. The picture frame shear test was used to

produce in-plane shear stress in the test panel. The shear panel was bonded to a frame constructed from four 1 in. x 1 in. steel edge bars designed to simulate fully clamped edge conditions. The panel specimen was bolted to a test frame by 0.375-in.-diameter bolts, seven per side. At each corner of the test frame, loads were applied to the pin joints by the testing machine. Tensile loads were applied to the vertical pins, and compressive loads were applied to the horizontal pins to produce shear loading in the test specimen.

TEST SPECIMEN

The test specimens were made using 7 in. x 7 in. borsic aluminum sandwich shear panels. With the addition of 1 in. x 1 in. steel edge bars, the overall dimensions of the shear panel specimen were 9 in. x 9 in. with a nominal thickness of 1 in. To permit installation of the pins on the test frame, a portion of the shear panel was cut away at each corner. Each corner had a radius of 0.25 in. The test specimen is shown schematically in figure 2.

The sandwich panel consisted of two face sheets separated by a honeycomb core. On each face sheet there were four plies (0.0285 in. thick) at a $\pm 45^\circ$ layup. The panel face sheets were cut from 10-in.-square laminates. The filaments of the laminate were parallel to the applied loads. Some of the test specimens were reinforced with titanium doublers (0.060 in. thick) in the vicinity of the corner radii.

ANALYSIS OF SHEAR TEST

Finite element analyses were made to determine the in-plane stress distributions in the shear panel. The finite models represented the shear panel specimens using orthotropic, two-dimensional plane stress elements. Two general purpose finite element computer programs were utilized in the analysis of the shear panel. The first was NASTRAN (NASA Structural Analysis Program) which was

used on the CDC-6600 computer at Langley Research Center. NASTRAN (ref. 2) is a general purpose digital computer program for the analysis of large complex structures. The second program, SAP IV (Structural Analysis Program), was executed on an IBM-370, Model 158 computer at Virginia Polytechnic Institute & State University through the computer center at Old Dominion University. SAP IV (ref. 3) is a structural analysis program for static and dynamic response of linear systems. Symmetry of loading, geometry, and material properties made the analysis of only one quarter of the specimen sufficient.

NASTRAN embodied a finite element approach, wherein the distributed physical properties of the shear specimen were represented by a model (fig. 3) consisting of 490 membrane elements that were interconnected at 529 grid points. The grid point definition formed the basic framework for the structural model. All other parts of the structural model were referenced either directly or indirectly to the grid points. Each grid point had two degrees of freedom, the in-plane displacements. The elements used in the analysis were the quadrilateral membrane element CQDMEM and the triangular membrane element CTRMEM.

The steel edge bars of the test specimen were represented in NASTRAN as rigid boundaries. The rigid boundaries were modeled using multipoint constraints in the NASTRAN program. The constraints were applied to grid points on the test frame edge of the finite element model so that these points deformed as a straight line. Static loads were applied to the structural model through nodes constrained to the rigid boundary.

The loads were from Langley Research Center Test 560, Run 7; a horizontal load of 5004.9 lb and a vertical load of 5039.4 lb are shown in figure 3 at the points of application.

SAP embodied a finite element approach where the shear specimen was represented by a model (fig. 4) consisting of 554 membrane elements that were interconnected at 595 nodal points.

The steel edge bars of the test specimen were represented in SAP as deformable boundaries. The deformable boundaries were simulated by the addition of 64 plane stress membrane elements to the NASTRAN model. The horizontal and vertical applied loads were represented by statically equivalent loads applied along the simulated boundary. Nine colinear loads were applied at nodal points nearest the center of each bolt hole. These loads were applied at an angle of 45 degrees. The magnitudes of these applied loads are given in figure 4. Stresses were computed at the centroid of each element using the stress print option available in SAP.

The titanium doublers used for local reinforcement at corner radii were modeled with an addition of 21 finite elements on existing elements at the extreme corner of the sandwich panel. The material elasticity matrix for titanium and borsic aluminum is given in table 1.

Table 1. Material elasticity matrix.

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} \\ G_{12} & G_{22} & G_{23} \\ G_{13} & G_{23} & G_{33} \end{bmatrix} \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix}, \text{ psi}$$

	G_{11}	G_{12}	G_{13}	G_{22}	G_{23}	G_{33}
Borsic Aluminum	2.81E+7	5.65E+6	0	2.81E+7	0	9.5E+6
Titanium	1.81E+7	6.15E+6	0	1.81E+7	0	6.15E+7

The NASTRAN finite element model of the shear panel, simulating a rigid boundary, had 1000 degrees of freedom. Using a CDC-6400 computer, it took 945 CPU seconds for the program to execute. In contrast to the NASTRAN model, the SAP finite model

had 1132 degrees of freedom with a bandwidth of 1106. Due to the excessive storage required by the large bandwidth, the SAP finite element program was unable to execute. To optimize the bandwidth, the nodes were renumbered using a computer program, BANSAP. With this renumbering, the SAP program had a final bandwidth of 69. It then completed execution in 160 CPU seconds.

RESULTS AND DISCUSSION

The stresses computed in the shear panel for the loads applied to the rigid (NASTRAN) and deformable (SAP) boundary models are given in figures 5 through 8. Normal stresses σ_x and σ_y are plotted as ordinates with the horizontal and vertical coordinates from the center of the shear panel as abscissae.

The stress distributions along the horizontal and vertical axes of both the rigid and deformable boundary models are uniform near the center of the specimen. The uniform stress values differ considerably between the two models; the uniform normal stresses predicted by the rigid boundary model are nearly three times the stresses predicted assuming a deformable boundary. These results indicate that the assumption of a rigid boundary should not be made.

There is an appreciable stress concentration at the corner fillets. The stress components perpendicular to the lines of symmetry rise sharply at the corners. For example, figure 7 shows that in the deformable boundary model the stress component σ_y increases from 10 ksi to about 105 ksi indicating a stress concentration factor of over 10.

Contour plots of the principal shearing stresses for the rigid and deformable boundary models are shown in figures 9 and 10. The shearing stresses are uniform only over a small portion of the specimen. Figure 10 shows that the shearing stress may vary by as much as 25 percent over the center one-half of the specimen.

The effect of the reinforcing titanium doubler on the normal stresses σ_x and σ_y is shown in figures 11 through 14. These results indicate that the doubler significantly reduced the stresses along the x-axis near the fillet. The critical stress σ_y on this axis was reduced by about one-half. However, stress distributions along the vertical axis and in the center of the shear panel show no reinforcing effects of the titanium doubler.

The contours of the principal shearing stress in the specimen with the titanium doubler are shown in figure 15. By comparing this figure with figure 10 it can be seen that the doubler tended to reduce the region of nearly uniform shearing stress since the contours in figure 15 are closer to the center of the panel. As expected there is also an appreciable local disturbance in the shearing stresses in the vicinity of the doubler.

CONCLUDING REMARKS

Two finite element analyses of a picture frame shear test of a basic aluminum test specimen have been performed. Two methods for modeling the specimen test frame have been investigated. Results for nominal stresses and principal shear stress have been presented for Test 560, Run 7 conducted at Langley Research Center.

There were striking differences in the stress distributions predicted by the rigid (NASTRAN) and deformable (SAP) boundary models. It was found that it is not realistic to assume the test fixture to be a rigid frame. In the regions of nearly uniform stress, the stresses predicted by the deformable boundary models were approximately one third of the stresses predicted by the rigid boundary model. In the vicinity of the corner, the stresses predicted by the two models nearly coincided.

The constant principal shear stress, τ_{max} was uniform over only a very small region in the center of the shear panel specimen. Moreover, at the corners near the fillets, there were steep gradients with stresses being highly concentrated.

The effect of a local reinforcing titanium doubler has been evaluated. It was found that the doubler reduced the maximum nominal stress in the vicinity of the fillet by about 50 percent.

REFERENCES

1. "Advanced Composites Design Guide", Air Force Materials Laboratory, F33615-71-C-1362 (USAF, dated January 1973), Wright-Patterson Air Force Base, Ohio 45433.
2. McCormick, C.W., Ed., The NASTRAN Users' Manuals, NASA, June 1972.
3. Wilson, E.L., Peterson, F.E., and Bathe, K., SAP IV: A Structural Analysis Program for Static and Dynamic Response of Linear Systems, Report No. EERC 73-11, Engineering Analysis Corporation, Berkeley, California, June 1973.

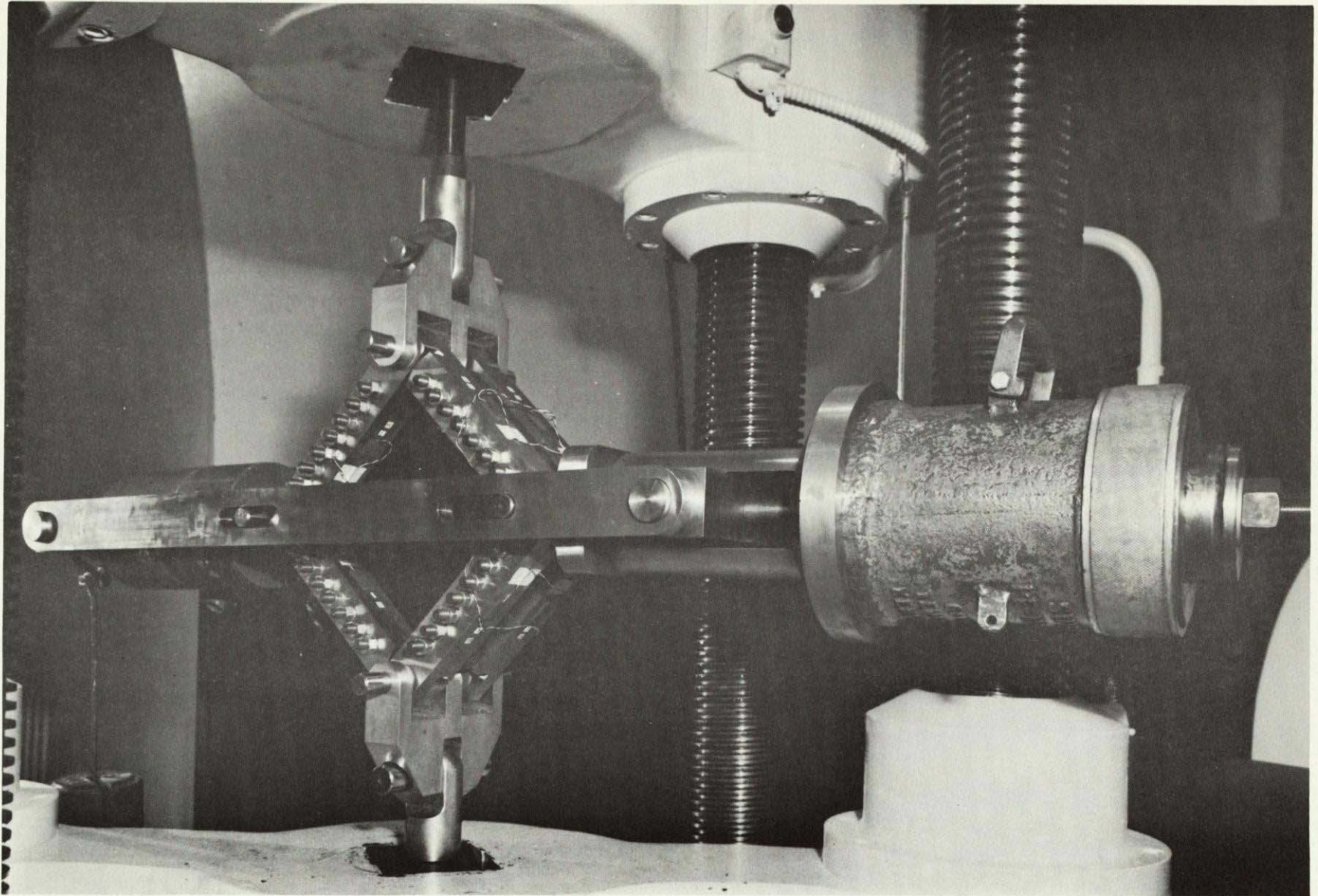


Figure 1. Picture Frame Shear Test Experimental Setup at Langley Research Center.

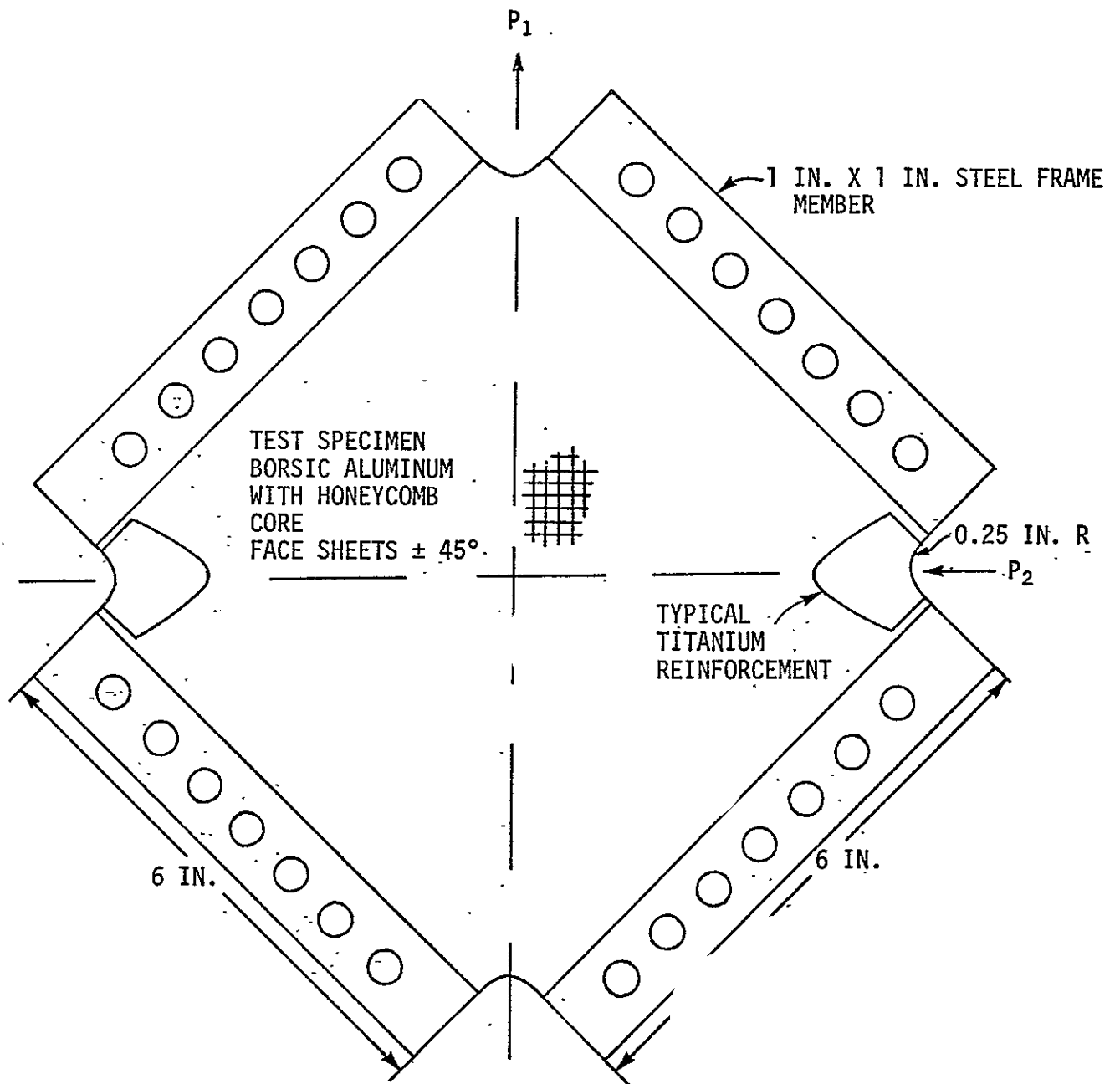


Figure 2. Schematic of Test Specimen.

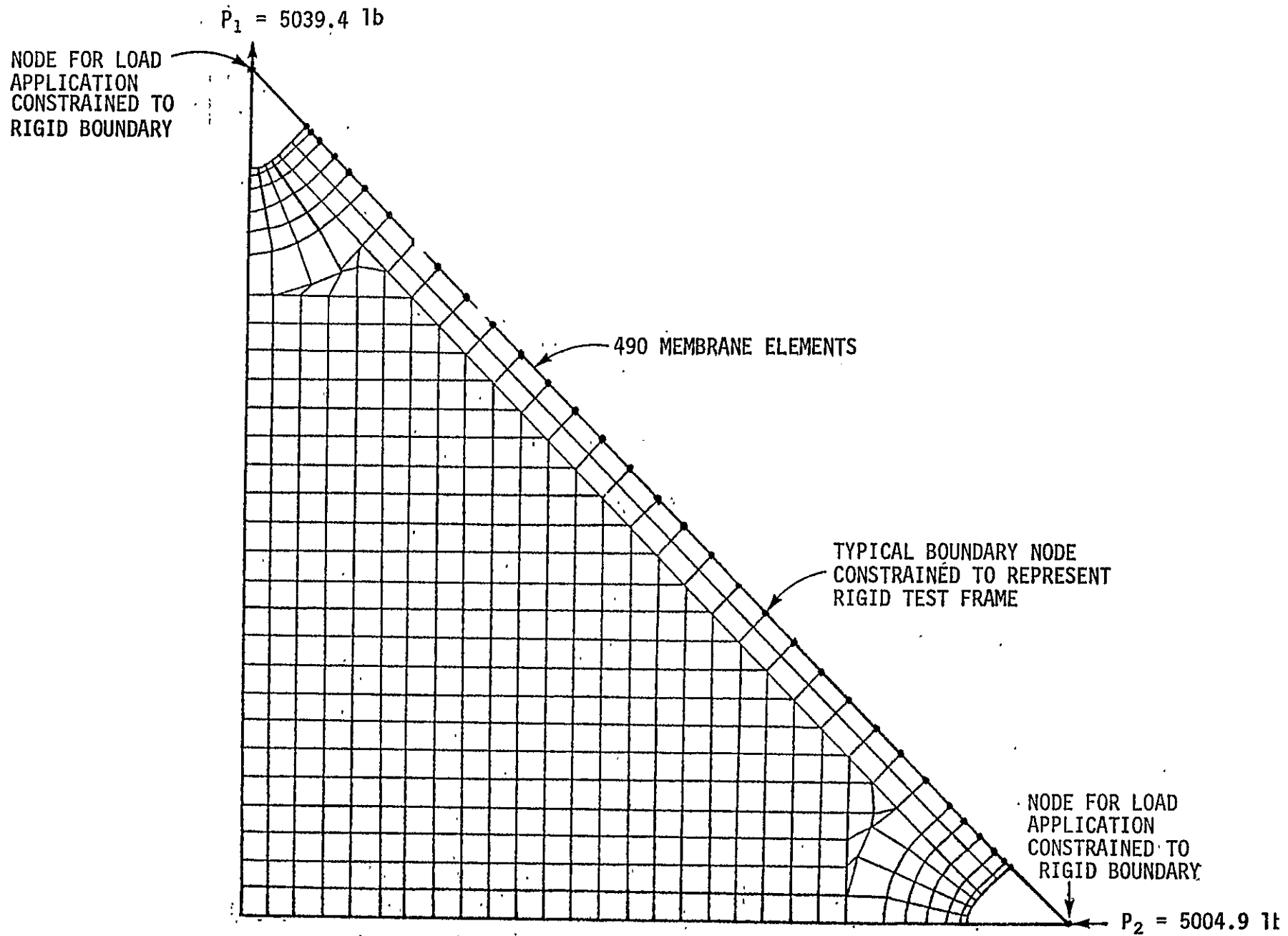


Figure 3. NASTRAN Finite Element Model of Shear Panel with Rigid Boundary.

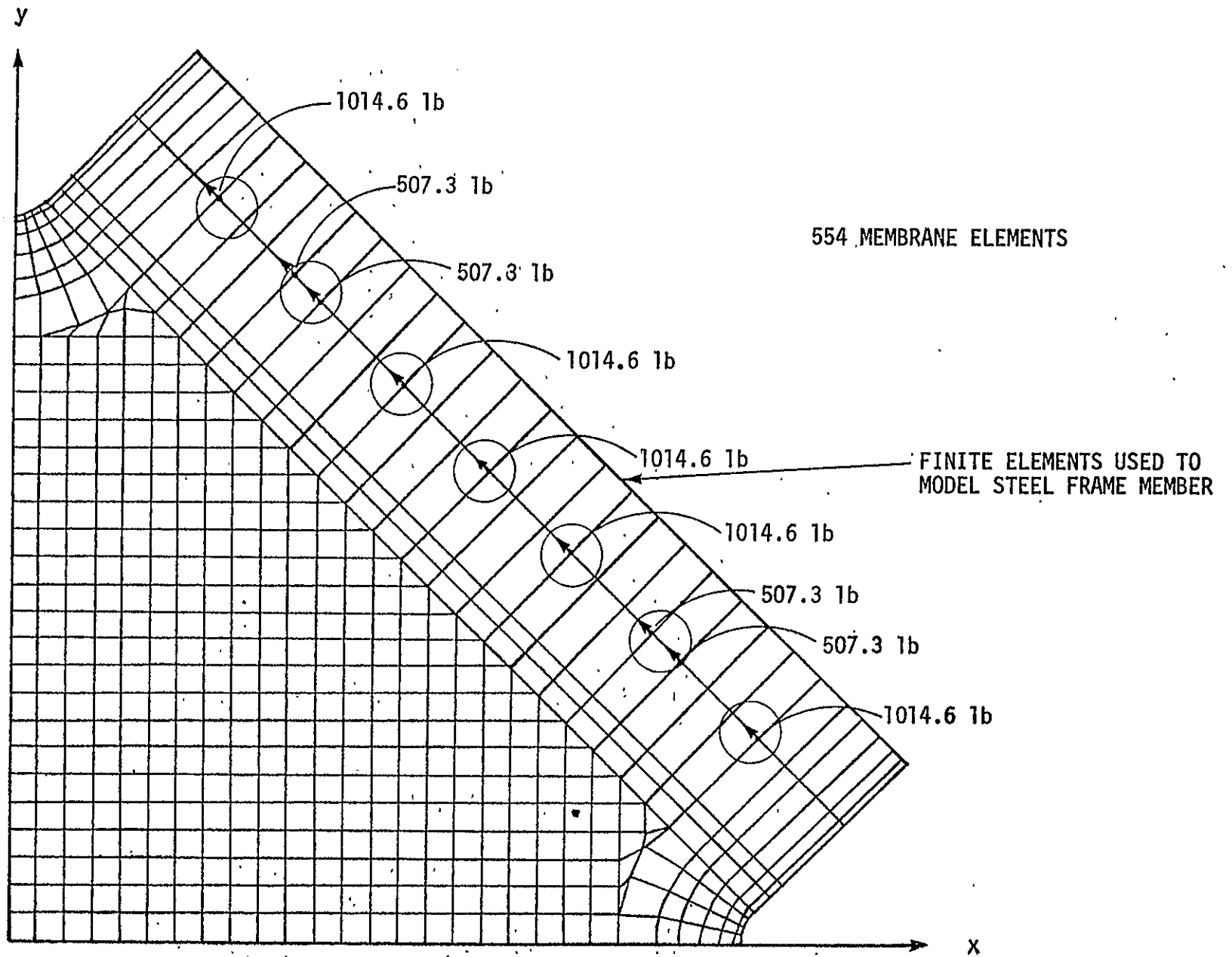


Figure 4. SAP Finite Element Model of Shear Panel (Deformable Boundary).

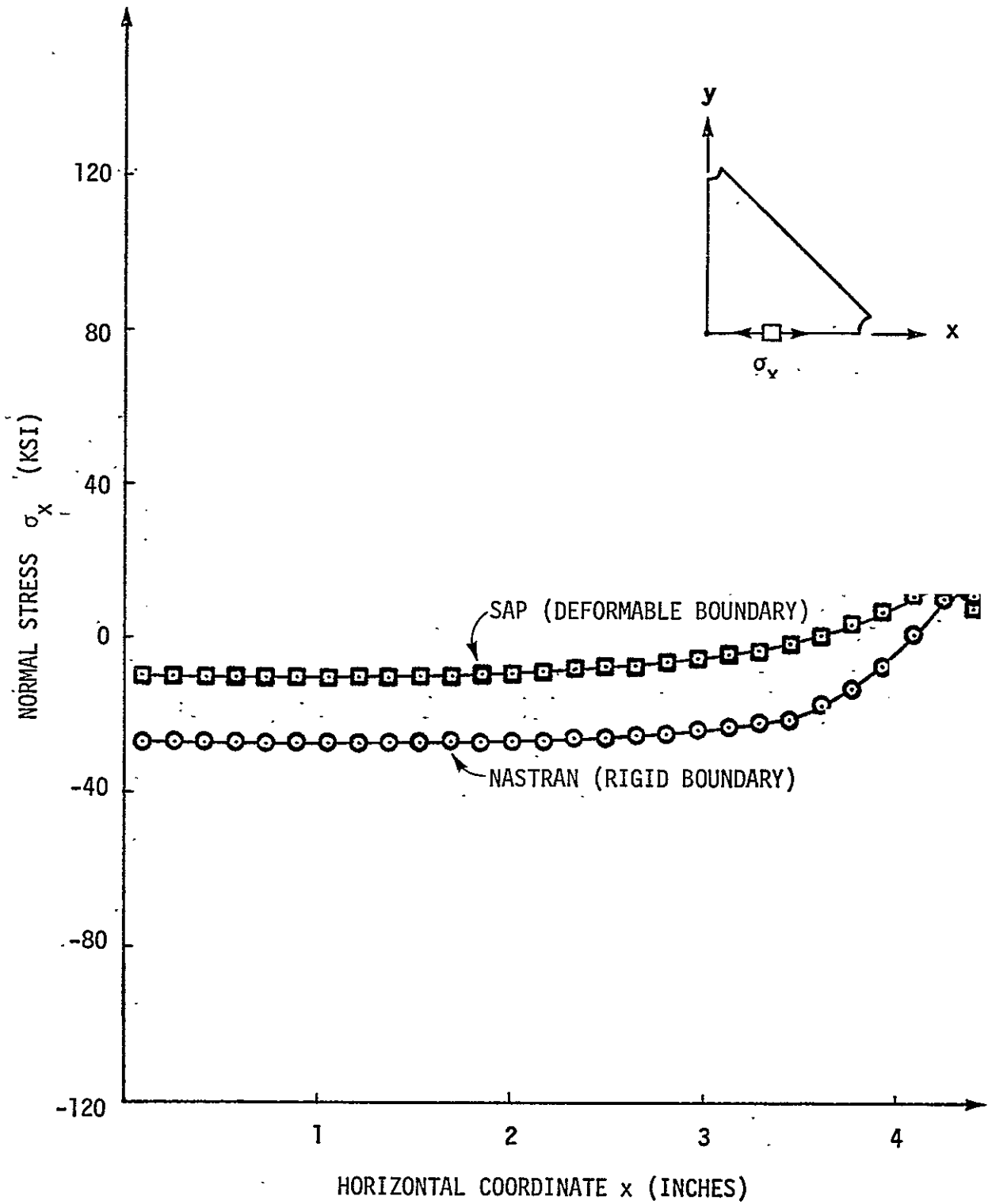


Figure 5. Normal Stress σ_x as a Function of Horizontal Coordinate Along Center Line of Shear Panel Specimen

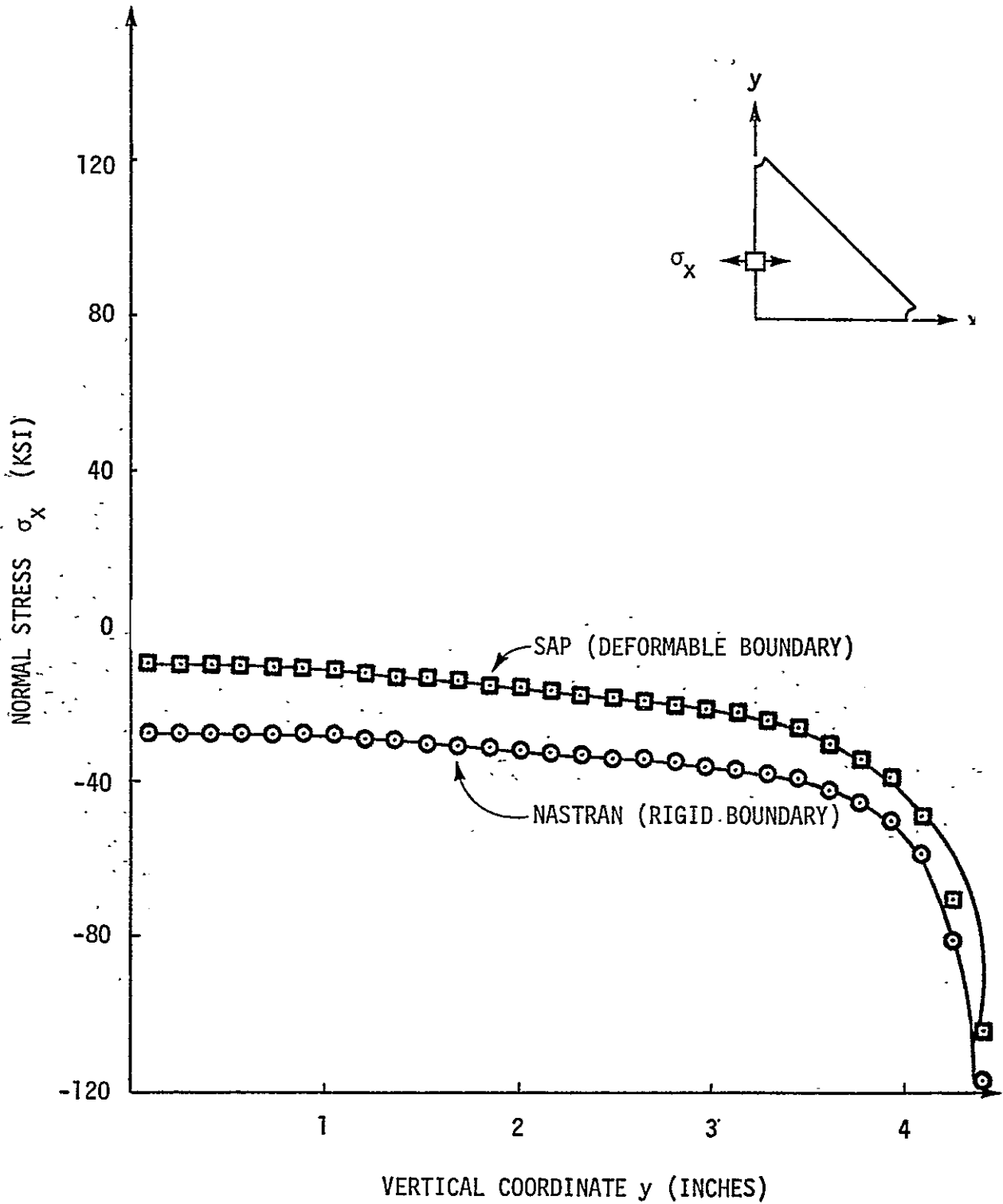


Figure 6. Normal Stress σ_x as a Function of Vertical Coordinate Along^xCenter Line of Shear Panel Specimen.

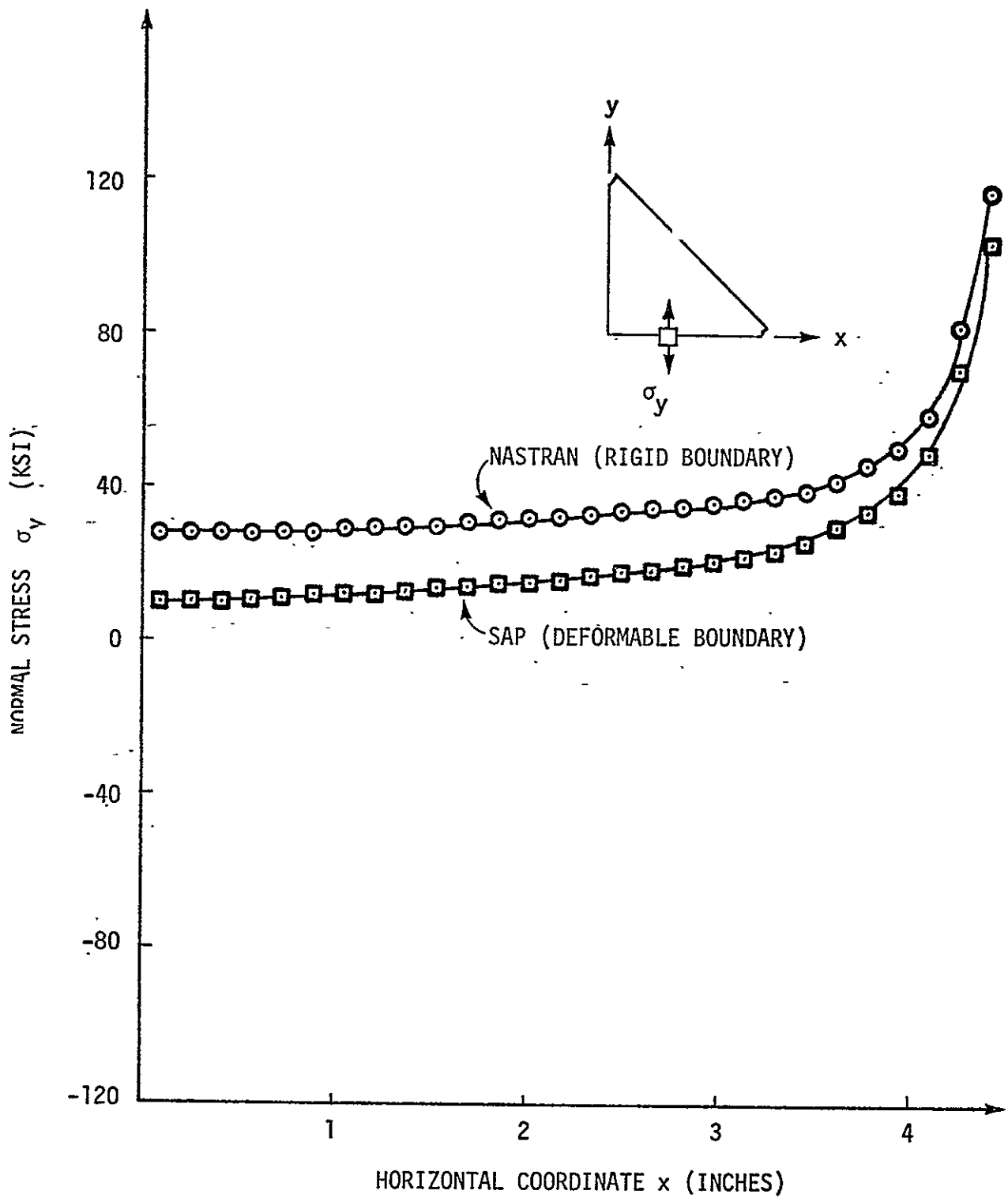


Figure 7. Normal Stress σ_y as a Function of Horizontal Coordinate Along y Center Line of Shear Panel Specimen.

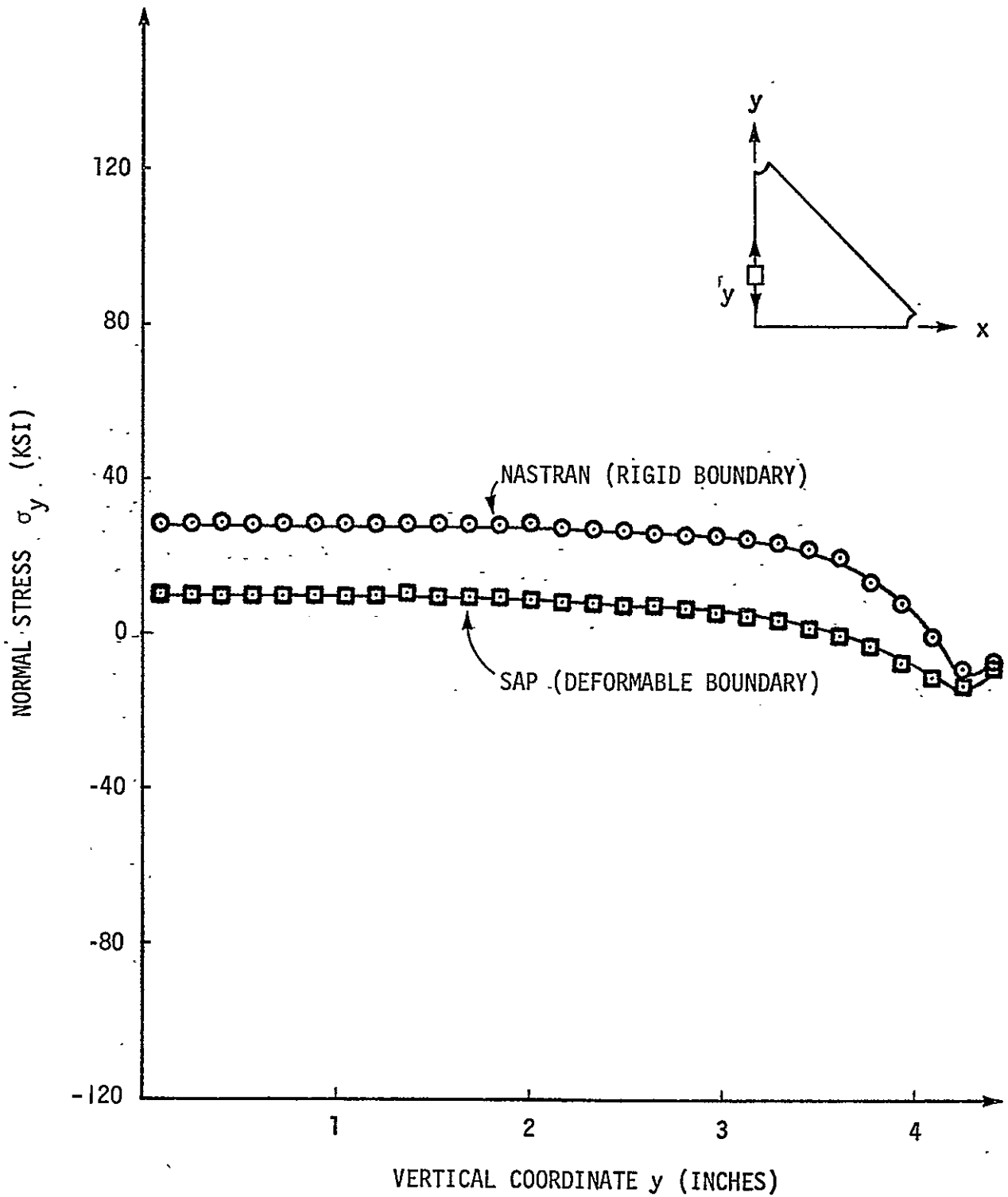


Figure 8. Normal Stress σ_y as a Function of Vertical Coordinate Along Center Line of Shear Panel Specimen.

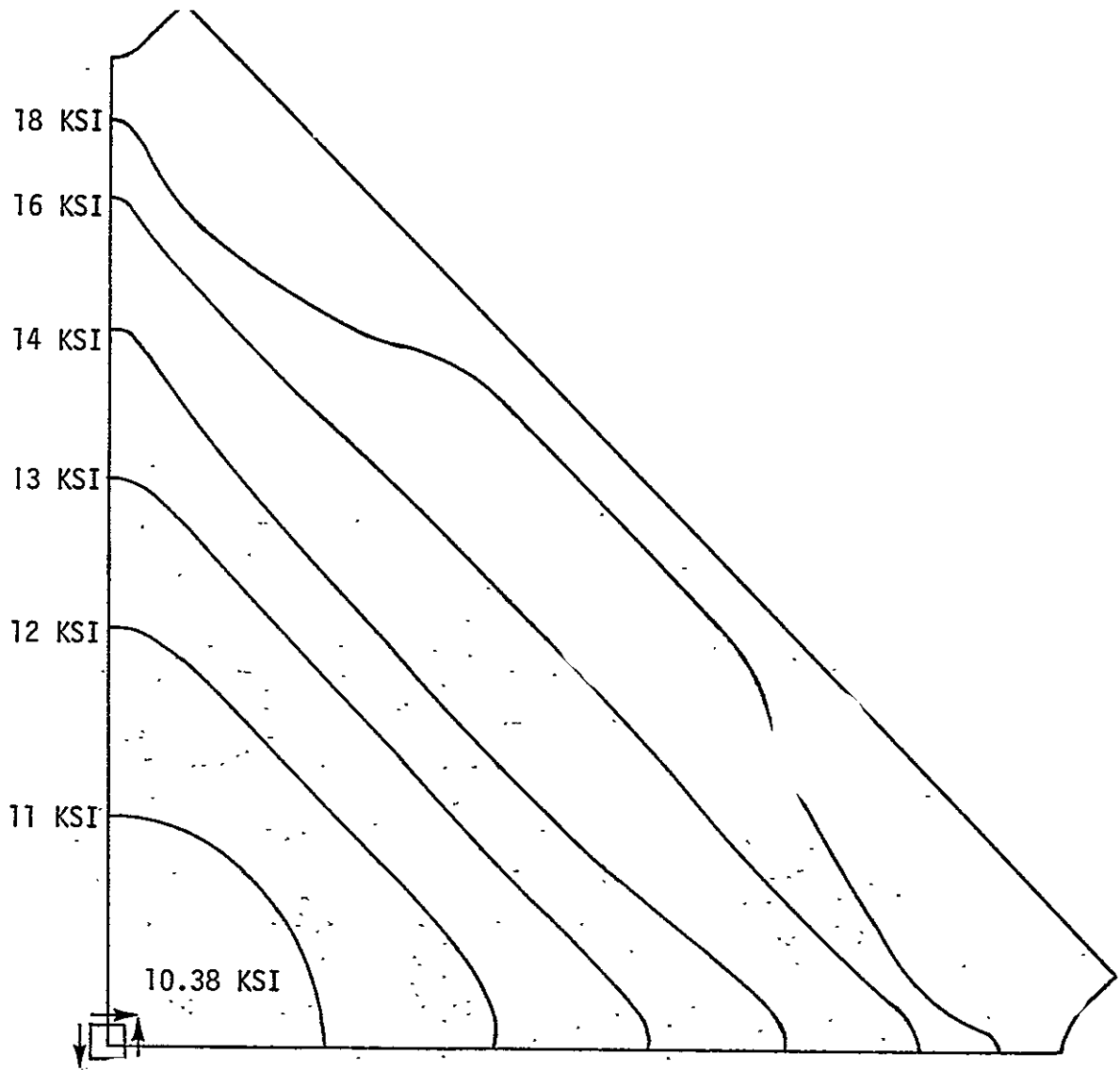


Figure 9. Contours of Constant Principal Shear Stress, τ_{max} , Predicted by NASTRAN (Rigid Boundary).

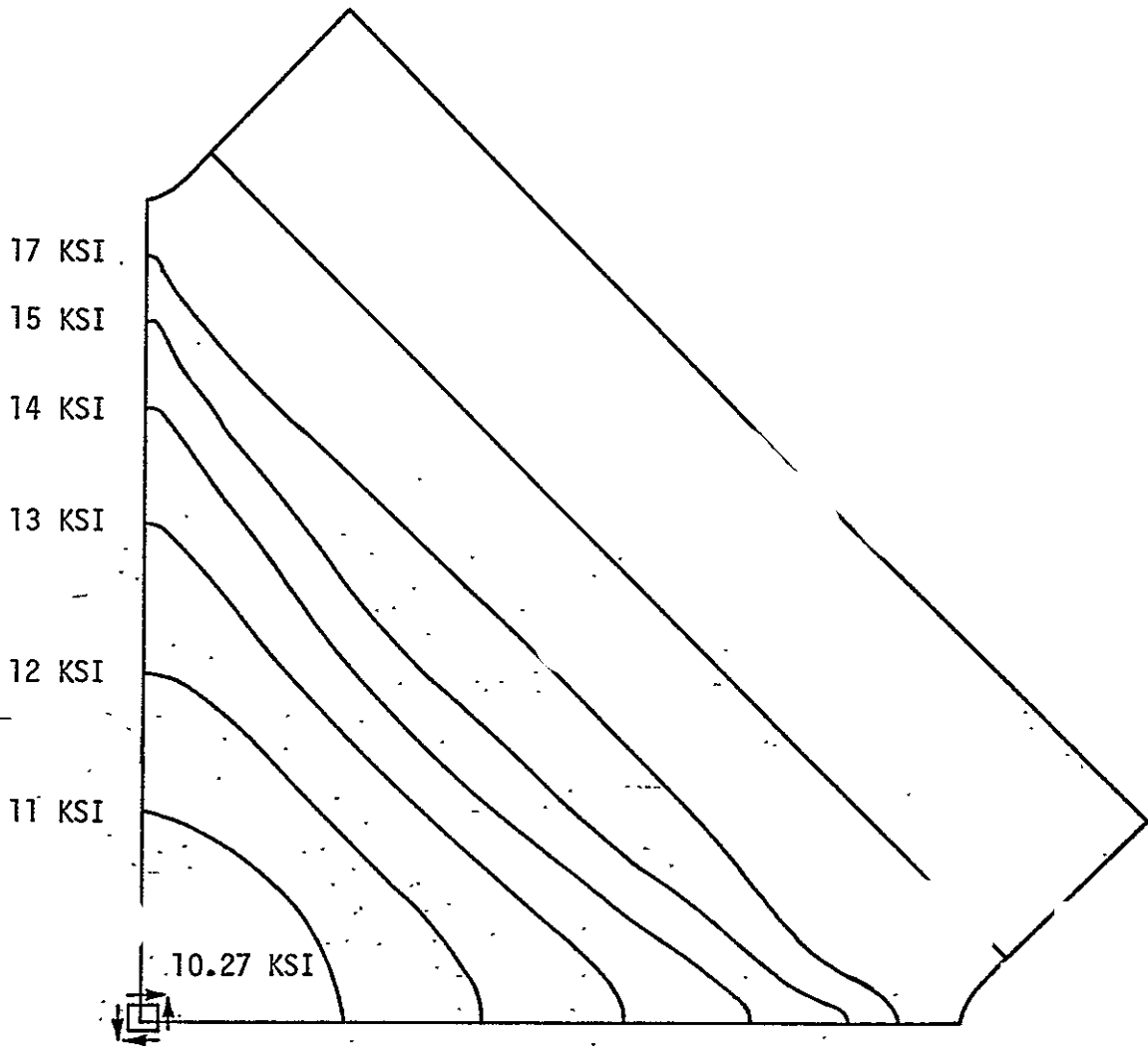


Figure 10. Contours of Constant Principal Shear Stress, τ_{max} , Predicted by SAP. (Deformable Boundary).

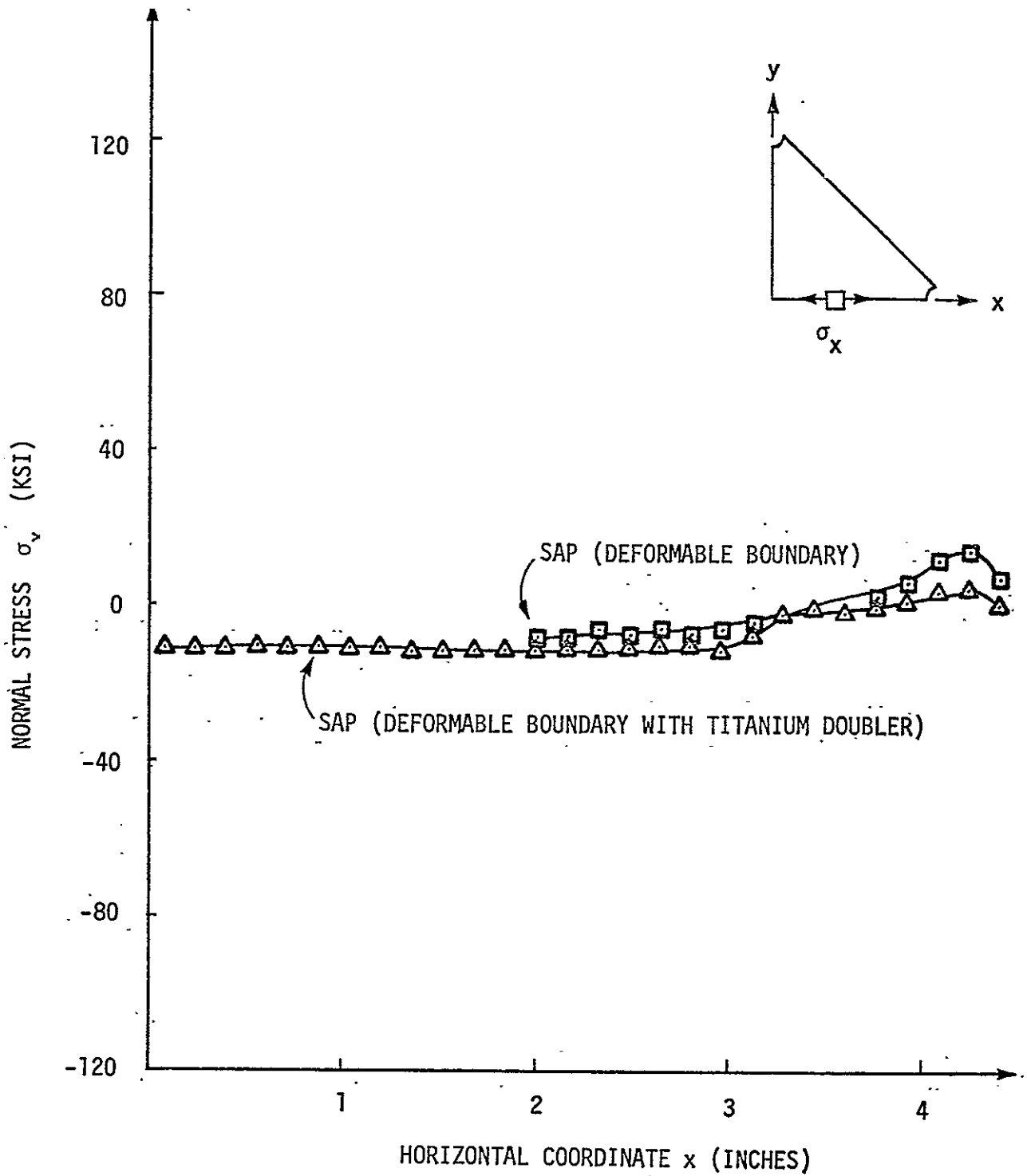


Figure 11. Normal Stress σ_v as a Function of Horizontal Coordinate Along^x Center Line of Shear Panel Specimen.

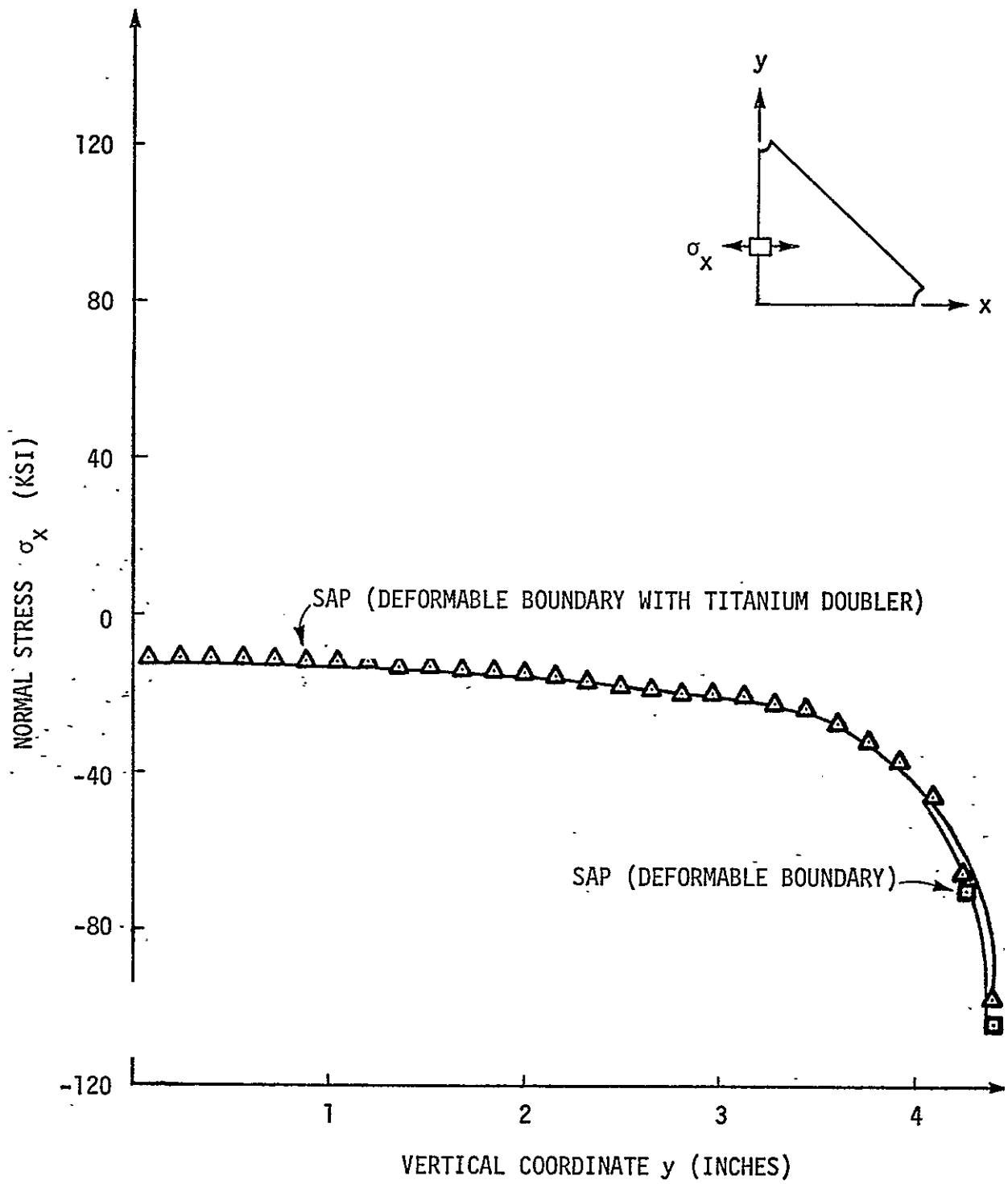


Figure 12. Normal Stress σ_x as a Function of Vertical Coordinate Along^xCenter Line of Shear Panel Specimen.

the centroidal coordinate of the finite elements bordering the hole. Comparison of figures 6 through 8 and figure 10 shows that qualitatively the finite element analysis of the anisotropic composite and the isotropic elasticity solution are in close agreement. This agreement serves to validate the finite element solution.

The variation of the longitudinal membrane force in an isotropic infinite medium is shown in figure 11 in terms of the x coordinate of the composite specimen to facilitate comparison with the finite element solution given in figure 9. The elasticity solution shows an extremely sharp gradient for the membrane force in the vicinity of the hole. This sharp variation raises questions about the accuracy of the finite element solution in this region. Since the NASTRAN finite element assumed constant stress within the element, it is possible that the peak stress was underestimated because not enough elements were used to accurately represent the stress gradient. The variation of the stress away from the hole according to the isotropic solution shows that in a distance of about five radii ($5a = 0.48$ in.) away from the hole the force has decreased to one-tenth of its maximum value. This result supports the findings of figures 6 through 8 in which the membrane force distributions in the center and outside holes were very nearly the same. This occurred because there were no hole interaction effects since the holes were more than five radii apart. Only very small edge effects were present for the same reason.

CONCLUDING REMARKS

A finite element analysis of an extra graphitic reinforced bolted joint specimen has been performed. Two methods were used to represent bolt transfer loads. The first method assumed a perfect fit and modeled the bolt loading as a cosine distribution over one-half of the boundary of the hole. The second method assumed an imperfect fit and used a nonlinear computer analysis to determine the contact area and bolt transfer loads. The

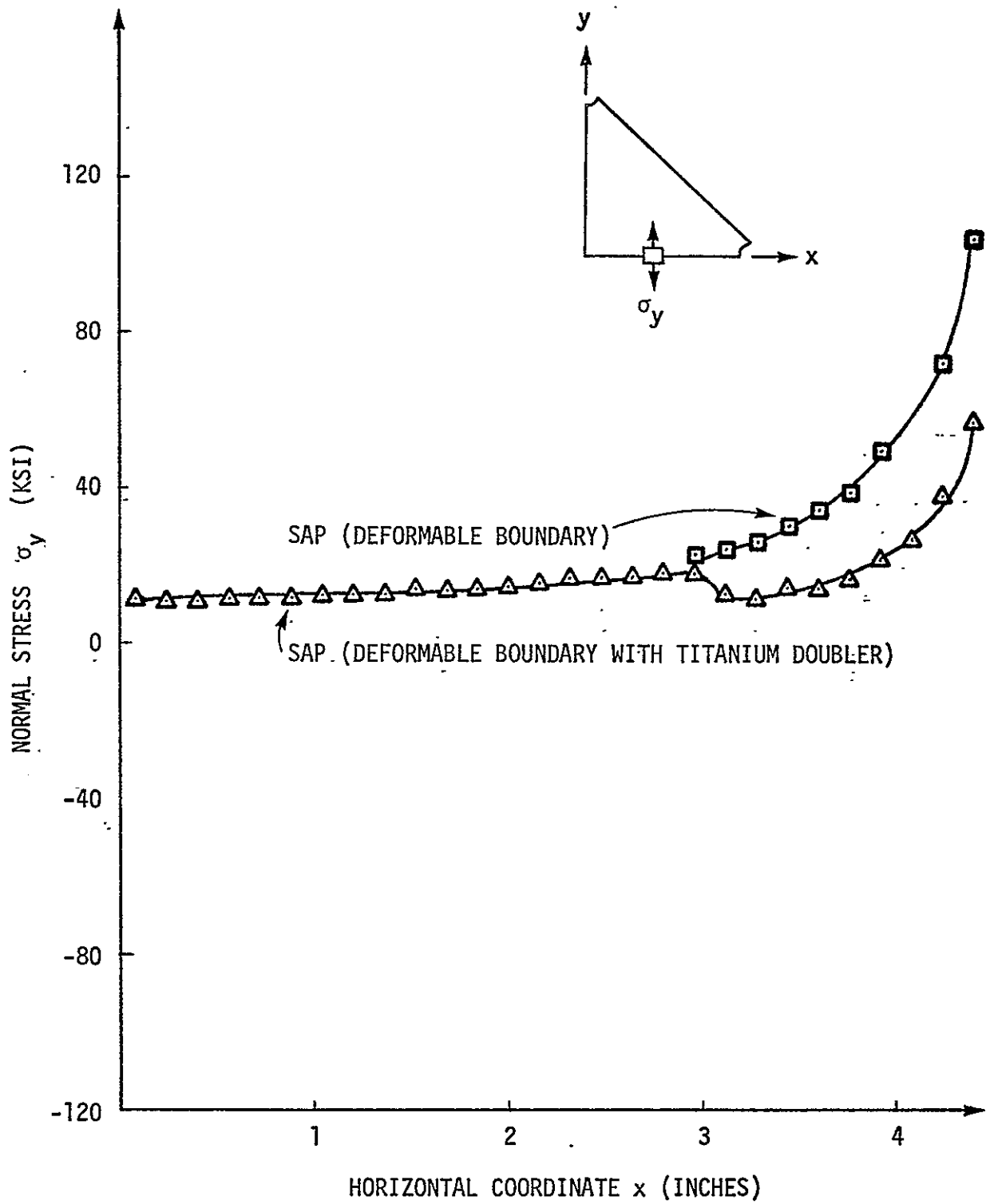


Figure 13. Normal Stress σ_y as a Function of Horizontal Coordinate Along y Center Line of Shear Panel Specimen.

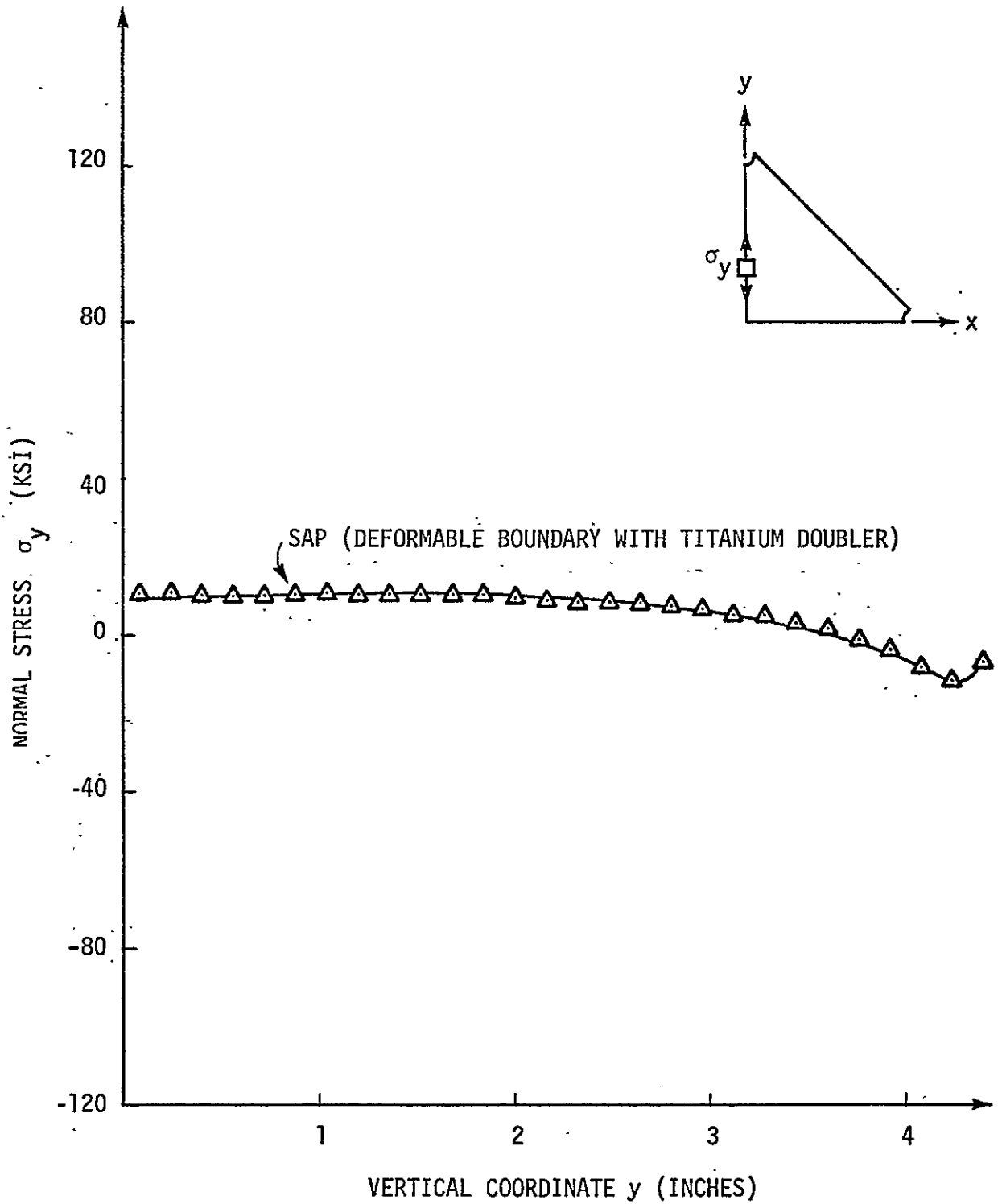


Figure 14. Normal Stress σ_y as a Function of Vertical Coordinate Along Center Line of Shear Panel Specimen.

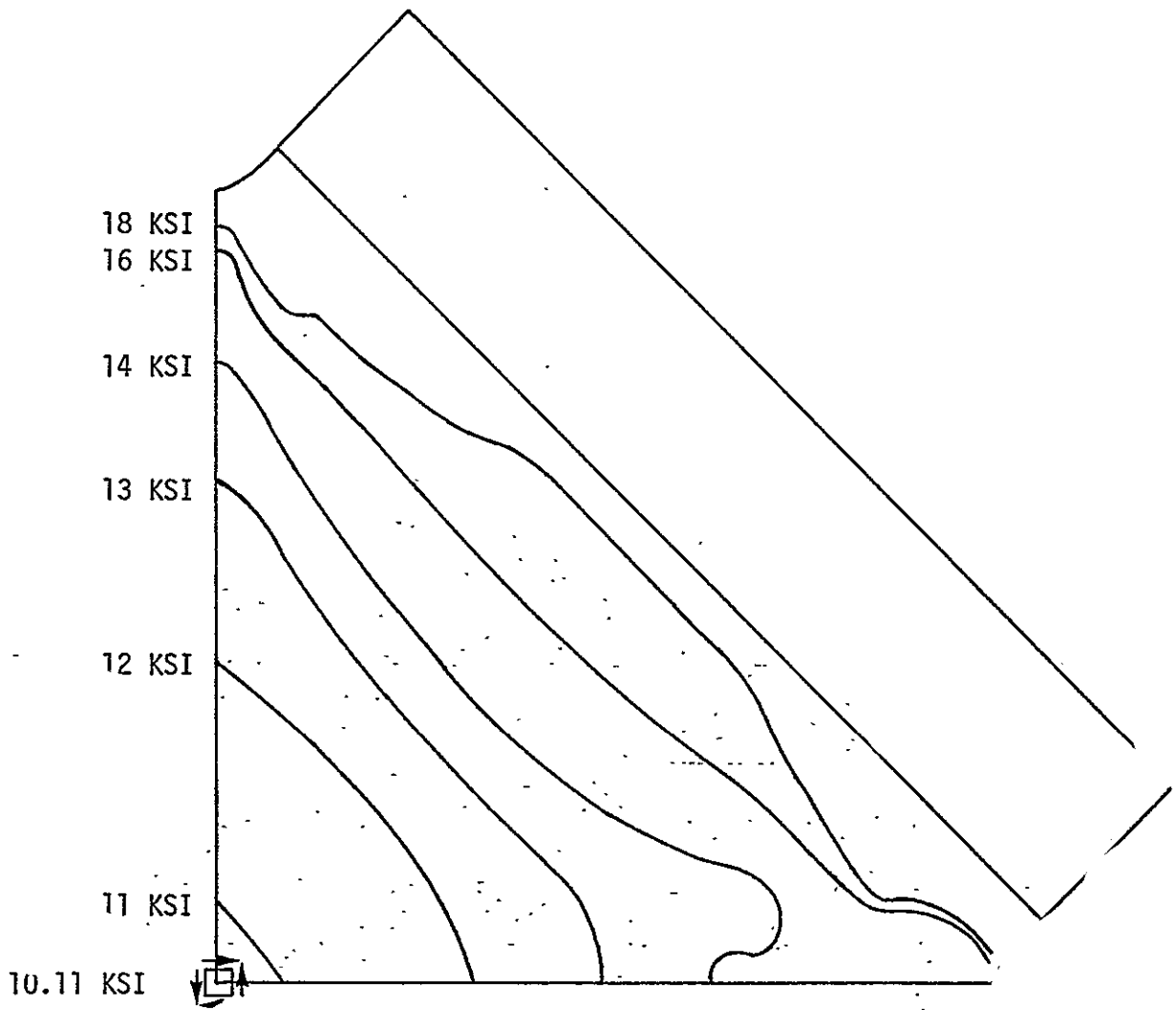


Figure 15. Contours of Constant Principal Shear Stress, τ_{max} , Predicted by SAP (Deformable Boundary with Reinforcing Titanium Doubler).

BANSAP: A BANDWIDTH REDUCTION PROGRAM FOR SAP IV

By

Donna E. Holzmacher

BANSAP: A BANDWIDTH REDUCTION PROGRAM FOR SAP IV

By

Donna E. Holzmacher

INTRODUCTION

For analysis, a structure may be broken down into parts known as finite elements. The elements of the structure may be one-dimensional such as a rod, two-dimensional such as a triangle, or quadrilateral, or three-dimensional such as a parallelepiped. The elements are positioned and described by nodes which, when connected, describe the structure. Static analysis using finite elements is accomplished by solving simultaneous equations. These equations when written in matrix form are characterized by banded coefficient matrices. Computer time and storage can be saved if the bandwidth of the matrix is a minimum. This occurs with adept numbering of the nodal points of the structure. If the nodes are numbered in an optimum way the non-zero values in the matrix will lie in a band about the diagonal. The bandwidth of a matrix is defined here as the maximum difference between any two connected nodes plus one to take into account the diagonal term.

As a particular example, consider the plane structure shown in figure 1. The displacements of this structure are determined by solving

$$(K) \{U\} = \{P\}$$

where (K) is the stiffness matrix, $\{U\}$ is the displacement vector, and $\{P\}$ is the load vector. The (K) matrix is arranged according to the connectivity of the nodes 1 through 9 of the triangular elements. The connectivity matrix for the above structure is represented in figure 2 showing that node 1 is connected to nodes 2, 8, and 9, and node 2 is connected to

nodes 1, 2, 3, 4, and 9, etc. The actual values in the stiffness matrix, corresponding to the positions of the matrix above, depend on the geometry and material of the structure.

The bandwidth of the connectivity matrix shown in figure 2 is 9. If the nodes are renumbered as in figure 3, the corresponding connectivity matrix as shown in figure 4 has a new, reduced bandwidth of 4.

In order to efficiently renumber the nodes of structures for finite element analysis a number of algorithms have been developed and incorporated into bandwidth reduction programs. Prior to 1969, authors who developed techniques to reduce the bandwidths of matrices included Always and Martin, Tewarson, Rosen, and Akyus and Utku (refs. 1 through 4). In 1969, Cuthill and McKee's (ref. 5) algorithm arranged the rows of the connectivity matrix with regard to the increasing number of non-zero off-diagonal elements. This algorithm was used in a program called BANDIT which serves as a preprocessor for NASTRAN.

H.R. Groom's algorithm for bandwidth reduction was introduced in 1972 (ref. 6). Groom systematically moved closer together rows and columns which were far apart and coupled.

In 1973, Collins (ref. 7) presented the algorithm upon which the program, BANSAP, developed in this study is based. After work on BANSAP had begun, Rodrigues (ref. 8) presented a new algorithm which, for two sample problems presented, showed a smaller bandwidth than the Cuthill and McKee, the Groom, or the Collins algorithms.

The objective of this paper is to describe a study undertaken to incorporate the Collins bandwidth algorithm in a data preprocessing computer program for the finite element program SAP IV (Structural Analysis Program - IV). First to be presented will be Collins' algorithm for bandwidth reduction which contains two subroutines, SETUP and OPTNUM. A description of the SAP IV preprocessing program BANSAP will then be given including its capabilities and limitations. Results from application of the

program to example problems will be presented and discussed. User instructions, the BANSAP program listing, and sample output are presented in appendices.

COLLINS BANDWIDTH REDUCTION ALGORITHM

Collins' algorithm for bandwidth reduction includes two subroutines, SETUP and OPTNUM. His procedure shall be illustrated using the structure in figure 1.

In the first subroutine, SETUP, a list is generated showing the connections between the different nodes shown in figure 1. The relations established by SETUP are displayed in table 1. The information is stored in arrays suitable for use in subroutine OPTNUM. The subroutine SETUP also determines the original bandwidth of the structure.

The subroutine used to renumber related nodes is OPTNUM. OPTNUM locates the origin of the different numbering schemes at each node in turn, making the number of permutations of schemes equal to the number of nodes. In other words, OPTNUM first renumbers the nodes around old node number one making old node number one the origin of the new scheme. OPTNUM then determines the bandwidth of this scheme. Next OPTNUM goes to old node number two and starts its new origin in the position of this node. It renumbers the nodes connecting node two, one at a time, and determines the maximum difference between the new connected nodes. If the maximum difference is less than the lowest maximum difference of the preceding schemes, it continues with the renumbering until the scheme is complete. If not, the current scheme is abandoned. After completion or abandonment of a scheme, OPTNUM proceeds to the next scheme starting with a new origin at the next sequential old node number. The scheme which is retained by OPTNUM is that which exhibits the lowest maximum difference between related nodes. The sequence of renumbering schemes for figure 1 is shown in table 2.

Collins' algorithm is set up to handle the renumbering of nodes for elements containing up to four nodes. Reference 5 indicates that this method has been applied to solid elements but not very successfully.

SAP IV PREPROCESSING PROGRAM

A program, BANSAP, has been written using the Collins algorithm as a preprocessing program for SAP IV. BANSAP consists of four subroutines: SAPIN, SETUP, OPTNUM, and SAPOUT as shown in figure 5.

The first subroutine, SAPIN, reads the data in the formats stipulated by SAP IV and stores element and node connections according to type. BANSAP is set up to handle two basic types of finite elements: elements connecting two nodes, and elements connecting three or four nodes. The two node elements which can be entered into subroutine SAPIN are either the truss, beam, or boundary. The actual renumbering of a two node element is the same for either element. The only difference in the handling of these elements by BANSAP is in their SAP IV formats. The three or four node elements which may be entered into subroutine SAPIN include membranes, axisymmetric two-dimensional elements and plate bending elements. Again, the only difference in the handling of these three and four node elements is in their SAP IV formats. If more than one type of element comprises the structure, the elements may be grouped according to their type. As is required for SAP IV, nodes must be sequentially numbered from one.

From subroutine SAPIN, BANSAP goes on to subroutines SETUP and OPTNUM. The new bandwidth is printed and a list of old number node numbers and new numbers is generated. As a user option the subroutine SAPOUT will punch the original elements with the new node numbers. Program BANSAP has been dimensioned in this paper to permit up to 1000 nodes and 1000 elements.

APPLICATIONS OF BANSAP

Applications of BANSAP are presented in table 3. The first two problems shown are illustrations of reduction in bandwidth which may be attained for simple problems. Problem 3 is an example taken from structural analysis of a ship radar tower. The last two entries are practical problems encountered in finite analysis of composite material structures.

The first illustration is the sample finite element scheme shown in figure 1. After renumbering by BANSAP the bandwidth was reduced from 9 to 4 and the final scheme is shown in figure 3.

The truss problem shown in figure 6a is a wagonwheel. After processing by BANSAP the bandwidth was reduced from 9 to 6. It has been found, however, that this value is not the optimum bandwidth. Collins has noted that the wagonwheel problem is a special case and the true optimum bandwidth occurs when the node number of the hub of the wheel is set equal to half the number of spokes plus one. The optimum bandwidth of the wagonwheel shown in figure 6 is actually 5.

The third structure is the ship's radar tower shown in figure 7. The original numbering scheme shown is nearly optimum with a bandwidth of 12 since the renumbering scheme only reduces the bandwidth to 9. For such structures there is no appreciable gain by using BANSAP as the structures could easily be numbered by hand to obtain a small bandwidth.

The shear panel of figure 8 is an example of a greatly enlarged bandwidth which can occur from the addition of new finite elements after the original structure has been numbered. With the addition of new elements for the shear panel a bandwidth of 406 was obtained, but after BANSAP, the bandwidth was reduced to 35.

The bolted joint specimen illustrated in figure 9 is a good example of how BANSAP can be used to obtain an optimum bandwidth when the numbering scheme is difficult to select by hand. The

nodes of the bolted joint specimen were originally numbered to permit easier data generation using a FORTRAN program. After the cards had been generated, BANSAP renumbered the nodes to reduce the bandwidth from 168 to 28.

CONCLUDING REMARKS

A FORTRAN program has been written for bandwidth reduction by nodal renumbering. The program is based upon the Collins algorithm and serves as a data preprocessor for the finite element program SAP IV. Applications of the preprocessing program to a number of simple and realistic problems have been presented.

Nodal renumbering for finite element analysis may be required for a variety of reasons. Renumbering may be needed if new elements were to be added onto a previously numbered structure or if a structure is difficult to optimally number by hand. It may also be needed if the element and nodal data were prepared by data generation programs. Such reasons clearly show a need and use for a program such as BANSAP.

BANSAP is an effective preprocessing program for SAP IV. The algorithm used greatly reduces the bandwidth for reduced computer time and storage during the finite element analysis.

APPENDIX A
USER INSTRUCTIONS

USER INSTRUCTIONS

CONTROL CARD (3I5)

Columns 1 - 5 Number of different groups of elements
6 - 10 Total number of nodes
11 - 15 The number zero for nopunched output and
any number greater than zero for punched
output

The following types of elements are permitted in the program.

Type 1 TRUSS

CONTROL CARD (3I5)

Columns 1 - 5 The number 1
6 - 10 The number of elements in group 1

Element Data Cards (3I5, 2A10)

Columns 1 - 5 Element number
6 - 10 Node number I
11 - 15 Node number J

Type 2 BEAM

CONTROL CARD (3I5)

Columns 1 - 5 The number 2
6 - 10 The number of elements in group 2

Element Data Cards (4I5, 5A10)

Columns 1 - 5 Element number
6 - 10 Node number I
11 - 15 Node number J
16 - 20 Node number K; K is any nodal point which
lies in the local 1 - 2 plane but not on
the 1 axis (see ref. 9, page iv.2.2)

Type 3 MEMBRANE

CONTROL CARD (3I5)

Columns 1 - 5 The number 3
6 - 10 The number of elements in group 3

Element Data Cards (5I5, 5A10)

Columns 1 - 5 Element number
6 - 10 Node number I
11 - 15 Node number J
16 - 20 Node number K
21 - 25 Node number L

Type 4 TWO D

CONTROL CARD (3I5)

Columns 1 - 5 The number 4
6 - 10 The number of elements in group 4

Element Data Cards (5I5, 5A10)

Columns 1 - 5 Element number
6 - 10 Node number I
11 - 15 Node number J
16 - 20 Node number K
21 - 25 Node number L

Type 6 PLATE

CONTROL CARD (3I5)

Columns 1 - 5 The number 6
6 - 10 Number of plate elements

Element Data Cards (5I5, 5A10)

Columns 1 - 5 Element number
6 - 10 Node number I
11 - 15 Node number J
16 - 20 Node number K
21 - 25 Node number L

Type 7 BOUNDARY (LINEAR SPRING)

CONTROL CARD (3I5)

Columns 1 - 5 The number 7
6 - 10 The number of elements in group 7

Element Data Cards (2I5, 6A10)

Columns 1 - 5 Node N, at which the element is placed
6 - 10 Node I

APPENDIX B
BANSAP SOURCE LISTING

```

PROGRAM BANSAP(INPUT,OUTPUT,PUNCH,
*          TAP5=INPUT,TAP6=OUTPUT,TAP7=PUNCH)
000003 DIMENSION NEWJT(1000),JOINT(1000)
000003 COMMON          LMENTS,JT(4000),MEMJT(8000),JMEM(1000),JNT(1000)
000003 COMMON/BAND/IDIFF,MINMAX
000003 COMMON/CONTR/NELG,ITYPE(5),NEL(5),NODES,IPUNCH
000003 COMMON/JUNK/A(1000,6)
000003 COMMON/UNIT/ IN,IT,IP
000003 IN = 5
000004 IT = 6
000005 IP = 7

C
C          JMEM(I) = NUMBER OF NODES TO WHICH A SINGLE NODE IS CONNECTED
C          JT(I)   = WORKING ARRAY
C          MEMJT(I) = IDENTITIES OF NODES TO WHICH A NODE IS CONNECTED
C

000006 WRITE(IT,12)
000012 12 FORMAT(1H1,9(/),
1 36X,52HB8888B      AAAAA N    N    SSSSS AAAAA P PPPP/
2 36X,53HB B A A NN N S A A P P,
3 36X,53HB B A A N N N S A A P P,
4 36X,52HB8888B AAA'AAAA N N N SSSS AAAAAA P PPPP/
5 36X,52HB B A A N N N S A A P /
6 36X,52HB B A A N NN S A A P /
7 36X,52HB8888B A A N N SSSSS A A P ,)

000012 WRITE(IT,16)
000015 16 FORMAT(1H1,5X,19H I N P U T D A T A ,//, )
000016 DO 10 I=1,1000
000020 10 JNT (I)= 0
000023 DO 20 I=1,4000
000024 20 JT(I) = 0
000027 DO 30 I=1,8000
000030 30 MEMJT(I) =0

C
C          CALL SAPIN
C
C          SUBROUTINES SETUP AND OPTNUM FROM/
C          -BANDWIDTH REDUCTION BY AUTOMATIC RENUMBERING-, R.J. COLLINS,
C          INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
C          VOLUME 6, 1973, PP 345-356.
C

000034 CALL SETJP

```

```

C
000035 WRITE(IT,32)
000041 DO 40 I= 1, NODES
000043 NO = JMEM(I)
000045 L1 = 8*(I-1) + 1
000047 L2 = L1 + NO -1
000051 WRITE(IT,34) I, NO, (MEMJT(L), L=L1, L2)
000070 40 CONTINUE
000073 MINMAX = IDIFF + 1
000075 WRITE(IT,36) MINMAX

C
000102 CALL OPTNUM

C
000103 MINMAX = MINMAX +1
000105 WRITE(IT,38) MINMAX
000112 WRITE(IT,42)

C
000116 CALL SAPJUT

C
000117 32 FORMAT(1H1,12X,4HNODE,3X,4HJMEM,16X,5HMEMJI,/)
000117 34 FORMAT(11X,2I5,10X,9I6)
000117 36 FORMAT(//,20X,20HORIGINAL BANDWIDTH =,I4 )
000117 38 FORMAT(//,20X,14HNEW BANDWIDTH=,I4)
000117 42 FORMAT(1H1,10X,33HOLD NODE NUMBER NEW NODE NUMBER,/)
000117 STOP
000121 END

```

```

000002 SUBROUTINE SAPIN
000002 COMMON LMENTS, JT(4000), MEMJT(8000), JMEM(1000), JNT(1000)
000002 COMMON/BAND/IDIFF, MINMAX
000002 COMMON/CONTR/ NELG, ITYPE(5), NEL(5), NODES, IPUNCH
000002 COMMON/JUNK/A(1000,6)
000002 COMMON/UNIT/ IN, IT, IP

```

```

C
C
C
C   NELG   = NUMBER OF DIFFERENT GROUPS OF ELEMENTS(LESS THAN 5
C   NODES  = TOTAL NUMBER OF NODES
C   IPUNCH = ZERO FOR NO PUNCHED OUTPUT, NUMBER GREATER THAN
C   ZERO FOR PUNCHED OUTPUT
C   N      = ELEMENT TYPE
C   NE     = NUMBER OF ELEMENTS OF TYPE N
C   LMENT  = TOTAL NUMBER OF ELEMENTS

```

```

C
C   ITYPE(I) = TYPE OF ELEMENT
C   NEL(I)   = NUMBER OF ELEMENTS IN A GROUP

```

ITYPE	ELEMENT	NUMBER OF NODES
1	TRUSS	2
2	BEAM	2
3	MEMBRANE	3 OR 4
4	TWO D	3 OR 4
5	BRICK	8
6	PLATE	4
7	BOUNDARY	2

```

C   READ ELEMENT CARDS AND STORE CONNECTIONS.

```

```

000002 READ(IN,12) NELG, NODES, IPUNCH
000014 WRITE(IT,14) NELG, NODES, IPUNCH
000026 DO 200 II= 1, NELG
000030 READ(IN,12) N, NE
000037 WRITE(IT,10) II, NE, N
000051 WRITE(IT,50)
000055 ITYPE(II) =N
000057 NEL(II)   =NE
000061 LMENTS =0

```

```

C
C   READ ELEMENT CONNECTIONS. FOR TRUSS, BEAM, OR BOUNDARY
C   ELEMENTS ONLY TWO CONNECTIONS I AND J ARE NEEDED. FOR

```

```

C      ALL OTHER TYPES FOUR CONNECTIONS ARE POSSIBLE- I,J,K,L.
C
C      STORE NODE CONNECTIONS ACCORDING TO TYPE.
C
000062      DO 210 JJ = 1,NE
000063      GO TO (1,2,3,3,5,3,7),N
C
000075      1 CONTINUE
000076      READ(IN,102) I,J,( A(JJ,L),L=1,2)
000115      102 FORMAT(5X,2I5,2A10)
000116      GO TO 300
C
000117      2 CONTINUE
000117      READ(IN,104) I,J,( A(JJ,L),L=1,6)
000137      104 FORMAT(5X,2I5,A5,5A10)
000137      GO TO 300
C
000140      3 CONTINUE
000140      READ(IN,106) I,J,K,L,( A(JJ,L),L=1,5)
000164      106 FORMAT(5X,4I5,5A10)
000164      GO TO 300
C
000165      5 CONTINUE
000165      WRITE(IT,108)
000171      108 FORMAT(5X,42HTHREE DIMENSIONAL ELEMENTS NOT IMPLEMENTED,/)
000171      GO TO 200
C
000172      7 CONTINUE
000172      READ(IN,110) I,J,(A(JJ,L),L=1,6)
000212      110 FORMAT(2I5,6A10)
000212      300 CONTINUE
000212      IF(N.EQ.7) GO TO 20
000214      III = JJ + ELEMENTS
000216      JJJ = III + 1000
000220      JT(III) = I
000221      JT(JJJ) = J
000223      IF( N.LE. 2) GO TO 205
000226      KKK = III + 2000
000230      LLL = III + 3000
000232      JT(KKK) = K
C
C      FOR TRIANGULAR ELEMENT SET REPEATED NODE NUMBER EQUAL TO ZERO.
C

```

```

000234      IF(K.EQ.L) L=0
000236      JT(LLL) = L
000240 205 CONTINUE
000240      WRITE(IT,30) JJ,JT(III),JT(JJJ),JT(KKK),JT(LLL)
000256 210 CONTINUE
000261      LMENTS = LMENTS + NE
000262 200 CONTINUE
000265      50 FORMAT(10X,7HELEMENT,5X,1HI,7X,1HJ,7X,1HK,7X,1HL,/)
000265      10 FORMAT(/,5X,13HELEMENT GROUP,12,4H HAS,13,17H ELEMENTS OF TYPE,
1/)
000265      12 FORMAT(3I5)
000265      14 FORMAT( ,2(/)
1 10X,25HNUMBER OF ELEMENT TYPES =,15,/
2 10X,25HNUMBER OF NODAL POINTS =,15,/
3 10X,25HPUNCHED ELEMENT CARDS =,15,/
4 10X,25H .EQ. 0 NO ,/
5 10X,25H .EQ. 1 YES ,)
000265 30 FORMAT(10X,15,4I8)
000265      RETURN
000266      END

```

```

SUBROUTINE SETUP
000002 COMMON LMENTS, JT(4000), MEMJT(8000), JMEM(1000), JNT(1000)
000002 COMMON/BAND/IDIFF, MINMAX
000002 COMMON/CONTR/NELG, ITYPE(5), NEL(5), NODES, IPUNCH
C
C     NODES     = TOTAL NUMBER OF NODES
C     JNTI      = ELEMENT NODE UNDER CONSIDERATION
C     JSUB      = LOCATION IN MEMJT(I) OF BEGINNING OF LIST OF
C     NODES RELATED TO JNTI
C     LMENT     = TOTAL NUMBER OF ELEMENTS
C     JMEM(I)   = NUMBER OF NODES TO WHICH A SINGLE NODE IS CONNECTED
C     MEMJT(I)  = IDENTITIES OF NODES TO WHICH A NODE IS CONNECTED
C     IDIFF     = BANDWIDTH = IDIFF+1 FOR ORIGINAL SCHEME
C
000002     IDIFF = 0
000003     DO 10 J= 1, NODES
000005 10 JMEM(J)= 0
000011     DO 60 J= 1, LMENTS
000012     DO 50 I= 1, 4
C
C     NEXT STATEMENT DEPENDS ON THE NUMBER OF NODES FOR WHICH THE
C     PROGRAM IS DIMENSIONED. CURRENTLY THE MAXIMUM NUMBER OF NODES
C     IS 1000.
C
000013     JNTI = JT(1000* (I-1) + J)
C
C     IF JNTI EQUALS ZERO ALL NODES OF ELEMENT J HAVE BEEN
C     CONSIDERED.
C
000017     IF(JNTI.EQ.0) GO TO 60
000021     JSUB = (JNTI - 1) * 8
000023     DO 40 II = 1, 4
000025     IF(II.EQ.1) GO TO 40
C
C     NEXT STATEMENT DEPENDS ON THE NUMBER OF NODES FOR WHICH THE
C     PROGRAM IS DIMENSIONED. CURRENTLY THE MAXIMUM NUMBER OF NODES
C     IS 1000.
C
C     RELATED NODES ARE IDENTIFIED BELOW.
C
000027     JJT = JT(1000 * (II-1) + J )
000033     IF(JJT.EQ.0) GO TO 50

```



```
C
C      DETERMINE WHETHER RELATIONSHIP BETWEEN JNTI AND JJT
C      HAS BEEN ESTABLISHED.
C
```

```
000035      MEM1 = JMEM(JNTI)
000036      IF(MEM1.EQ.0) GO TO 30
000040      DO 20 III =1, MEM1
000041      IF(MEMJT(JSUB +III).EQ.JJT) GO TO 40
```

```
C
C      FIND WIDTH OF ORIGINAL MATRIX BAND.
C
```

```
000044      20 CONTINUE
000046      30 JMEM(JNTI) =JMEM(JNTI) + 1
000050      IDUM = JSUB + JMEM(JNTI)
000052      MEMJT(IDUM) = JJT
000055      IF(IABS(JNTI-JJT).GT.IDIFF) IDIFF = IABS(JNTI -JJT)
000062      40 CONTINUE
000064      50 CONTINUE
000066      60 CONTINUE
000071      RETURN
000072      END
```

```

SUBROUTINE OPTNUM
000002 DIMENSION NEWJT(1000),JOINT(1000)
000002 COMMON      LMENTS,JT(4000),MEMJT(8000),JMEM(1000),JNT(1000)
000002 COMMON/BAND/IDIFF,MINMAX
000002 COMMON/CONTR/NELG,ITYPE(5),NEL(5),NODES,IPUNCH
000002 COMMON/UNIT/ IN,IT,IP
C
C      JOINT(I) = WORKING ARRAY
C      NEWJT(I) = WORKING ARRAY
C      JNT(I)   = NEW NUMBERING SCHEME
C      -MINMAX  = BANDWIDTH = MINMAX+1 FOR NEW SCHEME
C
C      MINMAX IS INITIALIZED.
C
000002 MINMAX = IDIFF
C
C      NEW SCHEME STARTS AT NODE OF OLD NODE NUMBER IK.
C
000004 DO 60 IK=1,NODES
000005 DO 20 J=1,NODES
C
C      JOINT(J) AND NEWJT (J) INITIALIZED TO ZERO FOR EACH NEW
C      NUMBERING SCHEME.
C
000006 JOINT(J)= 0
000007 20 NEWJT(J)= 0
C
C      INITIALIZE FOR NEW NODE NUMBER ONE.
C
000013 MAX = 0
000013 I = 1
000014 NEWJT(1) = IK
000016 JOINT(IK) = 1
000017 K = 1
000021 30 CONTINUE
000021 JDUM = NEWJT(I)
000023 K4 = JMEM(JDUM)
000025 IF(K4.EQ.0) GO TO 45
C
C      LOCATE RELATED NODES IN MEMJT(I).
C
000026 JSUB = (NEWJT(I) - 1) * 8

```

```

000031      DO 40 JJ= 1,K4
000032      K5 = MEMJT(JSUB +JJ)
000034      IF( JOINT(K5) .GT. 0 ) GO TO 40
000040      K = K+1
000041      NEWJT(K) =K5
000042      JOINT(K5)=K

C
C      CHECK DIFFERENCE BETWEEN NEW NUMBERS OF RELATED NODES.
C
000043      NDIFF = IABS(I-K)

C
C      SCHEME ABANDONED IF DIFFERENCE GREATER THAN BANDWIDTH OF
C      PRVIOUS SCHEME, NEW SCHEME STARTED.
C

000045      IF(NDIFF.GE.MINMAX) GO TO 60
000050      IF(NDIFF.GT.MAX) MAX =NDIFF
000053      40 CONTINUE
000056      IF(K.EQ.NODES)GO TO 50
000060      45 I=I+1
000062      GO TO 30
000062      50 MINMAX = MAX
000064      DO 55 J=1,NODES
000065      55 JNT(J) = JOINT(J)
000071      60 CONTINUE
000074      RETURN
000075      END

```

```

SUBROUTINE SAPOUT
000002 COMMON LMENTS, JT(4000), MEMJT(8000), JMEM(1000), JNT(1000)
000002 COMMON/CONTR/ NELG, ITYPE(5), NEL(5), NODES, IPUNCH
000002 COMMON/JUNK/A(1000,6)
000002 COMMON/UNIT/IN, IT, IP
000002 DO 10 I= 1, NODES
000004 WRITE(IT, 12) I, JNT(I)
000013 10 CONTINUE
000016 LMENTS = 0
000017 WRITE(IT, 14)
000022 DO 20 II=1, NELG
000024 N = ITYPE(II)
000026 NE= NEL(II)
000027 DO 21 JJ =1, NE
000031 I = JT( JJ+LMENTS
000034 J = JT( JJ+LMENTS +1000)
000036 NI= JNT(I)
000040 NJ= JNT(J)

C
C OUTPUT NODE CONNECTIONS ACCORDING TO TYPE.
C
000042 GO TO (1,2,3,3,5,3,7), N
C
000055 1 CONTINUE
000055 WRITE(IT, 102) JJ, NI, NJ, (A(JJ, L), L=1, 2)
000077 IF(IPUNCH.GT.0)
*WRITE(IP, 102) JJ, NI, NJ, (A(JJ, L), L=1, 2)
000122 102 FORMAT(3I5, 2A10)
000122 GO TO 21
C
000123 2 CONTINUE
000123 K = A(JJ, 1)
000126 NK= JNT(K)
000130 WRITE(IT, 104) JJ, NI, NJ, NK, (A(JJ, L), L=2, 6)
000153 IF(IPUNCH.GT.0)
*WRITE(IP, 104) JJ, NI, NJ, NK, (A(JJ, L), L=2, 6)
000200 104 FORMAT(4I5, 5A10)
000200 GO TO 21
C
000201 3 CONTINUE
000201 K = JT( JJ+LMENTS +2000)
000204 L = JT( JJ+LMENTS +3000)

```

```

000206         IF(L.EQ.0) L=K
000210         NK= JNT(K)
000212         NL= JNT(L)
000214         WRITE(IT,106)JJ,NI,NJ,NK,NL,(A{JJ,L},L=1,5)
000242   106  FORMAT(5I5,5A10)
000242         IF(IPUNCH.GT.0)
000271         *WRITE(IP,106)JJ,NI,NJ,NK,NL,(A{JJ,L},L=1,5)
000271         GO TO 21
C
000272         5  CONTINUE
000272         GO TO 21
C
000273         7  CONTINUE
000273         WRITE(IT,108)NI,NJ,(A{JJ,L},L=1,6)
000313         IF(IPUNCH.GT.0)
000334         *WRITE(IP,108)NI,NJ,(A{JJ,L},L=1,6)
000334   108  FORMAT(2I5,6A10)
000334         21  CONTINUE
000337         LMENTS = LMENTS +NE
000340         20  CONTINUE
000342         12  FORMAT(16X,I5,13X,I5)
000342         14  FORMAT(1H1,/,/,4X,17HNEW ELEMENT CARDS,/)
000342         .30  FORMAT( 5I5)
000342         RETURN
000343         END

```

APPENDIX C
SAMPLE BANSAP OUTPUT

BBBBB	AAAAA	N	N	SSSSS	AAAAA	PPPPP
B B	A A	NN	N	S	A A	P P
B B	A A	N N	N	S	A A	P P
BBBBB	AAAAAAA	N N	N	SSSS	AAAAAAA	PPPPP
B B	A A	N N	N	S	A A	P
B B	A A	N	NN	S	A A	P
BBBBB	A A	N	N	SSSSS	A A	P

I N P U T D A T A

NUMBER OF ELEMENT TYPES = 1
NUMBER OF NODAL POINTS = 9
PUNCHED ELEMENT CARDS = 0
.EQ. 0 NO
.EQ. 1 YES

ELEMENT GROUP 1 HAS 8 ELEMENTS OF TYPE 3

ELEMENT	I	J	K	L
1	1	9	8	-0
2	1	2	9	-0
3	2	4	9	-0
4	2	3	4	-0
5	8	6	7	-0
6	8	9	6	-0
7	9	5	6	-0
8	9	4	5	-0

NODE	JMEM	MEMJT					
1	3	9	8	2			
2	4	1	9	4	3		
3	2	2	4				
4	4	2	9	3	5		
5	3	9	6	4			
6	4	8	7	9	5		
7	2	8	6				
8	4	1	9	6	7		
9	6	1	8	2	4	6	5

ORIGINAL BANDWIDTH = 9

NEW BANDWIDTH= 4

OLD NODE NUMBER	NEW NODE NUMBER
1	4
2	2
3	1
4	3
5	6
6	8
7	9
8	7
9	5

NEW ELEMENT CARDS

1	4	5	7	7
2	4	2	5	5
3	2	3	5	5
4	2	1	3	3
5	7	8	9	9
6	7	5	8	8
7	5	6	8	8
8	5	3	6	6

REFERENCES

1. "An Algorithm for Reducing the Bandwidth of a Matrix of Symmetrical Configuration", by Always, G.G. and Martin, D.W., Computer Journal, Volume 8, 1965, pp. 264-272.
2. "Row Column Permutation of Sparse Matrices", by Tewarson, R.P., Computer Journal, Volume 10, 1967, pp. 300-305.
3. "Matrix Bandwidth Minimization", by Rosen, R., Proceedings of 23rd National Conference, Association for Computing Machinery, Brandon Systems Press, Princeton, NJ, 1968, pp. 585-595.
4. "An Automatic Relabelling Scheme for Bandwidth Minimization of Stiffness Matrices", by Akyuz, F.A. and Utku, S., Journal of the American Institute of Aeronautics and Astronautics, Volume 6, 1968, pp. 728-730.
5. "Reducing the Bandwidth of Sparse Symmetric Matrices", by Cuthill, E., and McKee, J., presented at National Conference of Association for Computing Machinery, San Francisco, CA, 1969, pp. 157-172.
6. "Algorithm for Matrix Bandwidth Reduction," by Grooms, H.R. Journal of the Structural Division, ASCE, Volume 98, 1972, pp. 203-214.
7. "Bandwidth Reduction by Automatic Renumbering", by Collins, R.J., International Journal for Numerical Methods in Engineering, Volume 6, 1973, pp. 345-356.
8. "Node Numbering Optimization in Structural Analysis", by Rodrigues, J.S., Journal of the Structural Division, ASCE, Volume 101, 1975, pp. 361-375.
9. "SAP IV, A Structural Analysis Program for Static and Dynamic Response of Linear Systems", by Bathe, K., Wilson, E.L., and Peterson, F.E., Report No. EERC 73-11, June 1973.

Table 1. Example node connections determined in subroutine SETUP.

<u>Node</u>	<u>Number of Connected Nodes</u>	<u>Connected Nodes</u>
1	3	9, 8, 2
2	4	1, 9, 4, 3
3	2	2, 4
4	4	2, 9, 3, 5
5	3	9, 6, 4
6	4	8, 7, 9, 5
7	2	8, 6
8	4	1, 9, 6, 7
9	6	1, 8, 2, 4, 6, 5

Table 2. Trail Numbering Schemes Used in OPTNUM.

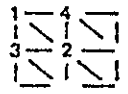
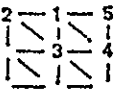
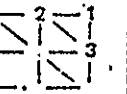
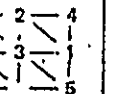
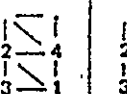
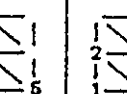
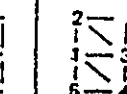
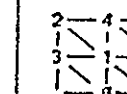

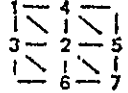
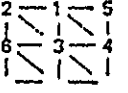
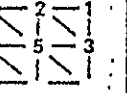
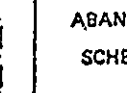
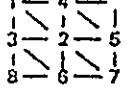
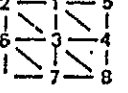
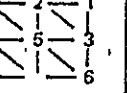
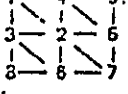
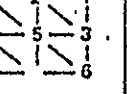
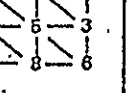
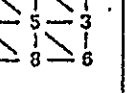
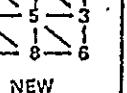
		OLD NODE NUMBER AT WHICH ORIGIN OF NEW NUMBERING SCHEME IS SET.								
		1	2	3	4	5	6	7	8	9
NEW NODE NUMBERS WHOSE RELATED NODES ARE ASSIGNED NEW NUMBERS.	1									
		DIFF = 3	DIFF = 4	DIFF = 2	DIFF = 4	DIFF = 3	DIFF = 4	DIFF = 2	DIFF = 4	DIFF = 6
	2				ABANDON SCHEME	ABANDON SCHEME	ABANDON SCHEME		ABANDON SCHEME	ABANDON SCHEME
		DIFF = 5	DIFF = 5	DIFF = 3				DIFF = 3		
	3							ABANDON SCHEME		
			DIFF = 5	DIFF = 5				ABANDON SCHEME		
	4		ABANDON SCHEME							
	NEW BANDWIDTH = DIFF + 1 = 6	ABANDON SCHEME	DIFF = 5							
5										
			DIFF = 5							
6										
			DIFF = 5							
7										
			DIFF = 5							
			NEW BANDWIDTH = DIFF + 1 = 4							

Table 3. Summary of applications of BANSAP.

<u>Structure</u>	<u>Element Type</u>	<u>Number of Nodes</u>	<u>Number of Elements</u>	<u>Old Bandwidth</u>	<u>New Bandwidth</u>
Sample problem Figure 1	Membrane	9	8	9	4
Wagonwheel Figure 6	Truss	9	16	9	6
Ship tower Figure 7	Beam	25	65	12	9
Shear panel specimen Figure 8	Membrane	595	554	406	35
Bolted joint specimen Figure 9	Membrane	398	349	168	28

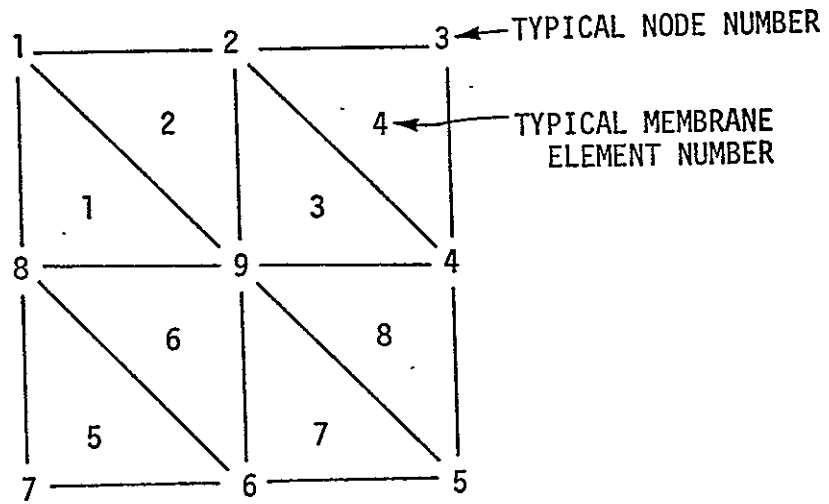


Figure 1. Sample Finite Element Scheme

		NODE								
		1	2	3	4	5	6	7	8	9
NODE	1	X	X						X	X
	2	X	X	X	X					X
	3		X	X	X					
	4		X	X	X	X				X
	5				X	X	X			X
	6					X	X	X	X	X
	7						X	X	X	
	8	X					X	X	X	X
	9	X	X		X	X	X		X	X

Figure 2. Connectivity Matrix of Sample Scheme.

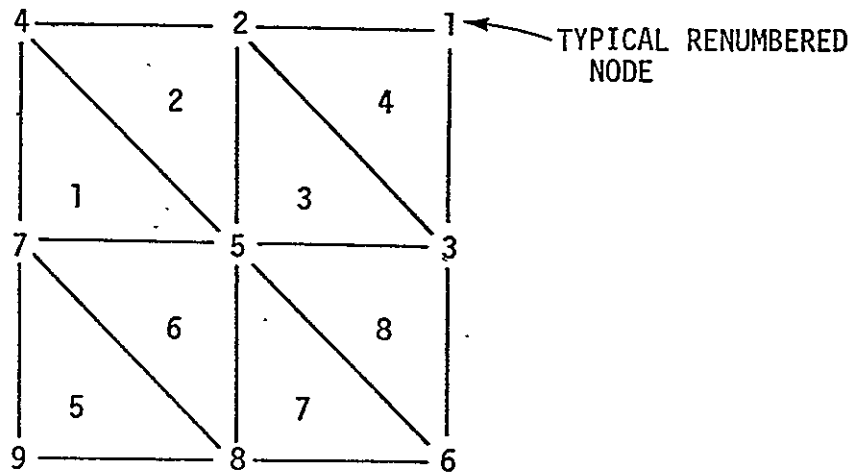


Figure 3. Renumbered Finite Element Scheme.

		NODE								
		1	2	3	4	5	6	7	8	9
NODE	1	X	X	X						
	2	X	X	X	X	X				
	3	X	X	X		X	X			
	4		X		X	X		X		
	5		X	X	X	X	X	X	X	
	6			X		X	X		X	
	7				X	X		X	X	X
	8					X	X	X	X	X
	9							X	X	X

Figure 4. Connectivity Matrix of Renumbered Scheme.

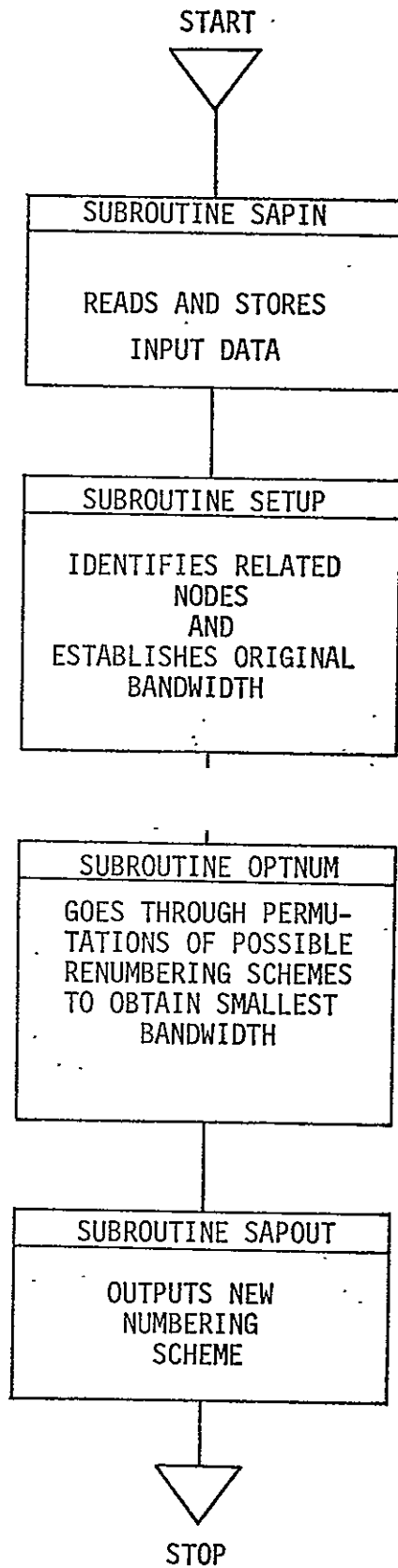
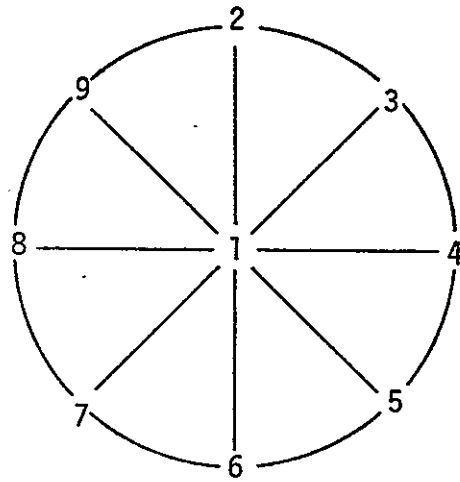
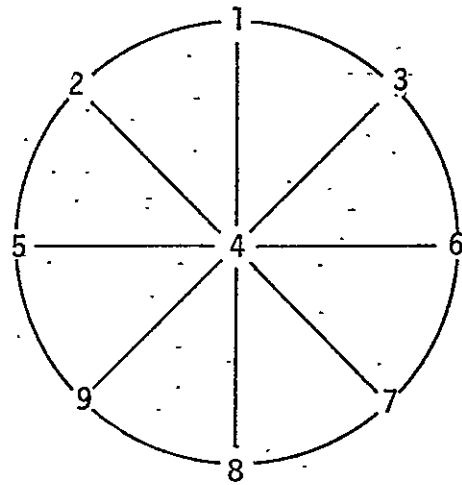


Figure 5. Flowchart of SAP IV BANSAP Preprocessing Program.



(a) Original Scheme



(b) Renumbered Scheme

Figure 6. Wagonwheel Truss.

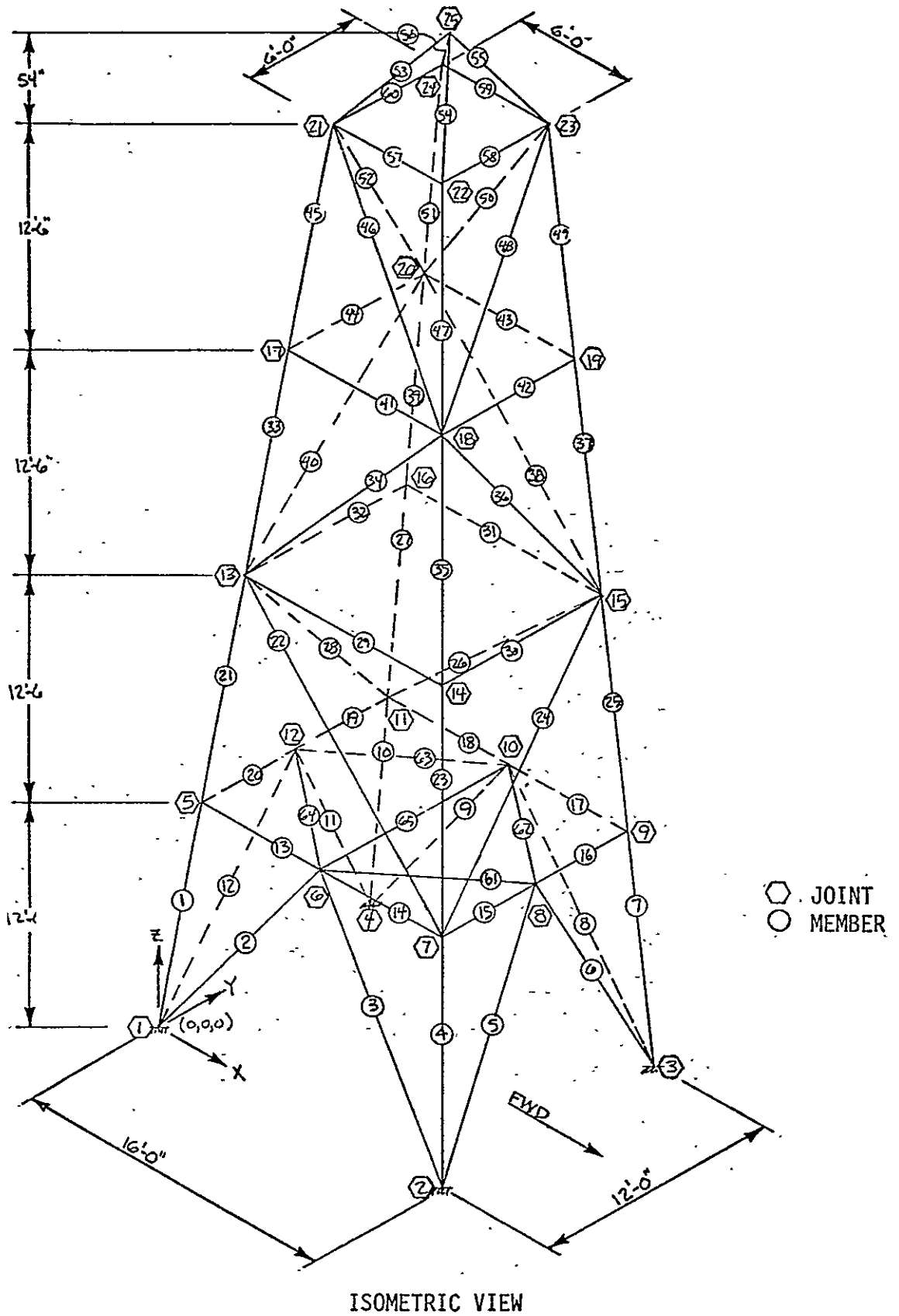


Figure 7. Ship Radar Tower.

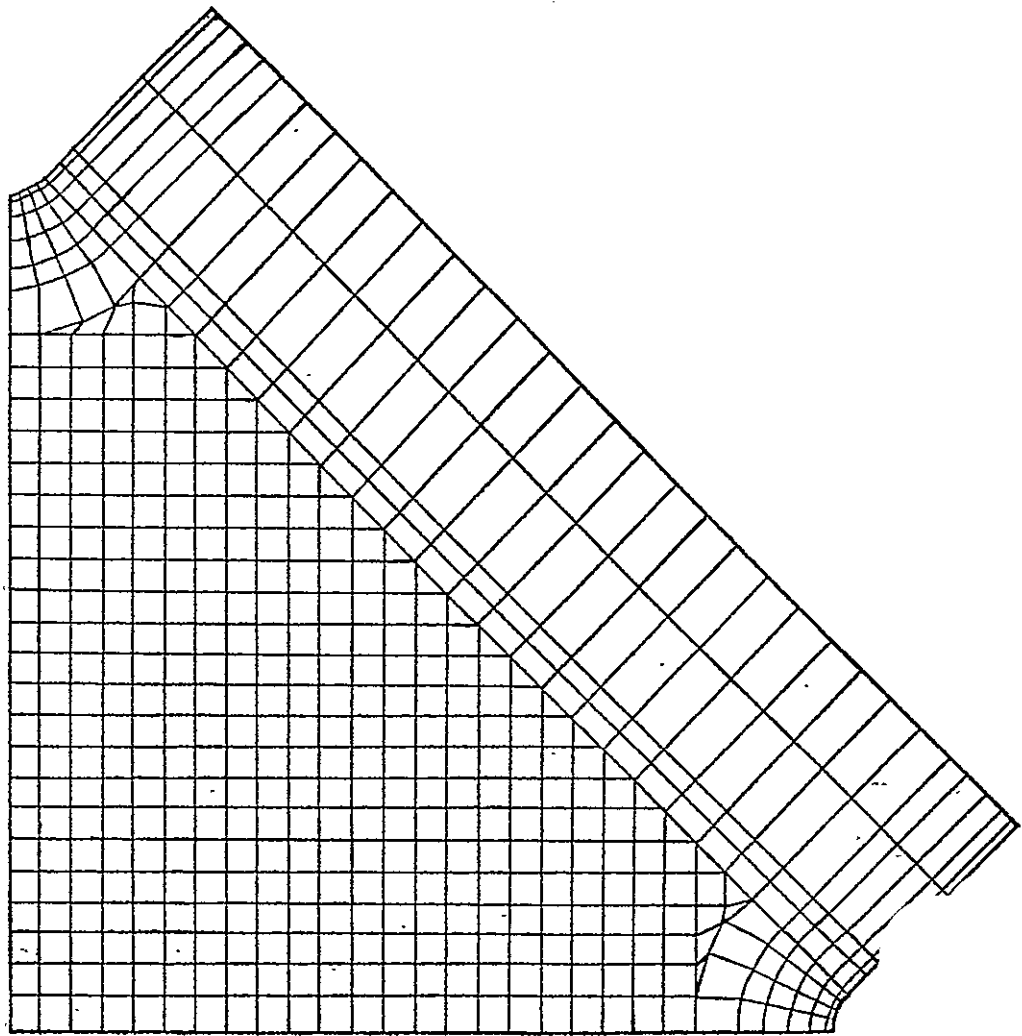


Figure 8. Shear Panel Specimen.

TRIANGULAR AND QUADRILATERAL MEMBRANE ELEMENTS

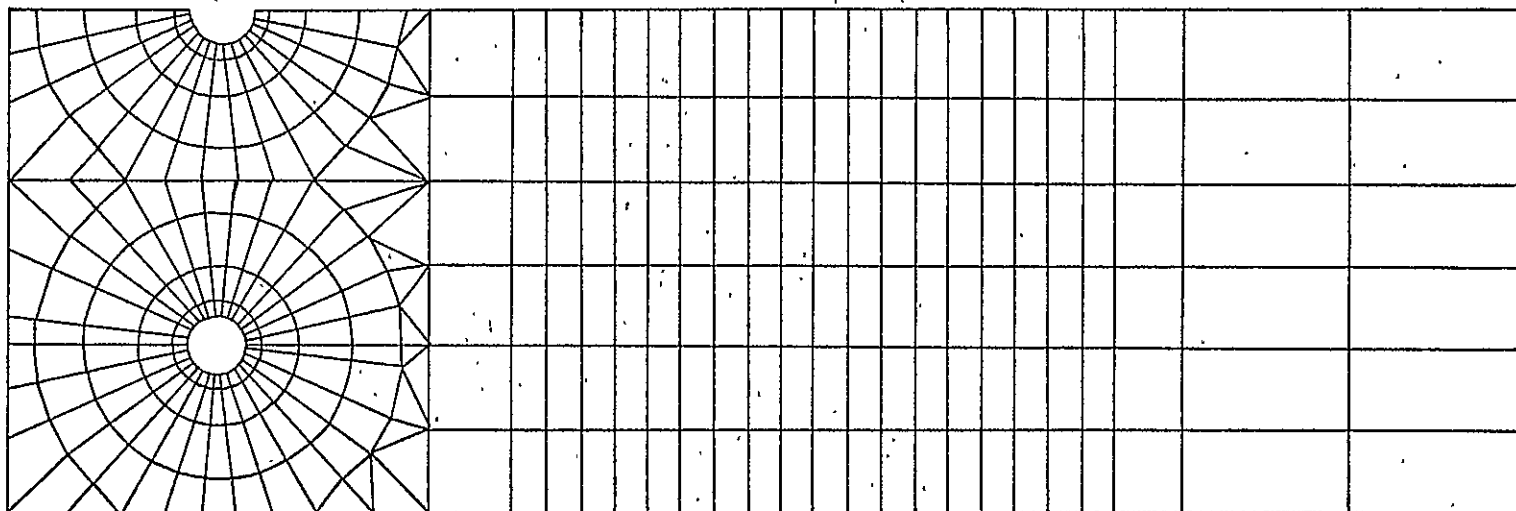


Figure 9: Finite Element Mesh for Composite Bolted Joint Specimen.

FEMESH: A FINITE ELEMENT MESH GENERATION PROGRAM
BASED ON ISOPARAMETRIC ZONES

By

Zoa C. Lane

FEMESH: A FINITE ELEMENT MESH GENERATION PROGRAM
BASED ON ISOPARAMETRIC ZONES

By

Zoa C. Lane

INTRODUCTION

Finite element analysis programs greatly facilitate the determination of deformations and stresses in structures. A major inconvenience in utilizing this analysis technique is the large amount of input data required by the computer programs. This data includes, in addition to material characteristics, the node numbers defining the elements and the spatial coordinates for each node.

Current mesh generation methods include for simple problems data preparation by hand, and for more complex problems, the coding and executing of FORTRAN mesh generation programs which generate data for a general structure.

W.R. Buell and W.A. Bush surveyed some techniques used in current mesh generation schemes (ref. 1). The techniques presented by Buell and Bush are: a straight line interpolation technique, a sides and parts technique for axisymmetric structures, electro-mechanical techniques for two- and three-dimensional structures, and a simplified finite difference technique and equipotential technique for general structure shapes.

The advantages of general structure mesh generation programs (ref. 1) are: (1) reduced cost due to reduction of man hours and computer time needed to generate and check data; (2) reduced number of errors; (3) insured regularity of finite elements; and (4) application to a variety of structural shapes.

O.C. Zienkiewicz (ref. 2) utilizes a technique involving the mapping of isoparametric quadrilaterals from a natural to a cartesian coordinate system in an automatic mesh generation

scheme for plane and curved surfaces. This scheme is applicable to non-quadrilateral structures if the structure is divided into quadrilateral regions. Zienkiewicz's technique for mesh generation was used by S.J. Womack (ref. 3) as a preprocessor for TEXGAP, a finite element program for the analysis of two-dimensional linearly elastic plane or axisymmetric bodies (ref. 4).

The objective of this study is to utilize the technique developed by Zienkiewicz in a mesh generation scheme for two-dimensional planar surfaces. Presented in this paper are a description of the mapping technique, a description of the computer program, and three examples of meshes generated by the program. A set of user instructions and a listing of the program are included in the appendices.

INTERPOLATION FUNCTION TECHNIQUE FOR FINITE ELEMENT GENERATION

The algorithm used by Zienkiewicz to map an isoparametric quadrilateral is the displacement interpolation equations used in isoparametric finite elements (ref. 5). The interpolation equations for quadratic bounded surfaces (which are listed in table 1), are a function of a set of dimensionless coordinates, ξ and η , which define a natural coordinate system.

In the natural coordinate system (fig. 1), a planar surface is represented as a square whose dimensions are 2 x 2 units and whose center is at the origin. To map a surface into the cartesian coordinate system, eight boundary points (x_i and y_i) and the ξ and η values of each grid point on the surface to be mapped are substituted in the displacement interpolation functions; the resulting values are the cartesian coordinates of the grid points.

A mesh is generated by dividing the square into the desired number of subdivisions, calculating the ξ and η coordinates for each grid point, and mapping each point to the cartesian coordinate system. A graduation of a generated mesh is obtained

by offsetting the midside node from the midpoint of a side of the quadrilateral (fig. 2). The generated elements will vary in size along that side; smaller elements will be in the direction of the offset.

Meshes for complex structures are generated by dividing the structure into quadrilateral zones. The mesh for each zone is generated independent of other zones. Connection of zones is accomplished by eliminating node numbers and coordinates which were duplicated on zone boundaries.

PROGRAM FEMESH

Program FEMESH is a FORTRAN IV code for generating finite element data for two-dimensional planar surfaces. The algorithm used to generate the node coordinates is based on the displacement interpolation functions (table 1) described in the preceding paragraph.

Input data for FEMESH includes a title, the number of zones, the total number of zone nodes, the number of zone node coordinates to be read from cards, the first node and element numbers, a list of the eight nodes which define a zone, the dimensions of the desired mesh of each zone, and the zone node coordinates.

A zone is a quadrilateral region whose geometry is defined by eight zone nodes. (Zone nodes are used only in the input definition of the geometry; they are not included in the generated mesh.) The zone nodes are listed in counter-clockwise order. As indicated in figure 3, the first node identifies a corner of the quadrilateral. The second, fourth, sixth, and eighth nodes are referred to as midside nodes. If a midside node does not lie on the midpoint of a side, a graduation of the mesh results.

The general flow for the mesh generation program, FEMESH, is shown in figure 4. As indicated, the mesh for each zone is generated separately. The first step in the mesh generation scheme is to determine if the coordinates of the midside

nodes are defined (i.e., if their coordinates were supplied by the user). If the coordinates are not defined, the midside node is assumed to lie at the midpoint of a linear line segment. The second step is to determine if either of the four sides of the zone is connected to a zone for which a mesh was previously generated. If a side is connected to such a zone the node numbers and the x and y coordinates which have already been generated are used. The remainder of the mesh is then generated. This process is repeated until the meshes for all zones are generated.

The output of program FEMESH includes a listing of the elements, their four node numbers and the node coordinates. A plot of the mesh is also generated.

APPLICATIONS

Three finite element mesh generated by FEMESH are presented in this section. The first example is a sample problem illustrating the input and output of program FEMESH. The second is a quarter section of a shear panel. The third is a half section of a bolted joint specimen.

The first example is a simple structure originally used to validate the ability of FEMESH to properly connect zones. The structure (illustrated in fig. 5a) is divided into three zones. The eighteen zone nodes are labeled arbitrarily and illustrated in figure 5b. Figures 5a and 5b represent the input required by program FEMESH to generate the mesh illustrated in figures 5c and 5d. Figure 5c illustrates the node numbers, and 5d illustrates the element numbers.

The input data for this problem is tabulated in table 2 (see Appendix A for user instructions). The data includes a title card, a control card, three zone description cards, and eight node coordinate cards. The control card specifies the number zones (3), the number of zone nodes (18), the number of zone node coordinate cards to be read (8), the first node

number (100), and the first element number (1000). A typical zone description card lists the eight zone nodes defining each zone and the size of the finite element mesh to be generated.

The tabulated output for this problem appears in table 3. The output includes the input data, the element number, the four node numbers which define each element, and the cartesian coordinates of each node.

The shear panel illustrated in figure 6 is divided into four zones. The zones were established in such a way that the straight and curved segments of the corner fillets are assigned to different zones in order to obtain a closer approximation of the true boundary shape.

The mesh dimension for zone I is 20 x 20, for zone II is 20 x 3, for zone III is 20 x 20, and zone IV is 3 x 30. To avoid the generation of long, narrow rectangular elements, the midside nodes 2, 8, 9, and 14, 15, 16 are moved away from the midpoint of the line segment toward the fillets. The input data is summarized in table 4. The output is illustrated in figure 7. Because of the large number of generated elements, the output is not listed in tabular form; it is represented graphically by a computer plot of the generated mesh. The generated mesh is composed of 574 nodes and 520 elements.

The mesh for one-half of a bolted joint specimen was generated by dividing the specimen into 15 zones as illustrated in figure 8. The input data for this problem (table 5) consisted of 58 data cards, including 15 zone description cards, and 41 node coordinate cards. A graduation of the mesh of zones II, III, IV, V, VI, VIII, and IX was used to obtain a uniformity in the shape of the generated elements. The generated mesh, which is illustrated in figure 9, consists of 378 elements and 435 nodes.

CONCLUDING REMARKS

Program FEMESH, a FORTRAN IV code, has been developed to generate a finite element mesh for two-dimensional, planar

surfaces. The algorithm used is the displacement interpolation functions which were developed for mesh generation by Zienkiewicz.

A structure may be subdivided into a maximum of 15 zones. The maximum mesh for each zone is 24 x 24 elements (or 25 x 25 node points). FEMESH will compute a maximum of 4000 node points, and output the node numbers and their coordinates and the element numbers and their four identifying node numbers. A simple plot of the finite element mesh is also generated.

Presented in this paper is a description of the technique used in the mesh generation scheme, a description of program FEMESH and examples of the mesh generated for three problems. User instructions and a listing of the program are included in the appendices.

APPENDIX A
USER INSTRUCTIONS FOR FEMESH

USER INSTRUCTIONS FOR FEMESH

Program FEMESH generates isoparametric finite element meshes for two-dimensional planar surfaces. The input required by the program consists of four types of data cards: a title card, a control card, zone description cards, and node coordinate cards (fig. A1).

TITLE CARD (Format 10A4).

<u>Column</u>	<u>Variable</u>	<u>Description</u>
1-40	TITLE	Heading for output

CONTROL CARD (Format 6I5):

<u>Column</u>	<u>Variable</u>	<u>Description</u>
1-5	IZ	Number of zones ($IZ \leq 15$)
5-10	NT	Total number of zone nodes
11-15	NI	Number of zone node coordinates to be read as input on cards
16-20	INODE	First node number to be assigned to generated mesh
21-25	IELM	First element number to be assigned to generated mesh
26-30	IP	Punch indicator: 0 will not punch 1 punch

A zone is a quadrilateral with either linear or curved line segments. The geometry of the zone is defined by 8 zone nodes whose coordinates are supplied by the user (see node coordinate card).

The values of NI and NT may differ due to the ability of the program to linearly interpolate to define the coordinates of the midside node if those coordinates are not supplied by the user. Midside nodes are those zone nodes which lie between two corner nodes. It is not necessary that a midside node lie at the midpoint of a line segment.

ZONE DESCRIPTION CARD (Format 10I5):

<u>Column</u>	<u>Variable</u>	<u>Description</u>
1-5	NODE (I,1)	Zone nodes defining zone
6-10	NODE (I,2)	geometry
11-15	NODE (I,3)	I is the zone number
16-20	NODE (I,4)	
21-25	NODE (I,5)	
26-30	NODE (I,6)	
31-35	NODE (I,7)	
36-40	NODE (I,8)	
41-45	M	Number of subdivisions along the side defined by 1st, 2nd, and 3rd zone nodes
46-50	N	Number of subdivisions along the side defined by 3rd, 4th, and 5th zone nodes.

Zone numbers are determined by the order of the zone description cards. The first zone description card is assigned the number one, the second is assigned the number two, etc.

The interconnectivity of zones is indicated by assigning a negative magnitude to zone nodes which lie on a side connected to a zone with a smaller zone number. For example, if 4 zones are connected as shown in figure A2, then the first eight values of the zone description cards should be:

Card 1:	1	2	3	7	11	10	9	6
Card 2:	-3	4	5	8	13	12	-11	-7
Card 3:	-11	-12	-13	16	21	20	19	15
Card 4:	-9	-10	-11	-15	-19	18	17	14

A side which is divided into M subdivisions must not be connected to a side divided into N subdivisions unless the values M and N are equal.

NODE COORDINATE CARD (Format I5, 2F10.5):

<u>Column</u>	<u>Variable</u>
1-5	Node number
5-15	x coordinate
16-25	y coordinate

This card may be omitted for any midside node which lies on a straight line if a graduation of the mesh is not desired.

A graduation in the mesh occurs when the midside node is offset from the midpoint of the line segment. The smaller elements will be in the same direction as the offset.

Due to a restriction in the FORTRAN coding, a midside node should not be assigned the coordinates (0,0) if the line segment is not a straight line.

STATEMENT NUMBER						CONTINUATION						FORTRAN STATEMENT																																																
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61
												TITLE CARD																																																
						TITLE						FORMAT(10,A4)																																																
												CONTROL CARD																																																
IIZ		NT		NI		INODE		IELM		IP		FORMAT(6I5)																																																
												ZONE DESCRIPTION CARD																																																
NODE 1		NODE 2		NODE 3		NODE 4		NODE 5		NODE 6		NODE 7		NODE 8		M		N		FORMAT(10I5)																																								
												NODE COORDINATE CARD																																																
NODE NO.		X COORDINATE		Y COORDINATE		FORMAT(I5,2F10.5)																																																						

Figure A1. Input Data Formats for FEMESH.

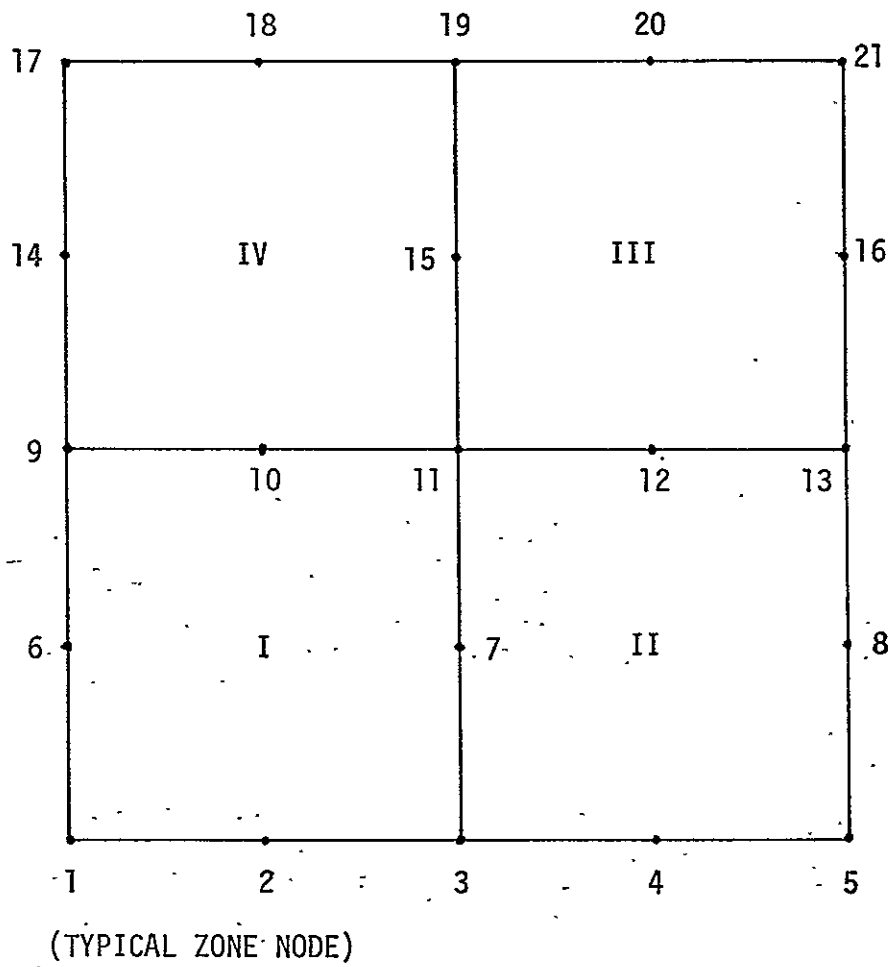


Figure A2. Simple Structure to Illustrate Zone Node Input Data.

APPENDIX B
FORTRAN LISTING OF MESH GENERATION PROGRAM, FEMESH
(LRC, CDC-6600 COMPUTER VERSION)

```

PROGRAM FEMESH(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,PUNCH)

PROGRAM FEMESH          CODED BY Z. C. LANE          MAY 31, 1975

PROGRAM FEMESH GENERATES FINITE ELEMENT DATA FOR TWO DIMENSIONAL
PLANAR SURFACES.  STRUCTURES MAY BE SUBDIVIDED INTO AS MANY AS 15
QUADRILATERAL ZONES.  THE MAXIMUM MESH DIMENSION IS 24 X 24 SUBDIVISIONS.

* NEGATIVE ZONE NODE NUMBERS FOR A ZONE IDENTIFIED BY A NUMBER **N**
  INDICATES THAT THE NEGATIVE NODE IS CONNECTED TO A ZONE WHICH IS
  IDENTIFIED BY A NUMBER LESS THAN **N**.

**  INODE FIRST NODE NUMBER          ** M,N NUMBER OF SUBDIVISIONS
**  ICTZ CURRENT ZONE NUMBER         ** NT TOTAL NUMBER OF INPUT NODES
**  IZ  TOTAL NUMBER OF ZONES        ** NI NUMBER OF NODES TO BE READ
**  IP  PUNCH WHEN IP=1              **

000003 DIMENSION A(8),TITLE(10),XNCDE(8),YNODE(8)
J00003 DIMENSION X1(78),Y1(78)
000003 COMMON ICTZ,IZONE(15,10),NODE(15,25,25),TEMP(25)
000003 COMMON X2(4002),Y2(4002)

000003 CALL PSEUDO
000004 CALL LEROY

* DETERMINE INPUT - OUTPUT DEVICES
J00005 IN = 5
000006 IOUT = 6

000007 1 FORMAT(6I5)
000007 2 FJRMAT(10I5)
000007 3 FJRMAT(I5,2F10.5)

```

```

000006      3 FORMAT(15,2F10.5)
000006      4 FORMAT(1H ,12HNO. OF ZONES,13,/)
000006      5 FORMAT(1H ,13,1X,8I5,1X,2I4)
000006      6 FORMAT(1H ,13,2X,F7.3,2X,F7.3)
000006      7 FORMAT(10A4)
000006      8 FORMAT(1H1,10A4//)
000006      9 FORMAT(1H ,4HZONE,15X,10HZONE NODES,18X,1HM,4X,1HN,/2X,3HNO.,4X,1H
000006      11,4X,1H2,4X,1H3,4X,1H4,4X,1H5,4X,1H6,4X,1H7,4X,1H8/)
000006     12 FORMAT(//,1X,4HNODE,4X,1HX,8X,1HY)

```

C
C
C
C

```

000006     INITIALIZE X1 AND Y1 TO BE FILLED FROM DATA READ OFF CARDS
000010     DO 34 I=1,78
000011     X1(I) = 0.
000011     Y1(I) = 0.
000012     34 CONTINUE

```

C
C
C
C

```

000014     * READ INPUT
000021     READ(IN,7)TITLE
000041     READ(IN,1)IZ,NT,NI,INODE,IELM,IP
000041     READ(IN,2)((IZONE(I,J),J=1,10),I=1,IZ)

```

C
C

```

000060     * WRITE INPUT
000066     WRITE(IOUT,8)TITLE
000074     WRITE(IOUT,4)IZ
000100     WRITE(IOUT,9)
000121     WRITE(IOUT,5)(I,(IZONE(I,J),J=1,10),I=1,IZ)
000121     WRITE(IOUT,12)

```

C

```

000125     DO 10 J=1,NI
000127     READ(IN,3) I,X1(I),Y1(I)
000140     WRITE(IOUT,6) I,X1(I),Y1(I)
000152     10 CONTINUE

```

C
C

```

000155     * SET COUNTER OF NODE NUMBERS, ICTN AND ZONE,ICTZ
000156     ICTZ = 0
000156     ICTN = INODE
000157     NCOR = 0

```

C

```

000160      DO 35 I=1,1661
000161      X2(I) = 0.
000162      Y2(I) = 0.
000163      35 CONTINUE
000165      DO 1010 I=1,10
000166      DO 1009 J=1,25
000167      DO 1008 K=1,25
000170      NODE(I,J,K) = 0
000175      1008 CONTINUE
000177      1009 CONTINUE
00020      1010 CONTINUE
      C
      C
000203      1000 CONTINUE
000203      ICTZ = ICTZ + 1
      C
      C
      C      SET INDICATORS TO ZERO
      C
000205      IS1 = 0
000206      IS2 = 0
000207      IS3 = 0
000210      IS4 = 0
      C
      C      * PULL FROM THE IZONE ARRAY THE ZONE NUMBERS AND THE ZONE COORDINATES
      C
000211      DO 20 I=1,8
000212      IC = IABS(IZONE(ICTZ,I))
000216      XNODE(I) = X1(IC)
000220      YNODE(I) = Y1(IC)
000221      20 CONTINUE
000223      M = IZONE(ICTZ,9)
000225      N = IZONE(ICTZ,10)
000226      NN = N + 1
000230      MM = M + 1
      C
      C
      C
      C      * TITLE THE OUTPUT FOR THIS ZONE
      C
000231      WRITE(IOUT,11)ICTZ
000237      11 FORMAT(1H1,28HCALCULATIONS FOR ZONE NUMBER,I4//)

```

```

C
C   IF NO VALUE IS GIVEN FOR MIDPOINTS, ASSUME A STRAIGHT LINE AND CALCULATE
C   THE MIDPOINT.
C
000237   GO 30 I=2,8,2
000241   IF(XNODE(I))30,25,30
000242   25 IF(YNODE(I))30,26,30
000244   26 K = 8-I
000246   IF(K)30,27,28
000247   28 XNODE(I) = (XNODE(I+1)+XNODE(I-1))/2.
000253   YNODE(I) = (YNODE(I+1)+YNODE(I-1))/2.
000256   GO TO 30
000256   27 XNODE(I) = (XNODE(1)+XNODE(7))/2.
000262   YNODE(I) = (YNODE(1)+YNODE(7))/2.
000264   30 CONTINUE
000266   WRITE(IOUT,31)(I,XNODE(I),YNODE(I),I=1,8)
000303   31 FORMAT(1H,10HZONE NODE ,I4,F15.2,F25.2)
000303   WRITE(IOUT,32)
000307   32 FORMAT(/,1X,8HNODE NO.,5X,1HX,8X,1HY/)
C
C
C
C
C   * IF ZONE NUMBER IS ONE, FILL NODE ARRAY AND SKIP TO X, Y COORDINATE
C   CALCULATIONS.
C
000307   IF(ICTZ-1)199,190,201
000312   190 GO 192 J=1,NN
000314   DO 191 I=1,MM
000315   NODE(I,I,J) = ICTN
000322   ICTN = ICTN + 1
000323   191 CONTINUE
000325   192 CONTINUE
000330   GO TO 3000
000330   199 WRITE(IOUT,200) ICTZ
000336   200 FORMAT(1H1,28HERROR ... ZONE NUMBER ICTZ =.I4)
000336   201 CONTINUE

```

DETERMINE WHICH SIDES ARE CONNECTED
FILL THE NODE ARRAY


```

C
C   SIDE ONE
C
000336   IF(IZONE(ICTZ,2))212,220,220
000340   212 CALL FIND(1,3,1)
000343       DO 213 I=1,MM
000345       NODE(ICTZ,I,1) = TEMP(I)
000353   213 CONTINUE
000355       IS1 = 1

C
C   SIDE 2
C
000356   220 CONTINUE
000356       IF(IZONE(ICTZ,4))222,230,230
000360   222 CALL FIND(3,5,2)
000363       DO 223 I=1,NN
000365       NODE(ICTZ,MM,I) = TEMP(I)
000375   223 CONTINUE
000377       IS2 = 1

C
C   SIDE 3
C
000400   230 CONTINUE
000400       IF(IZONE(ICTZ,6))232,240,240
000402   232 CALL FIND(5,7,3)
000405       DO 233 I=1,MM
000407       NODE(ICTZ,I,NN) = TEMP(I)
000417   233 CONTINUE
000421       IS3 = 1

C
C   SIDE 4
C
000422   240 CONTINUE
000422       IF(IZONE(ICTZ,8))242,250,250
000424   242 CALL FIND(7,1,4)
000427       DO 243 I=1,NN
000431       NODE(ICTZ,1,I) = TEMP(I)
000437   243 CONTINUE
000441       IS4 = 1
000442   250 CONTINUE

```

```

C
C
C
C
C
C      FILL NODE ARRAY - JUMP THOSE POSITIONS. ALL READY FILLED
C
000442      DO 320 J=1,NN
000444      DO 310 I=1,MM
000445      IF(NODE(ICTZ,I,J))315,300,310
000453      300 CCNTINUE
000453      NODE(ICTZ,I,J) = ICTN
000461      ICTN = ICTN + 1
000462      GO TO 310
C
000463      315 CONTINUE
000463      WRITE(IOUT,316)
000467      316 FORMAT(48H NODE NO FOUND IN ST. N/. 300-320 LESS THAN ZERO)
000467      310 CONTINUE
000472      320 CONTINUE
000474      3000 CONTINUE
C
C
C      * COMPUTE THE X-Y COORDINATES, OMIT PREVIOUSLY COMPUTED SITES.
C
C      * DC, AND CN ARE THE INCREMENTAL VALUES IN THE M AND N
C      DIRECTIONS RESPECTIVELY.
C
000474      RM = M
000476      RN = N
000477      EC = 2./RM
000501      DN = 2./RN
C
C      33 WRITE(IOUT,33)DC, DN
C      33 FORMAT(1H ,4HDC= ,F5.2,4X,4HDN= ,F5.2)
C
000503      CCC = -1.
000504      CCN = -1.
C
000505      DO 810 J=1,NN
000506      IF(J-1)731,730,731
000510      730 IF(I S1-1)733,51,733
000513      731 IF(J-NN)733,732,733
000515      732 IF(I S3-1)733,51,733
000517      733 CCNTINUE
C

```

```

000517      DO 800 I=1,MM
C
000521      IF(I-MM)741,734,741
000523      734 IF(IS2-1)739,50,739
C
000526      741 IF(I-1)739,742,739
000530      742 IF(IS4-1)739,50,739
000532      739 CONTINUE
000532      S1 = 1.-CCC
000534      S2 = 1.-CCN
000536      S3 = CCC+CCN-1.
000540      S4 = 1.+CCC
000541      S5 = 1.+CCN
000543      A(1) = 1./4.*S1*S2*(-CCC-CCN-1.)
000552      A(2) = 1./2.*S1*S2*S4
000554      A(3) = 1./4.*S2*S4*(CCC-CCN-1.)
000563      A(4) = 1./2.*S4*S2*S5
000566      A(5) = 1./4.*S3*S4*S5
000571      A(6) = 1./2.*S1*S4*S5
000575      A(7) = 1./4.*S1*S5*(-CCC+CCN-1.)
000604      A(8) = 1./2.*S1*S2*S5
000610      NCOR = NCOR + 1
000611      DO 45 K=1,8
000613      X2(NCOR) = A(K) * XNODE(K) + X2(NCOR)
000617      Y2(NCOR) = A(K) * YNODE(K) + Y2(NCOR)
000622      45 CONTINUE
000624      WRITE(IOUT,2000) NCOR,X2(NCOR),Y2(NCOR)
000635      2000 FORMAT(I5,2X,F10.5,2X,F10.5)
000635      50 CONTINUE
000635      CCC = CCC+DC
000637      800 CONTINUE
000642      51 CONTINUE
000642      CCN = CCN + DN
000644      CCC = -1.
000646      810 CONTINUE
000650      WRITE(IOUT,54)
000654      54 FORMAT(///,1H ,15HNODE NO. MATRIX/)
000654      DO 53 J=1,NN
000656      WRITE(IOUT,52)(NODE(ICTZ,I,J),I=1,MM)
000673      52 FORMAT(1X,26I5)
000673      53 CONTINUE
000676      IF(ICTZ-IZ)1000,4000C,40000
000700      40000 CONTINUE

```

```

C
C   * LIST THE ELEMENT NUMBERS AND DEFINING NODE NUMBERS.
C
C
CC0700   WRITE(IOUT,900)
000704   900 FORMAT(1H1,7HELEMENT,8X,1H1,10X,1HJ,10X,1HK,10X,1HL/)
CC0704   CO 930 ICTZ=1,IZ
000706   M = IZONE(ICTZ,9)
CC0710   N = IZONE(ICTZ,10)
000711   CO 920 J=1,N
000713   CO 910 I=1,M
CC0714   II = I + 1
000716   JJ = J + 1
CC0717   INO = NODE(ICTZ,I,J)
CC0724   JNO = NODE(ICTZ,II,J)
000731   KNO = NODE(ICTZ,II,JJ)
000735   LNO = NODE(ICTZ,I,JJ)
000742   WRITE(IOUT,901)IELM,INO,JNO,KNO,LNO

C
C   PUNCHED OUTPUT IN FORMAT FOR USE IN PROGRAM SAP
000757   IF(IP .EQ. 1) PUNCH 902, IELM,INO,JNO,KNO,LNO

C
CC0777   901 FORMAT(I6,4(6X,15))
000777   902 FORMAT(5I5)
CC0777   IELM = IELM + 1
001001   910 CONTINUE
CC1003   920 CONTINUE
001006   930 CONTINUE

C
001010   CALL FEMPLT(IZ,INODE,NCOR)

C
C   * WRITE THE X AND Y COORDINATES OF THE NODE NUMBERS.
C
C
001013   WRITE(IOUT,940)
001017   940 FORMAT(1H1,4HNODE,11X,1HX,14X,1HY/)
CC1017   CO 950 I=1,NCCR
001021   WRITE(IOUT,941) INODE,X2(I),Y2(I)
CC1032   941 FORMAT(1H,15,2(5X,F10.4))
CC1032   INODE = INODE + 1
001034   950 CONTINUE

C
001036   *
CC1040   STOP
        END

```

```

SUBROUTINE FIND(LP1,LP2,ISD)
C
C SEARCHES PREVIOUSLY CALCULATED DATA TO FIND NODE NUMBERS ASSIGNED TO A
C ZONE BOUNDARY.
C
0C0006 COMMON ICTZ,IZONE(15,10),NODE(15,25,25),TEMP(25)
0C0006 COMMON X2(4002),Y2(4002)
C
C LP LOCATION OF PT. ON ZONE ICTZ
C
000006 IA = 0
000007 IT = 0
C
0C0010 DO 5 I=1,25
000011 TEMP(I) = 9999
000013 5 CCNTINUE
C
C DEFINE CORNER NODES ON ZONE ICTZ
C
0C0015 J1 = IABS(IZCNE(ICTZ,LP1))
000021 J2 = IABS(IZONE(ICTZ,LP2))
C
000024 IIZ = ICTZ
0C0025 10 IIZ = IIZ - 1.
000027 IF(IIZ)200,200,110
0C0030 110 CONTINUE
C
C SEARCH DATA OF ZONE IIZ
C
000030 DO 40 I=1,7,2
0C0032 II = I + 2
000034 IF(I-7)16,15,16
0C0036 15 II = 1
000037 16 CCNTINUE
000037 K1 = IABS(IZONE(IIZ,I))
000043 K2 = IABS(IZONE(IIZ,II))
C
C COMPARE ICTZ TO IIZ
C
0C0047 IF(J1-K1)30,20,30
000051 20 IF(J2-K2)40,21,40
000054 30 IF(J1-K2)40,31,40

```

```

000056      31 IF(J2-K1)40,21,40
000060      21 CONTINUE
000060      IA = (I+1)/2
000063      GO TO 41
000063      40 CONTINUE

C
000065      41 CONTINUE
000065      IF(IA)45,10,45.

C
C
C      PUT DESIRED CONTENTS CF IIZ IN TEMPORARY ARRAY
C
C      TEMP ARRAY MUST HAVE REVERSE ORDER IF ...
C      ISD = 1 OR 2   AND   IA = 1 OR 2
C      - OR -
C      ISD = 3 OR 4   AND   IA = 3 OR 4
C
C
000066      45 CONTINUE
000066      MMT = IZONE(IIZ,9) + 1
000071      NNT = IZONE(IIZ,10) + 1
000072      MK = MMT
000074      NK = NNT
000075      II = 0

C
000076      DO 100 I=1,25

C
000077      IF(IA-1)46,50,46
000101      46 IF(IA-2)47,60,47
000103      47 IF(IA-3)48,70,48
000105      48 IF(IA-4)100,80,100

C
000110      50 CONTINUE
000110      MK = I
000112      NK = 1
000113      IF(ISC-2)90,90,92

C
000116      60 CONTINUE
000116      NK = I
000120      IF(ISD-2)91,91,92

C
000123      70 CONTINUE

```

```

000123      MK = I
000125      IF(ISO-2)92,92,50

      C
000130      80 CONTINUE
000130      MK = 1
000131      NK = I
000133      IF(ISC-2)92,92,91

      C
      C
000136      90 II = MMT + 1 - I
000141      GO TO 93
000141      91 II = NNT + 1 - I
000144      GO TO 93
000144      92 II = I
000146      93 CONTINUE
000146      IF(II)200,200,94
000150      94 CONTINUE

      C
000150      TEMP (II) = NODE( IIZ, MK, NK )
000157      100 CONTINUE
000161      200 CONTINUE
000161      RETURN
000162      END

```

```

SUBROUTINE FEMPLT(IZ,INODE,NCOR)
C
C
C
C
000006      COMMON ICTZ,IZONE(15,10),NODE(15,25,25),TEMP(25)
000006      COMMON X2(4002),Y2(4002)
C
C
C
000006      SCALE DATA
000006      CALL ASCALE(X2,25.,NCOR,1,20.)
000011      CALL ASCALE(Y2,13.,NCOR,1,20.)
C
000016      XSCALE = X2(NCOR + 2)
000022      YSCALE = Y2(NCOR + 2)
C
000023      IF(XSCALE .GE. YSCALE) SF=XSCALE
000027      IF(YSCALE .GE. XSCALE) SF=YSCALE
C
C
000032      DO 100 ICTZ=1,IZ
000034      M = IZONE(ICTZ,9)
000036      N = IZONE(ICTZ,10)
C
000037      DO 90 J=1,N
000041      DO 80 I=1,M
C
000042      II = I + 1
000044      JJ = J + 1
C
C
C
000045      INO = NODE(ICTZ,I,J) - INODE + 1
000053      JNO = NODE(ICTZ,II,J) - INODE + 1
000061      KNO = NODE(ICTZ,II,JJ) - INODE + 1
000067      LNO = NODE(ICTZ,I,JJ) - INODE + 1
C
C
C
000075      XI = X2(INO)/SF
000100      XJ = X2(JNO)/SF
000102      XK = X2(KNO)/SF

```



```

000104      XL = X2(LND)/SF
      C
000106      YI = Y2(INO)/SF
000110      YJ = Y2(JNC)/SF
000112      YK = Y2(KND)/SF
000115      YL = Y2(LNC)/SF
      C
      C      PLOT THE 4 NODES.
      C
000117      CALL CALPLT(XI,YI,3)
000121      CALL CALPLT(XJ,YJ,2)
000124      CALL CALPLT(XK,YK,2)
000127      CALL CALPLT(XL,YL,2)
000132      CALL CALPLT(XI,YI,2)
      C
000135      80 CONTINUE
000142      90 CONTINUE
000144      100 CONTINUE
000147      RETURN
000150      END

```

REFERENCES

1. W.R. Buell and B.A. Bush, "Mesh Generation - A Survey" an ASME publication, Paper No. 73-WA/DEZ.
2. O.C. Zienkiewicz and D.R. Phillips, "An Automatic Mesh Generation Scheme for Plane and Curved Surfaces by 'Isoparametric' Coordinates", International Journal for Numerical Methods in Engineering, Vol. 3, 519-528, 1974.
3. S.J. Womack, "Shape Function Techniques for Generation of Finite Element Grids", TICOM Report 73-3, August 1973.
4. R.S. Dunham and E.B. Becker, "TEXGAP - The Texas Grain Analysis Program", TICOM Report 73-1, August 1973.
5. O.C. Zienkiewicz, The Finite Element Method in Engineering Science, McGraw-Hill, 1971.

Table 1. The shape functions.

$$x = \sum_{i=1}^8 N_i x_i$$

$$y = \sum_{i=1}^8 N_i y_i$$

$$N_1 = -\frac{1}{4} (1 - \xi) (1 - \eta) (\xi + \eta + 1)$$

$$N_2 = \frac{1}{2} (1 - \xi^2) (1 - \eta)$$

$$N_3 = \frac{1}{4} (1 + \xi) (1 - \eta) (\xi - \eta - 1)$$

$$N_4 = \frac{1}{2} (1 + \xi) (1 - \eta^2)$$

$$N_5 = \frac{1}{4} (1 + \xi) (1 + \eta) (\xi + \eta - 1)$$

$$N_6 = \frac{1}{2} (1 - \xi^2) (1 + \eta)$$

$$N_7 = \frac{1}{4} (1 - \xi) (1 + \eta) (-\xi + \eta - 1)$$

$$N_8 = \frac{1}{2} (1 - \xi) (1 - \eta^2)$$

Table 2. Input Data for Example Problem Shown in Figure 5.

EXAMPLE PROBLEM									
3	18	8	100	1000					
1	2	3	7	11	10	9	6	3	3
-3	4	5	8	13	12	-11	-7	2	3
-11	-12	-13	15	18	17	16	14	2	4
1		0.		0.					
3		3.		0.					
5		5.		0.					
9		0.		3.					
11		3.		3.					
13		5.		3.					
16		3.		7.					
18		5.		7.					

C-2

Table 3. Output for Example Problem Shown in Figure 5.

EXAMPLE PROBLEM

NO. OF ZONES 3

ZONE NO.	ZONE NODES								M	N
	1	2	3	4	5	6	7	8		
1	1	2	3	7	11	10	9	6	3	3
2	-3	4	5	8	13	12	-11	-7	2	3
3	-11	-12	-13	15	18	17	16	14	2	4

NODE	X	Y
1	0.000	0.000
3	3.000	0.000
5	5.000	0.000
9	0.000	3.000
11	3.000	3.000
13	5.000	3.000
16	3.000	7.000
18	5.000	7.000

(cont'd.)

Table 3. Output for Example Problem Shown in Figure 5 (continued).

ELEMENT	I	J	K	L
1000	100	101	105	104
1001	101	102	106	105
1002	102	103	107	106
1003	104	105	109	108
1004	105	106	110	109
1005	106	107	111	110
1006	108	109	113	112
1007	109	110	114	113
1008	110	111	115	114
1009	103	116	118	107
1010	116	117	119	118
1011	107	118	120	111
1012	118	119	121	120
1013	111	120	122	115
1014	120	121	123	122
1015	115	122	125	124
1016	122	123	126	125
1017	124	125	128	127
1018	125	126	129	128
1019	127	128	131	130
1020	128	129	132	131
1021	130	131	134	133
1022	131	132	135	134

(cont'd.)

Table 3. Output for Example Shown in Figure 5 (concluded).

MODE	X	Y
100	0.0000	0.0000
101	1.0000	0.0000
102	2.0000	0.0000
103	3.0000	0.0000
104	0.0000	1.0000
105	1.0000	1.0000
106	2.0000	1.0000
107	3.0000	1.0000
108	0.0000	2.0000
109	1.0000	2.0000
110	2.0000	2.0000
111	3.0000	2.0000
112	0.0000	3.0000
113	1.0000	3.0000
114	2.0000	3.0000
115	3.0000	3.0000
116	4.0000	0.0000
117	5.0000	0.0000
118	4.0000	1.0000
119	5.0000	1.0000
120	4.0000	2.0000
121	5.0000	2.0000
122	4.0000	3.0000
123	5.0000	3.0000
124	3.0000	4.0000
125	4.0000	4.0000
126	5.0000	4.0000
127	3.0000	5.0000
128	4.0000	5.0000
129	5.0000	5.0000
130	3.0000	6.0000
131	4.0000	6.0000
132	5.0000	6.0000
133	3.0000	7.0000
134	4.0000	7.0000
135	5.0000	7.0000

Table 4. Input for Shear Panel.

SHEAR PANEL									
4	21	17	1	1000					
1	2	3	4	5	8	11	10	20	10
-11	-8	-5	6	7	9	13	12	20	3
-1	-10	-11	15	19	18	17	14	10	20
-11	-12	-13	16	21	20	-19	-15	3	20
1	0.	0.							
2	3.0	0.							
3	4.35	0.							
4	4.3727	.1040							
5	4.435	.185							
7	4.0	.35							
8	3.825	.825							
9	3.975	.975							
11	2.31	2.31							
13	2.475	2.475							
14	0.	3.							
15	.825	3.825							
16	.975	3.975							
17	0.	4.35							
18	.1040	4.3727							
19	.185	4.435							
21	.350	4.0							

Table 5. Input for Bolted Joint Specimen.

BOLTED JOINT COMPOSITE STRUCTURE									
15	66	41	100	1000					
28	29	30	21	3	2	1	20	2	3
-30	22	7	6	5	4	-3	-21	5	3
-7	-22	-30	31	32	23	9	8	5	6
-9	-23	-32	24	13	12	11	10	5	3
-30	41	57	56	55	42	-32	-31	5	6
49	50	51	58	-57	-41	-30	40	5	6
47	48	-49	-40	-30	-29	-28	39	2	6
-51	-50	-49	00	61	59	53	52	5	6
-53	-59	-61	43	-32	-42	-55	54	5	6
-61	62	63	44	34	33	-32	-43	3	6
-32	-33	-34	25	15	14	-13	-24	3	3
-34	35	36	26	17	16	-15	-25	11	3
-36	37	38	27	19	18	-17	-26	6	3
-63	64	65	45	-36	-35	-34	-44	11	6
-65	66	67	46	-38	-37	-36	-45	6	6
1	0.		1.5						
3	0.13		1.5						
4	0.4290		1.5						
5	0.5345		1.5						
6	0.5418		1.4635						
7	0.5625		1.4325						
8	0.63		1.4045						
9	0.6975		1.4325						
10	0.7182		1.4635						
11	0.7255		1.5						
12	0.8308		1.5						
13	1.13		1.5						
15	1.5		1.5						
17	3.3		1.5						
19	4.5		1.5						
22	0.4292		1.284						
23	0.8308		1.284						
28	0.		1.0						
30	0.13		1.0						
32	1.13		1.0						
34	1.5		1.0						
36	3.3		1.0						
38	4.5		1.0						
41	0.4292		0.7008						
42	0.8308		0.7008						
47	0.		0.						
49	0.13		0.						
50	0.4292		0.2992						
51	0.5625		0.4325						
52	0.63		0.4045						
53	0.6975		0.4325						
54	0.7255		0.5						
55	0.6975		0.5675						
56	0.63		0.5955						
57	0.5625		0.5675						
58	0.5345		0.5						
59	0.8308		0.2992						
61	1.13		0.						
63	1.5		0.						
65	3.3		0.						
67	4.5		0.						

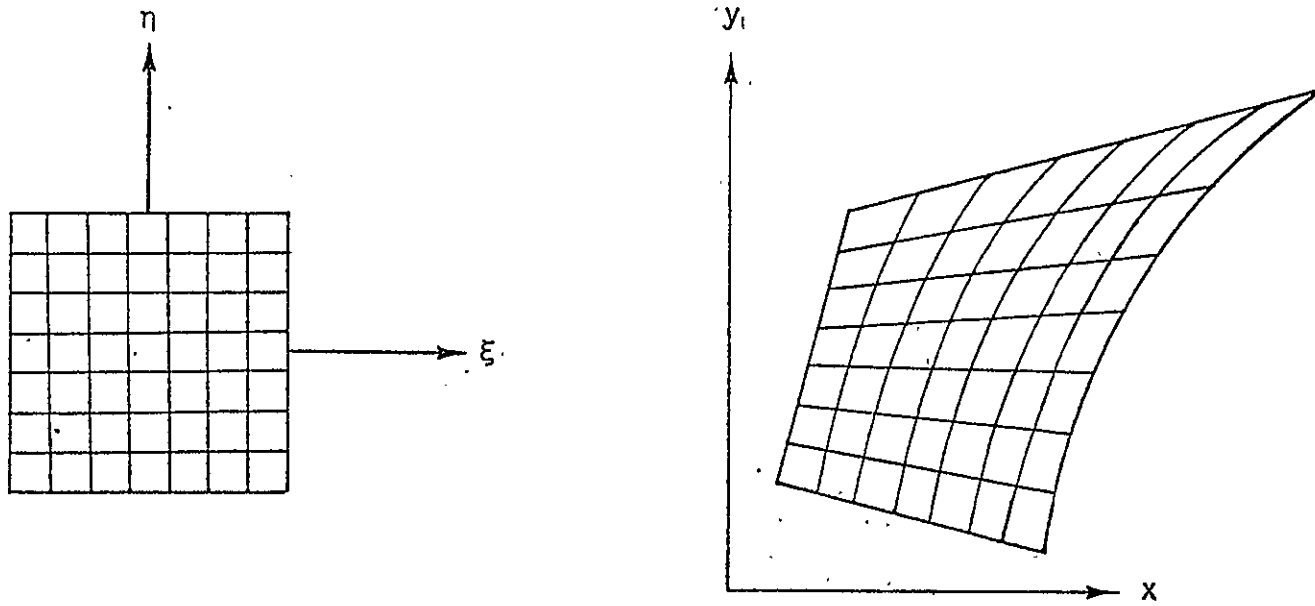
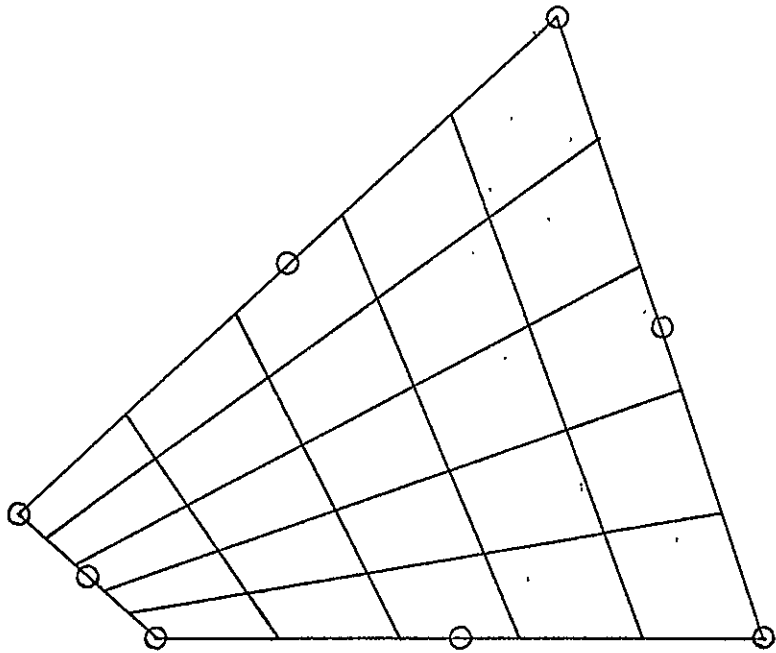
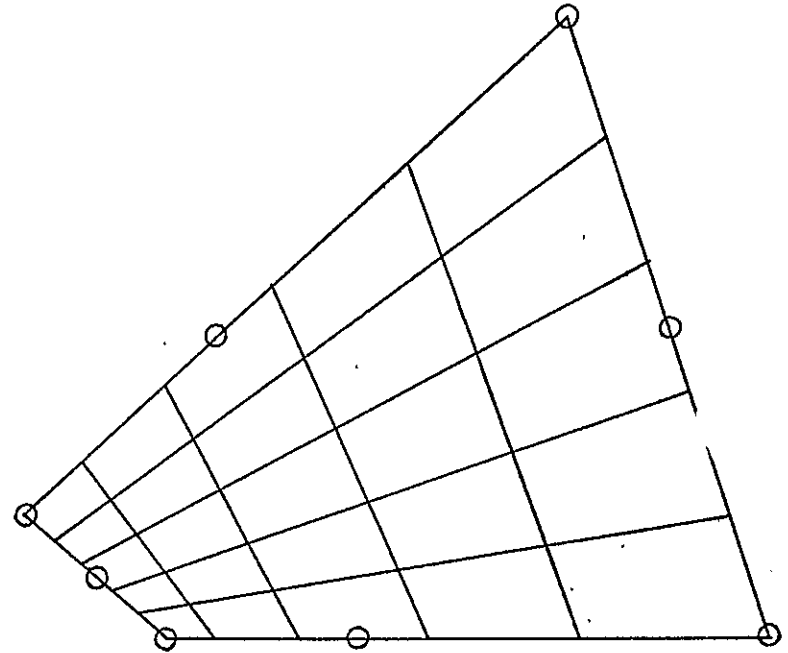


Figure 1. Mapping of a Quadrilateral from the Natural to the Cartesian Coordinate System.



(a) UNGRADED MESH



(b) GRADED MESH

Figure 2. Ungraded and Graded Mesh Generated in the Cartesian System.

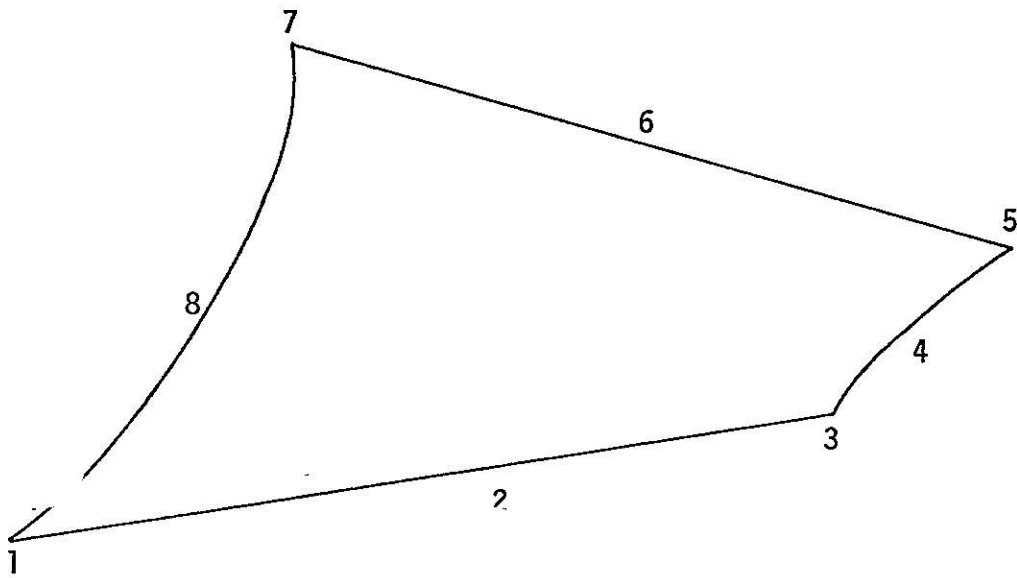


Figure 3. Numbering of Zone-Nodes.

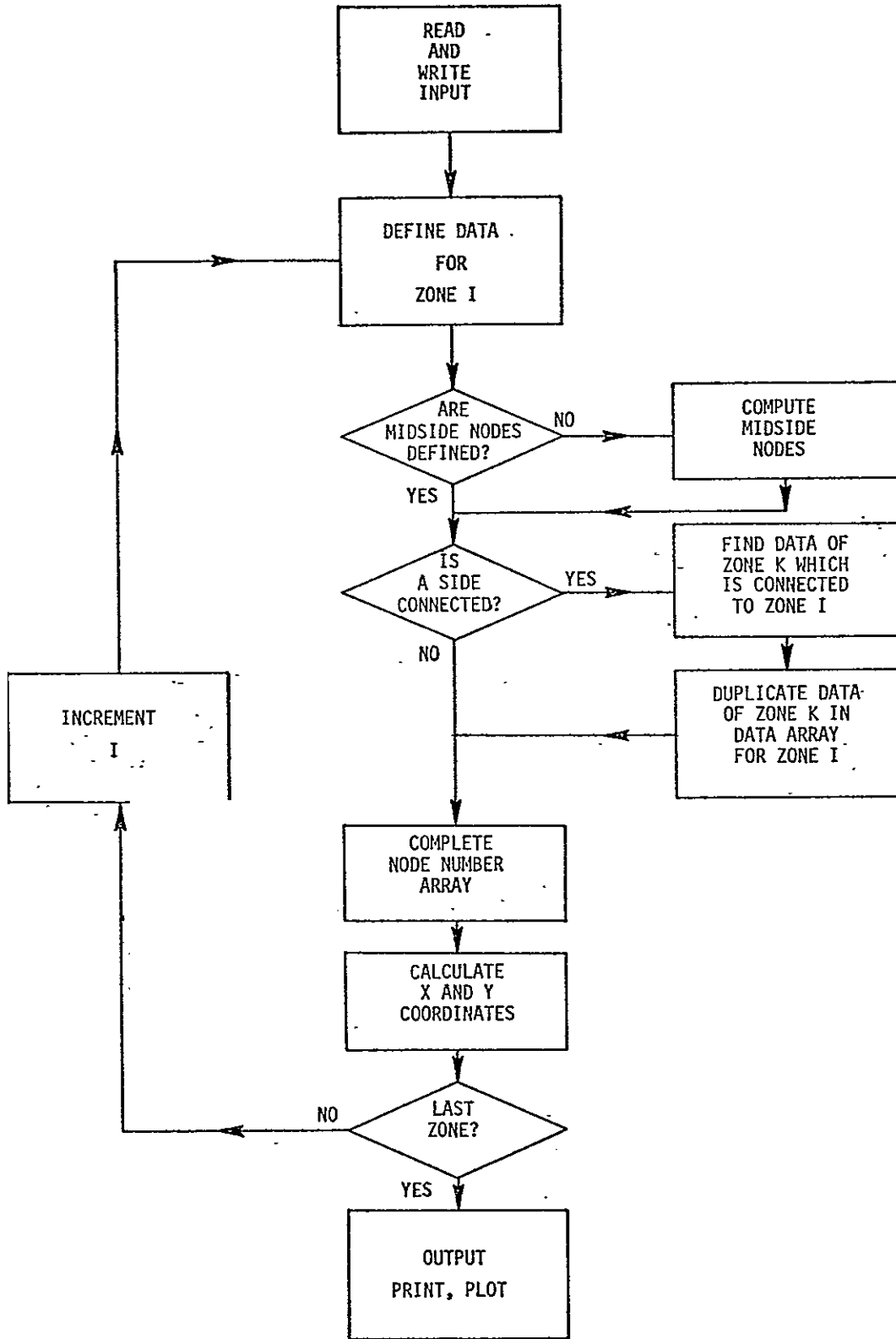
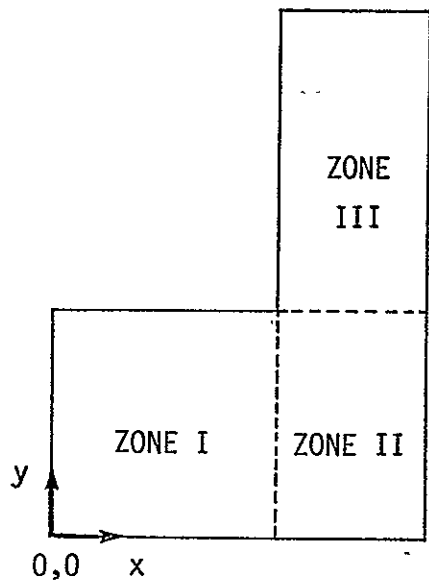
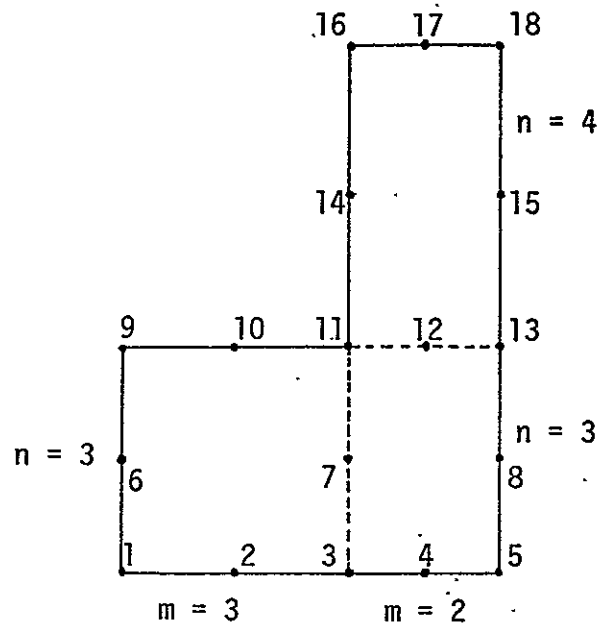


Figure 4. Flowchart for Mesh Generation Program, FEMESH.



(a) STRUCTURE AND ZONES



(b) ZONE NODE

			133	134	135	
			130	131	132	
			127	128	129	
			124	125	126	
112	113	114	115	122	123	
108	109	110	111	120	121	
104	105	106	107	118	119	
100	101	102	103	116	117	

(c) NODE NUMBERS

			1021	1022
			1019	1020
			1017	1018
			1015	1016
1006	1007	1008	1013	1014
1003	1004	1005	1011	1012
1000	1001	1002	1009	1010

(d) ELEMENT NUMBERS

Figure 5. Example of Mesh Generation for a Simple Structure.

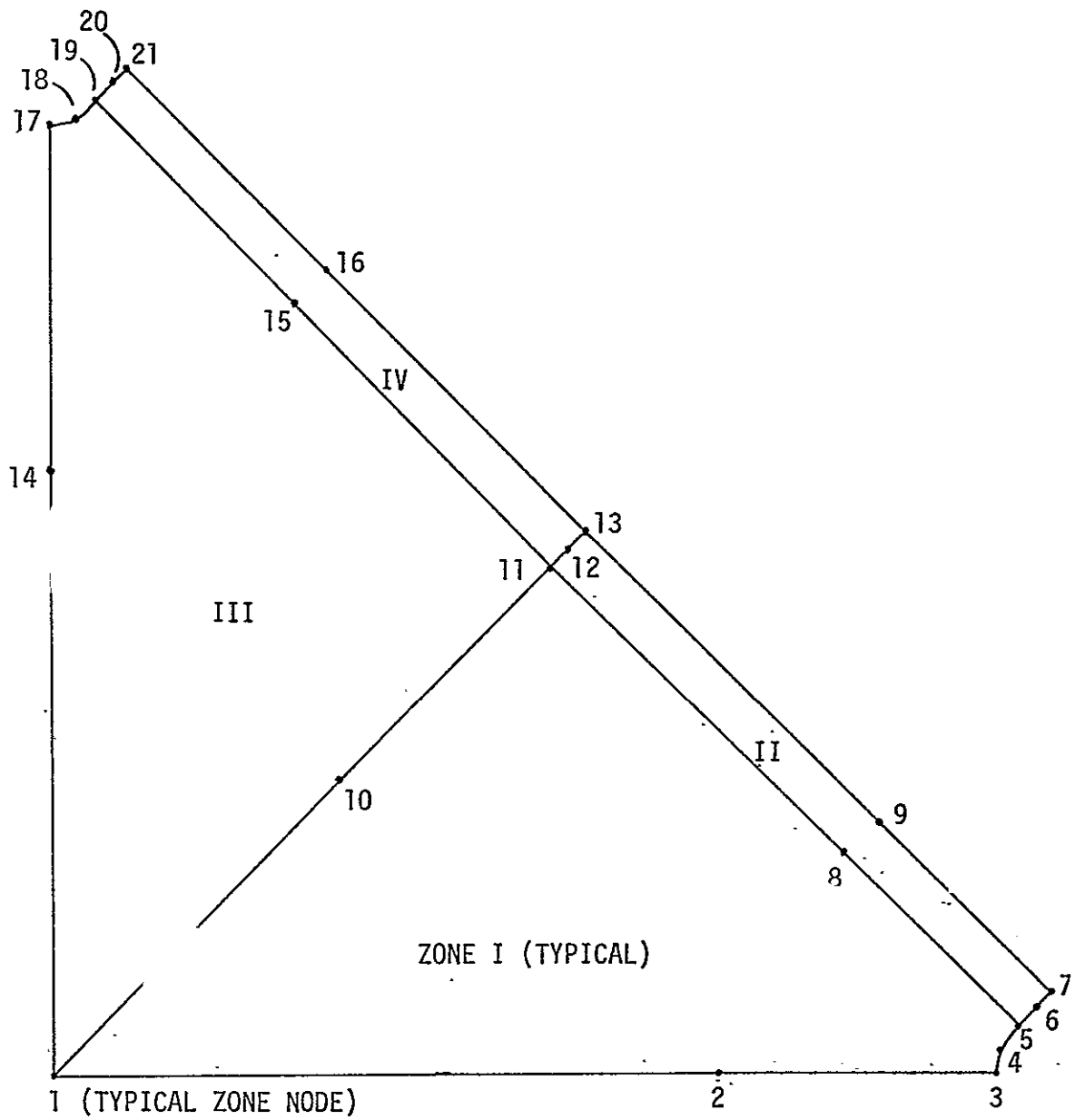


Figure 6. Zones of Shear Panel.

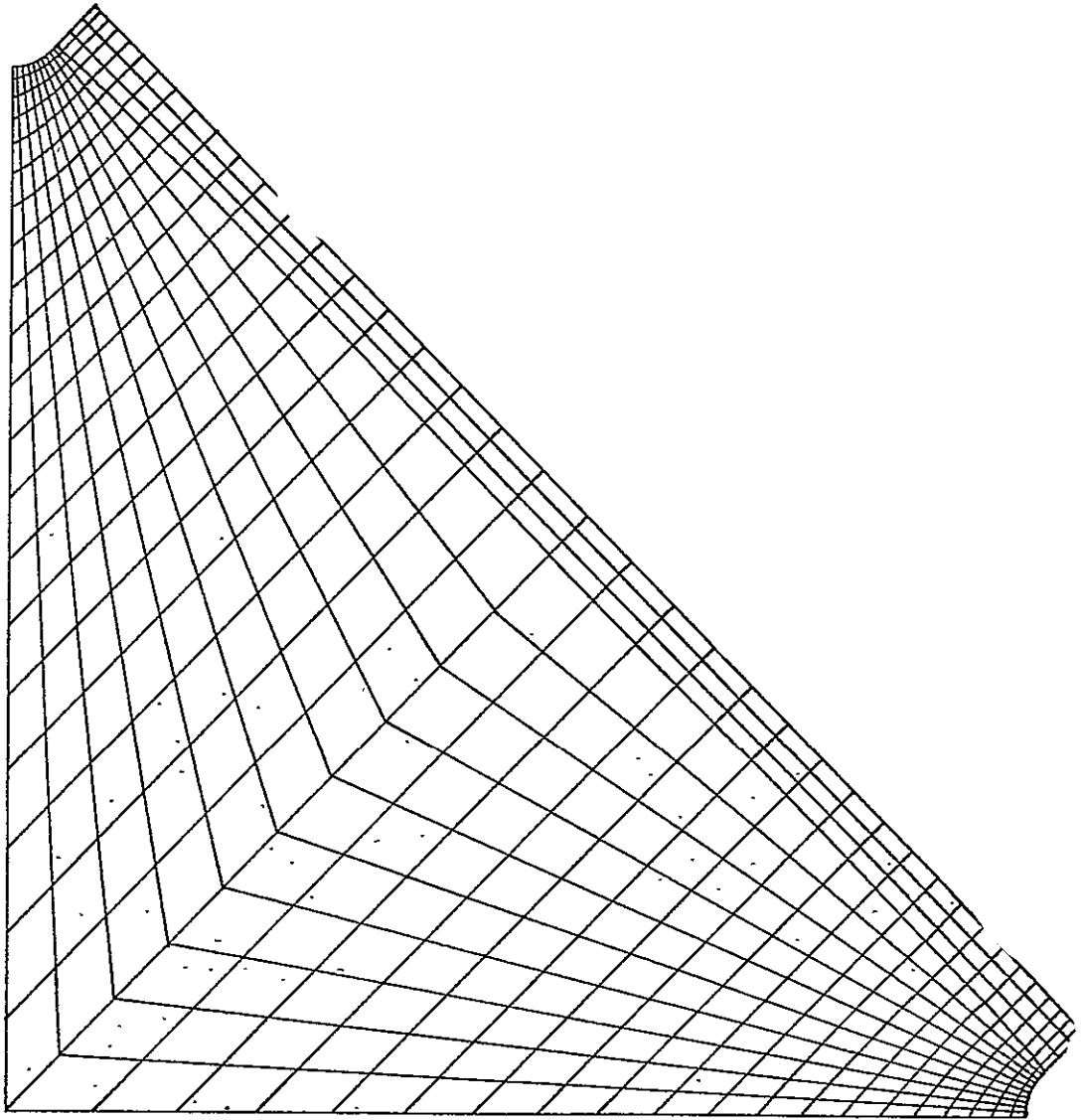


Figure 7. Mesh Generated for Quartersection of Shear Panel.

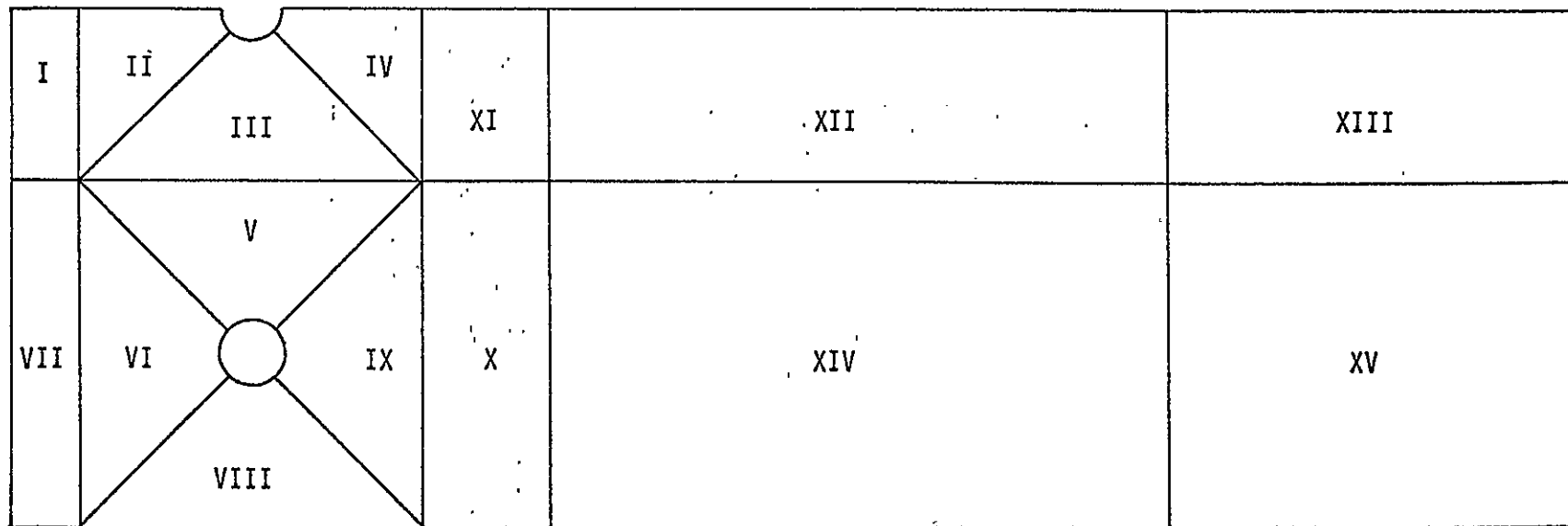


Figure 8. Zones of Bolted Joint Specimen.

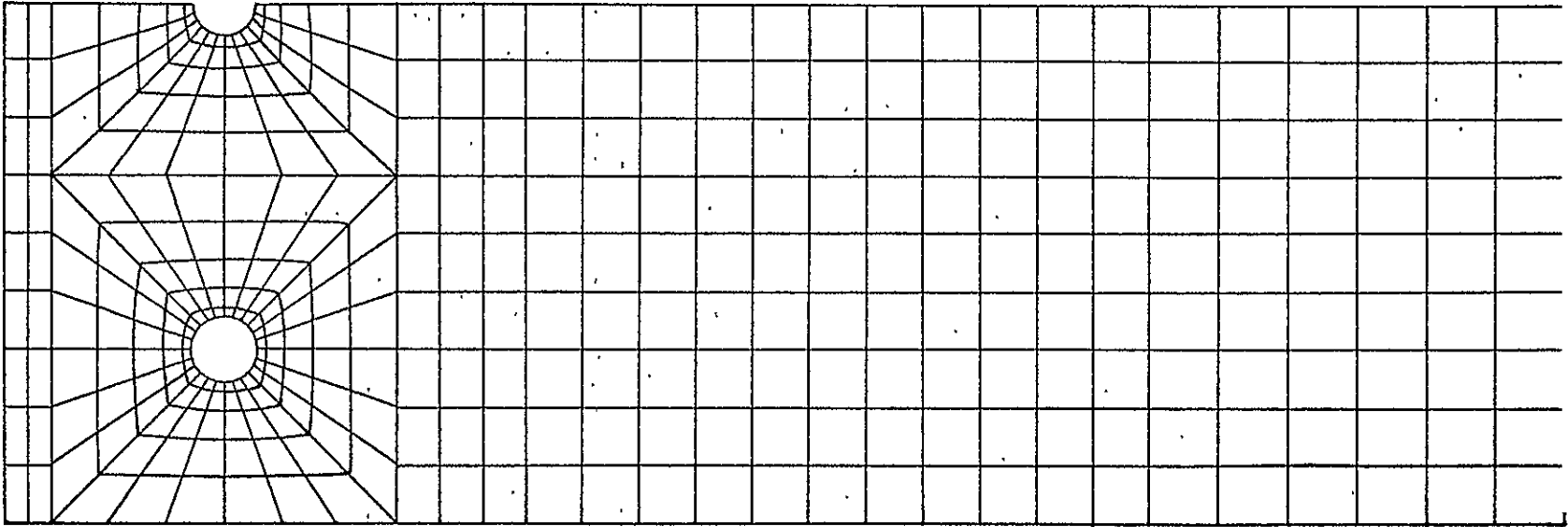


Figure 9. Mesh Generated for Half Section of Bolted Joint Specimen.

FINITE ELEMENT ANALYSIS OF A COMPOSITE
BOLTED JOINT SPECIMEN

By

Earl A. Thornton

FINITE ELEMENT ANALYSIS OF A COMPOSITE BOLTED JOINT SPECIMEN

By

Earl A. Thornton

INTRODUCTION

With high strength and weight savings, advanced composite materials have become increasingly important in aircraft structural design. The full potential for the increase of structural efficiency through the use of advanced composites has not yet been fully realized because of low efficiencies in mechanical joints. The Advanced Composites Design Guide (ref. 1) states that weight savings may be reduced by as much as 40 percent due to such practical constraints.

In the use of conventional materials, design methods for joints have evolved over a period of time from data gathered from experimental and analytical solutions and, in addition, are often based upon rules-of-thumb derived from experience. For advanced composites, such data and experience are relatively limited. To partially fill this need, test programs are underway at Langley Research Center (LRC) to establish data on a number of mechanical joint designs (ref. 2)

The purpose of the present study was to provide analytical support for the LRC bolted joint test program. Specific objectives of the study were to: (1) determine the laminate stress distribution in an extra graphite reinforced bolted joint specimen, and (2) compare two methods of modeling bolt transfer loads for determination of stress distributions in bolted joints.

This paper will describe the finite element model used to represent the bolted joint specimen. The two methods used to represent bolt transfer loads will be discussed. Laminate membrane force distributions predicted by the finite element

analysis will be presented, and force gradients at the bolt holes will be discussed. Differences in the results due to the methods of representing the bolt loads will also be discussed.

BOLTED JOINT SPECIMEN

The specimen analyzed in this study is the specimen denoted as extra graphite reinforced joint specimen number one, reference 2. The specimen is shown schematically in figure 1 with the dimensions used in the analysis. The specimen was fabricated from a basic layup of 15 plies reinforced by additional plies so that in the thick section where the bolt holes are located there are 49 plies. The ply stacking sequences are shown in figure 2 with cross-sectional details of the layup. Reinforcing plies increase by 0.1 in. in length per ply over the transition section from 49 plies to 15 plies.

ANALYTICAL PROCEDURES

Finite Element Model

The NASA Structural Analysis (NASTRAN) computer program (level 15.5) was used to compute the laminate stress distributions in the specimen. The specimen was assumed to be in-plane stress and due to symmetry only one-half of the specimen was represented with finite elements. The finite element representation is shown in figure 3. The specimen was represented by an assemblage of 349 quadrilateral and triangular membrane elements. The NASTRAN finite elements used have constant stress throughout each element. The mathematical model has 307 grid points and 573 degrees of freedom. Vertical displacements were set to zero on the top boundary of the finite element model to represent symmetry, and horizontal displacements at the right edge of the finite element model were set to zero to represent clamping in the test fixture.

In the analytical formulation underlying the present NASTRAN elements the element material is assumed homogeneous through the

thickness. The element extensional stiffnesses are obtained internally in NASTRAN by multiplying the material elasticity matrix by the thickness of the element, reference 3. However, the specimen in the present study is characterized by several layers of material which are assumed homogeneous within the individual layers only. Thus for the composite laminate the extensional stiffnesses, A_{ij} , were computed externally using laminated plate theory. The stiffnesses were then input to NASTRAN in place of the material elasticity matrix, and the thickness of the specimen was everywhere taken as unity.

The extensional stiffnesses A_{ij} , a 3 x 3 symmetric matrix, were computed from reference 4:

$$A_{ij} = \sum_{k=1}^N (Q_{ij})_k (z_k - z_{k-1}) \quad (1)$$

where $(Q_{ij})_k$ denotes the material elasticity matrix for a single layer and $(z_k - z_{k-1})$ denotes the thickness of the kth layer. The extensional stiffnesses relate the in-plane membrane forces (N_x, N_y, N_{xy}) to the midplane extensional strains $(\epsilon_x, \epsilon_y, \gamma_{xy})$ of the laminate. Since the extensional stiffnesses were input to NASTRAN in place of the NASTRAN material elasticity matrix, the NASTRAN membrane element stresses $(\sigma_x, \sigma_y, \gamma_{xy})$ were the laminate stress resultants (N_x, N_y, N_{xy}) .

In the present analysis the lamina elastic constants were taken as $E_{11} = 20 \times 10^6$ psi, $E_{22} = 2 \times 10^6$ psi, $G = 0.8 \times 10^6$ psi and $\nu_{12} = 0.3$. Each lamina had a thickness of 0.00542 in. To represent the tapered character of the specimen, extensional stiffnesses were computed for the 19 different cross-sectional layups. The values of the extensional stiffnesses for the specimen are given in table 1.

Bolt Loads

The specimen was analyzed for loading corresponding to the design failure load. This loading, estimated at 21 813 lb was assumed to be equally distributed to the three bolts such that the total load transmitted to the specimen per bolt was 7271 lb. In the finite element model, one-half of this load was applied to the center bolt hole and the full value was applied to the lower bolt hole.

Two methods were used to represent the transfer of the bolt forces to the finite element model. In the first approach the bolt was assumed to have a perfect fit, and the load transfer was assumed to take place over 180° of the bolt hole. The contact force was assumed to vary sinusoidally over this area of contact. Equilibrium of the bolt was then used to obtain the relation:

$$N = \frac{2}{\pi} \frac{2Q}{R} \cos \theta \quad (2)$$

where N denotes the contact force per unit arc length, $2Q$ is the total bolt load, and R is the radius of the bolt hole. The angle θ is measured from a horizontal axis through the hole. Equation (2) was used to compute equivalent grid point forces for each grid point in the contact region, ($-90^\circ < \theta < 90^\circ$). The equivalent grid point forces were computed by integrating Equation (2) through an angle of -6° to $+6^\circ$ at each grid point. The equivalent grid point loads are shown in figure 4.

In the second approach an imperfect fit was assumed and a nonlinear analysis of the bolt transfer loads was made. This analysis, made using the computer program CONTACT developed in reference 5, consists of increasing the bolt load in increments and determining the number of grid points in contact and their loads at each load increment. The analysis requires as part of its input the flexibility matrix for the bolt hole. This flexibility matrix was obtained from the finite element model by applying unit loads at each node of the center bolt hole. The

16 x 16 flexibility matrix was computed one column at a time for 16 unit load subcases. This matrix was then input to the CONTACT program and the bolt transfer forces were computed for several load increments. The bolt transfer forces and the region of contact for four load increments including the maximum load are shown in figure 5. These forces were computed using an initial lack of fit of -0.00287 in. This value, as defined in the program, denotes a clearance based upon the radius of the hole.

RESULTS AND DISCUSSION

The membrane force distributions at the center and outside bolt holes as predicted by the finite element analysis are shown in figures 6 through 8. Shown are plots of the radial force N_r , the circumferential force N_θ , and the in-plane shearing force $N_{r\theta}$ versus the angle θ from the centerline. Predictions based upon the two methods of representing the bolt transfer loads are compared.

There is very little, if any, difference in the membrane forces between the center bolt hole and the outside bolt holes. Each bolt was assumed to carry the same bolt load and there appears to be no interaction effects between holes nor edge effects upon the stress distributions in the outside holes. The magnitudes and variations of the membrane forces and the effects of the two methods of representing the bolt transfer loads can thus be discussed with regard to either hole.

The largest radial force intensity (fig. 6) occurs, as might be expected, on the centerline of the bolt hole. The nonlinear bolt loading method predicts the largest radial membrane forces with a value of 32 kips/in. compression which is about 23 percent higher than the value based upon the cosine bolt loading. The largest circumferential membrane force (fig. 7) of 30 kips/in. tension occurs at an angle of about 75° from the bolt centerline and is also predicted by the nonlinear bolt loading technique.

This stress is about 15 percent higher than the value based upon the cosine bolt loading. The in-plane membrane shear forces (fig. 8) tend to be smaller than the radial or circumferential membrane forces. The largest membrane shear force is about 9 kips/in. and is due to the nonlinear bolt loading. Since the in-plane shearing forces tend to be small, the principal values (not shown) of the membrane forces correspond in magnitude and location to the maximum radial and circumferential membrane forces.

The distribution of the longitudinal membrane force, N_x , along the specimen centerline is shown in figure 9. At $x = 0$ the membrane force should be zero since this edge is stress free; the small nonzero value is indicative of the error in the finite element solution. The membrane force at the left edge of the bolt hole ($x = 0.5$) rises very sharply due to the indirect bearing load of the bolt. On the right side of the bolt hole, the force should also be zero since the bolt is not in contact at this point. The finite element solution tends to zero at this point. Away from the hole for increasing x , the membrane force approaches a uniform value given by the total applied force (21,816 lb), divided by the specimen width (3 in.).

Further insight into the results of the finite element analyses can be obtained by considering an elasticity solution for an isotropic medium. In reference 6, Bickley presents the plane stress elasticity solution for a hole in an infinite medium loaded by a cosine pressure distribution over one-half of the boundary of the hole. Closed form solutions for the stress components are given in polar coordinates in terms of the radius of the hole and Poisson's ratio. Tabulated data of the stress components for Poisson's ratio of 0.25 are also presented.

In figure 10 are shown the membrane force distributions predicted by Bickley for an infinite isotropic medium with a hole equal in radius to the bolt hole in the composite specimen and loaded by the bolt load used in the finite element analysis. The plots are made for $r/a = 1.2$ which corresponds closely to

laminated composite material was represented in NASTRAN as a homogeneous material with equivalent extensional stiffness. Laminate membrane force distributions were predicted.

Comparison of the two methods of representing the bolt transfer loads showed the two methods were in qualitative agreement. The nonlinear analysis estimated membrane forces about 20 to 25 percent higher than the linear analysis. Peak forces were found to be a radial compressive force on the bolt centerline and a circumferential tensile force of the same magnitude at about 70° from the centerline. In-plane shear forces were found to be relatively small. There were little or no interaction effects between holes or boundaries of the specimen. Comparison of the finite element solution with an isotropic elasticity solution suggests that as a rule these effects will not be important for in-plane membrane forces provided offset distances between holes or edges are greater than five hole radii.

REFERENCES

1. Advanced Composites Design Guide, Third Edition, Air Force Materials Laboratory, Wright-Patterson Air Force Base, Ohio, Chapter 1.3.
2. "Design and Fabrication of Mechanical Joint Specimens for Graphite-Epoxy Composites", by Hoffman, D.J. and June, R.R., Boeing Commercial Aircraft Company Report D6-41817, May 1974.
3. The NASTRAN Theoretical Manual, edited by R.H. MacNeal, NASA SP-221, September 1970, pp. 5.8-2 to 5.8-4.
4. Primer on Composite Materials: Analysis, by Ashton, J.E., Halpin, J.C., and Petit, P.H., Technomic Publishing Company, 1969.
5. "Stress and Deflection Analysis of Mechanically Fastened Joints", by Harris, H.G., Ojalvo, I.U., and Hooson, R.E. of Grumman Aerospace Corporation, Air Force Flight Dynamics Laboratory Report AFFDL-TR-70-49, May 1960.
6. "The Distribution of Stress Round a Circular Hole in a Plate", by Bickley, W.G., Phil. Trans. Royal Society (London) volume 227A, 1928, pp. 383-415.

Table 1. Extensional stiffnesses.

Section	$A_{ij} \times 10^{-6}$ (lb/in.)					
	A_{11}	A_{12}	A_{13}	A_{22}	A_{23}	A_{33}
1	3.01	0.932	0	1.34	0	0.984
2	2.94	0.877	-0.0492	1.27	-0.0492	0.927
3	2.87	0.823	0	1.20	0	0.870
4	2.65	0.816	0	1.17	0	0.861
5	2.43	0.809	0	1.15	0	0.853
6	2.36	0.755	0.0492	1.08	0.0492	0.796
7	2.28	0.699	0	1.01	0	0.739
8	2.07	0.693	0	0.986	0	0.730
9	2.00	0.639	-0.0492	0.914	-0.0492	0.673
10	1.92	0.584	0	0.841	0	0.616
11	1.85	0.529	-0.0492	0.769	-0.0492	0.560
12	1.78	0.474	0	0.697	0	0.503
13	1.71	0.419	-0.0492	0.625	-0.0492	0.446
14	1.63	0.364	0	0.553	0	0.389
15	1.42	0.358	0	0.531	0	0.380
16	1.34	0.303	0.0492	0.459	0.0492	0.323
17	1.13	0.297	0.0492	0.437	0.0492	0.314
18	1.05	0.242	0	0.365	0	0.258
19	1.05	0.242	0	0.365	0	0.258

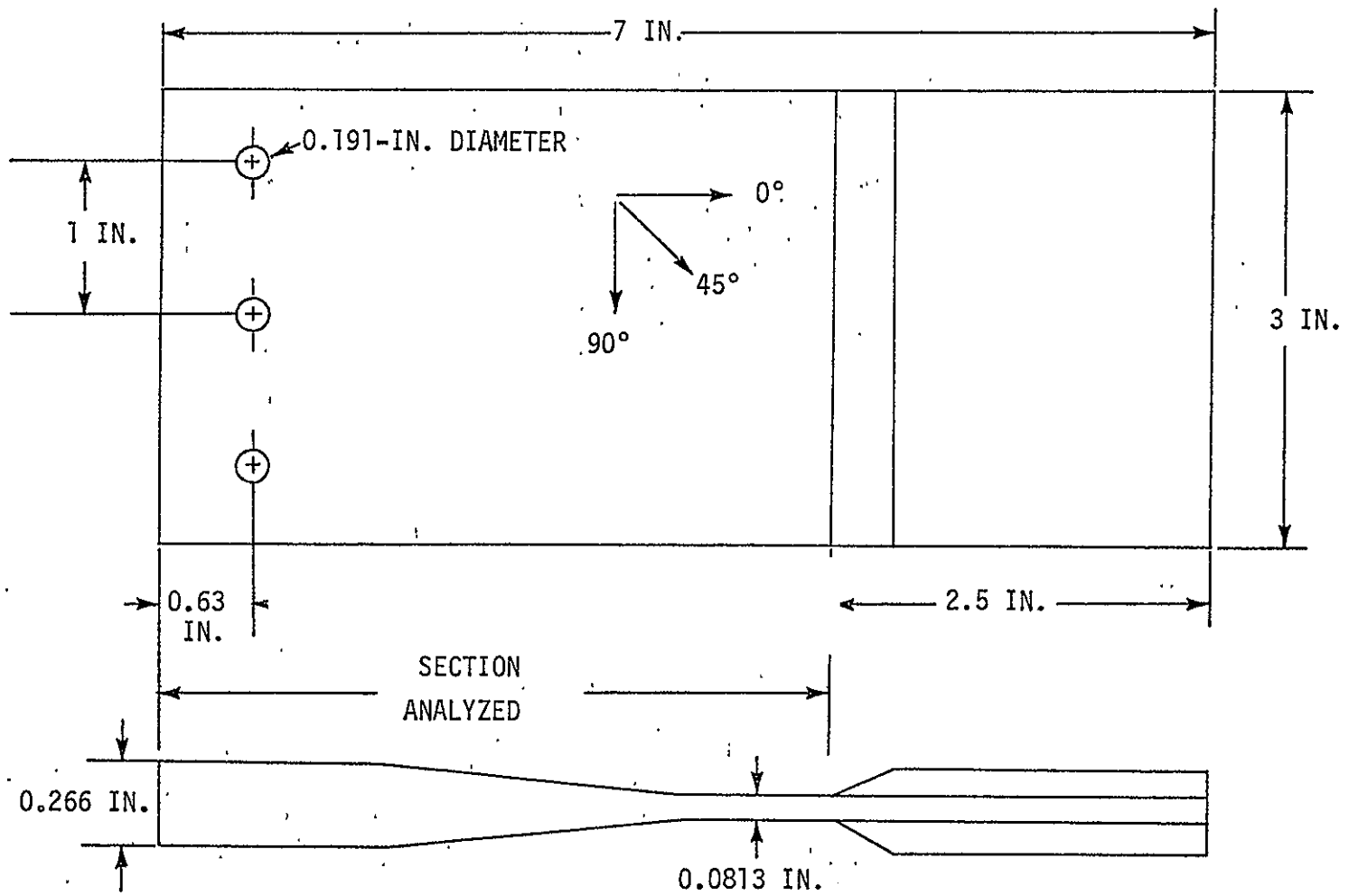
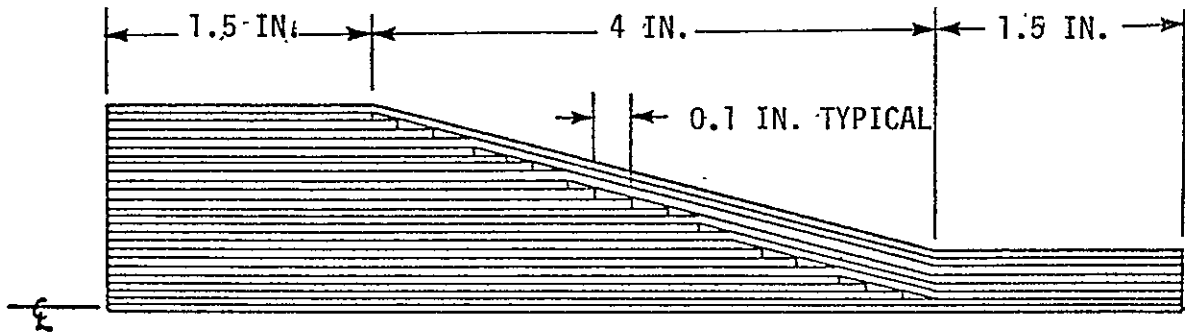


Figure 1. Extra Graphite Reinforced Joint Specimen.



ORIGINAL LAMINATE PLYS	REINFORCING PLYS
0	
	+45
	-45
	0
+45	
	0
	-45
	+45
-45	
	0
	+45
	-45
0	
	+45
	-45
0	
	+45
	-45
	0
+45	
	-45
	0
	+45
-45	
0	

Figure 2. Ply Stacking Sequence for Graphite Reinforced Specimen 1.

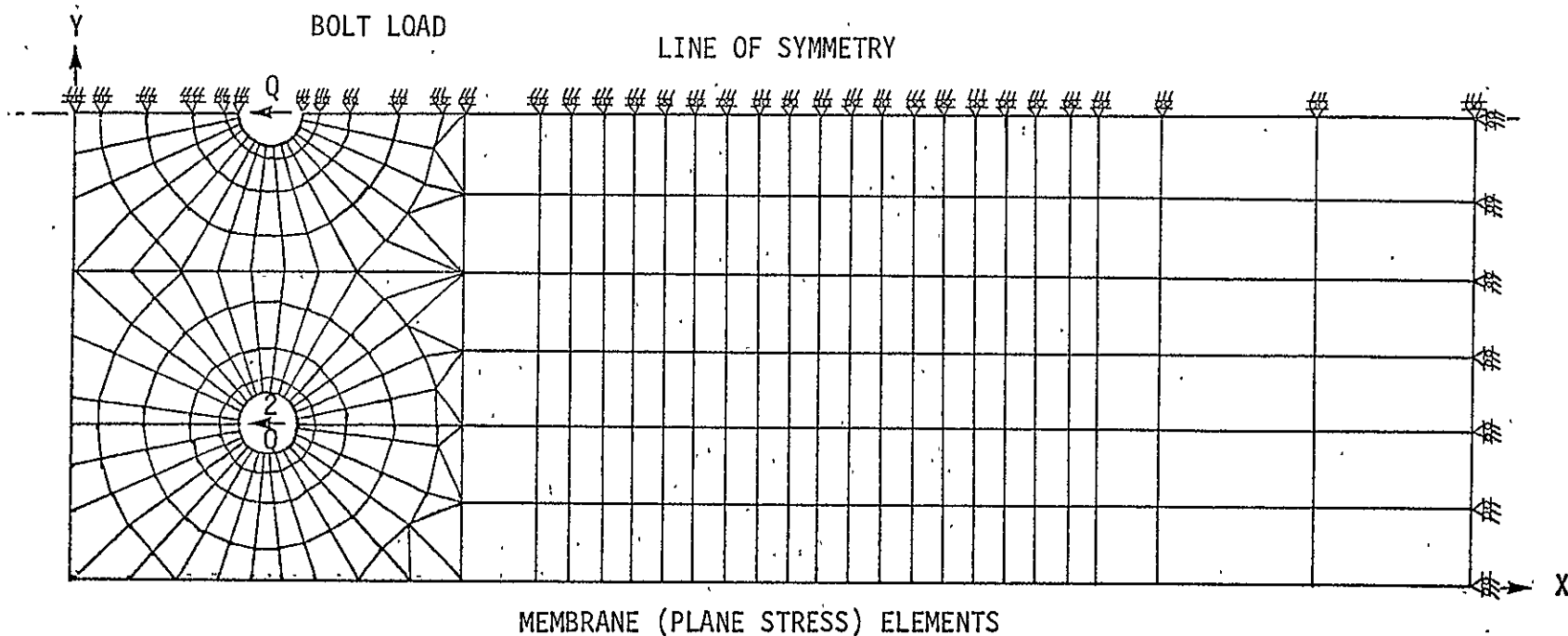


Figure 3. Finite Element Representation of Bolted Joint Specimen.

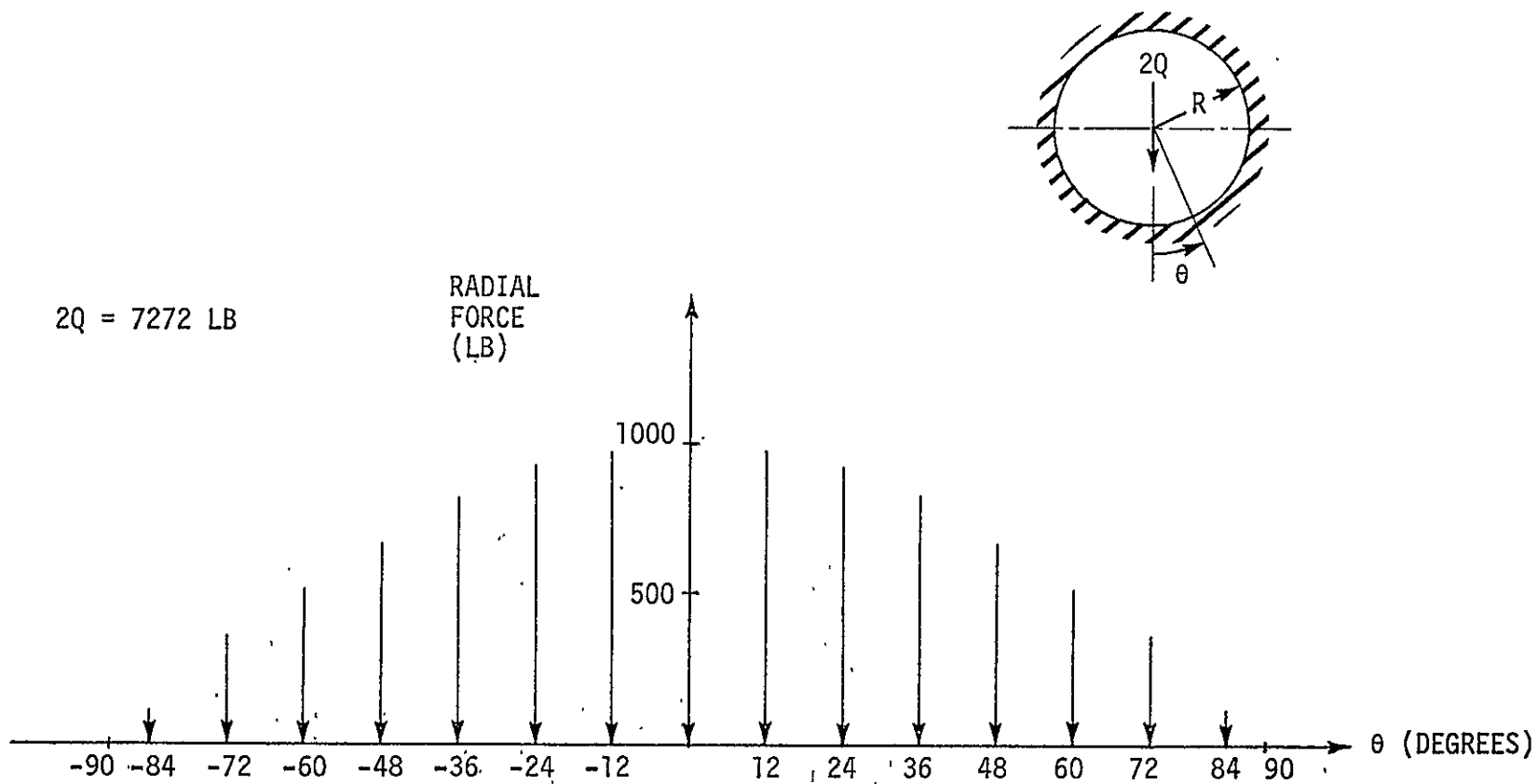


Figure 4. Bolt Transfer Loads for Cosine Load Distribution.

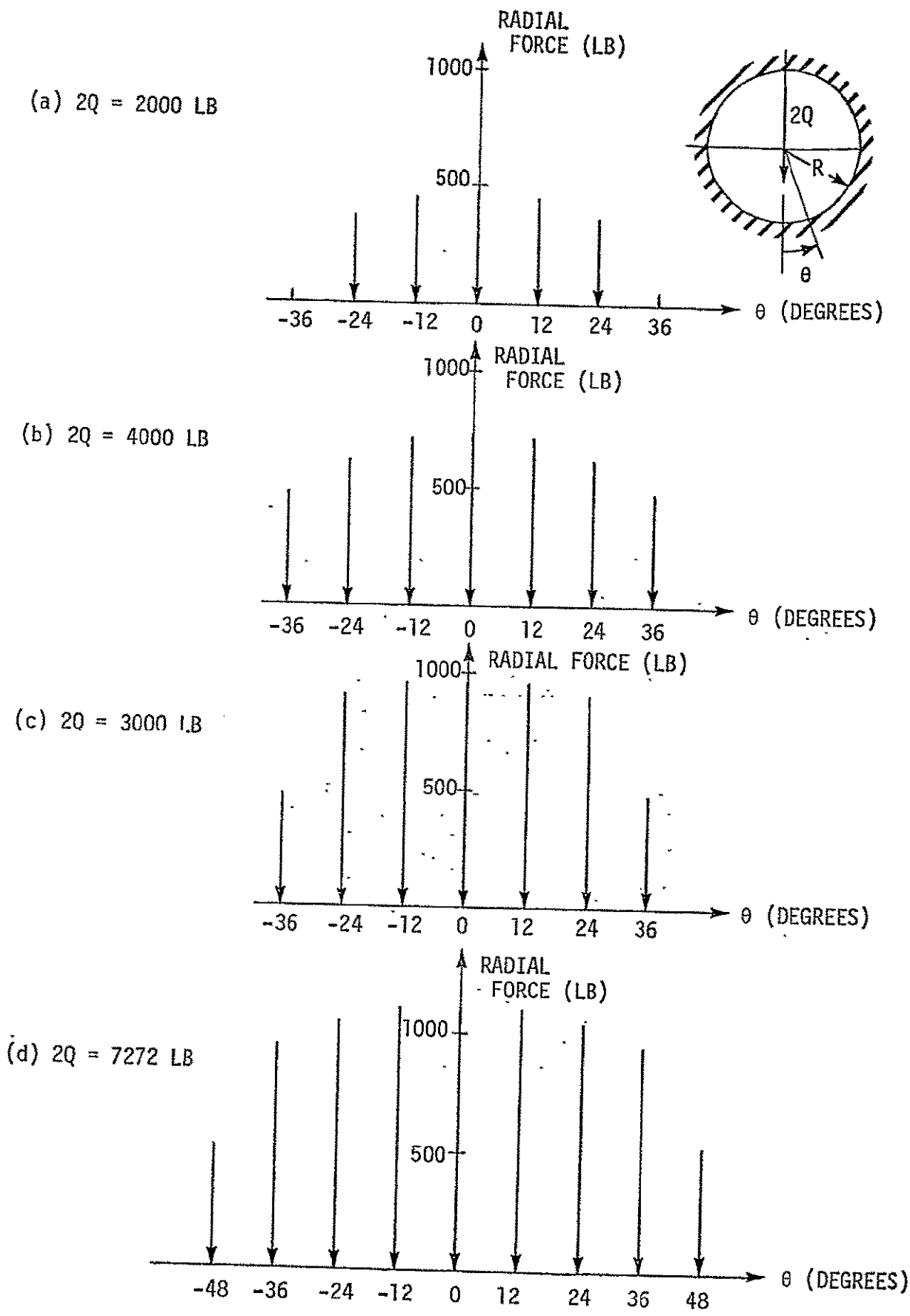


Figure 5. Bolt Transfer Loads from Nonlinear Loading Analysis.

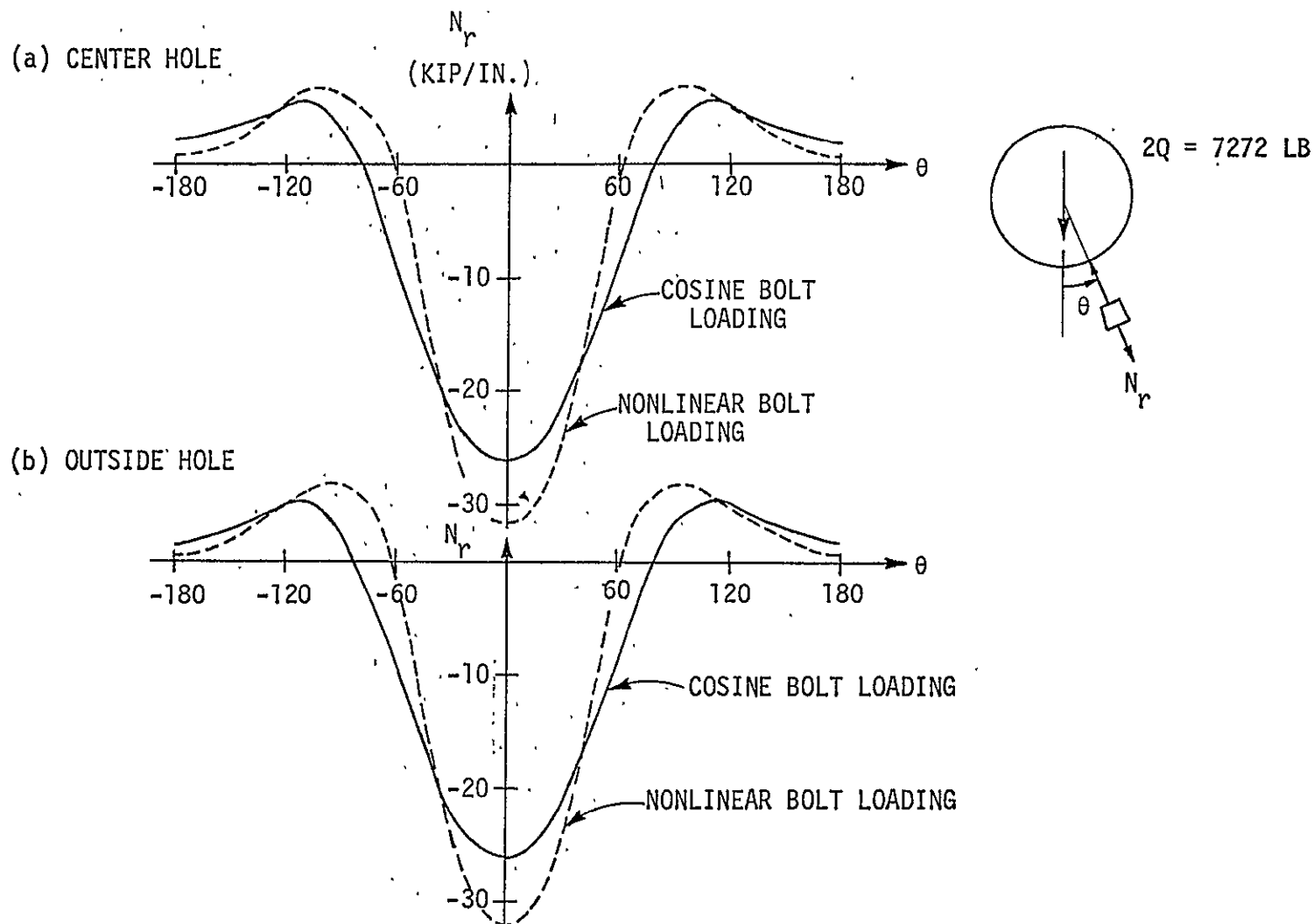


Figure 6. Radial Membrane Force Distributions at the Bolt Holes.

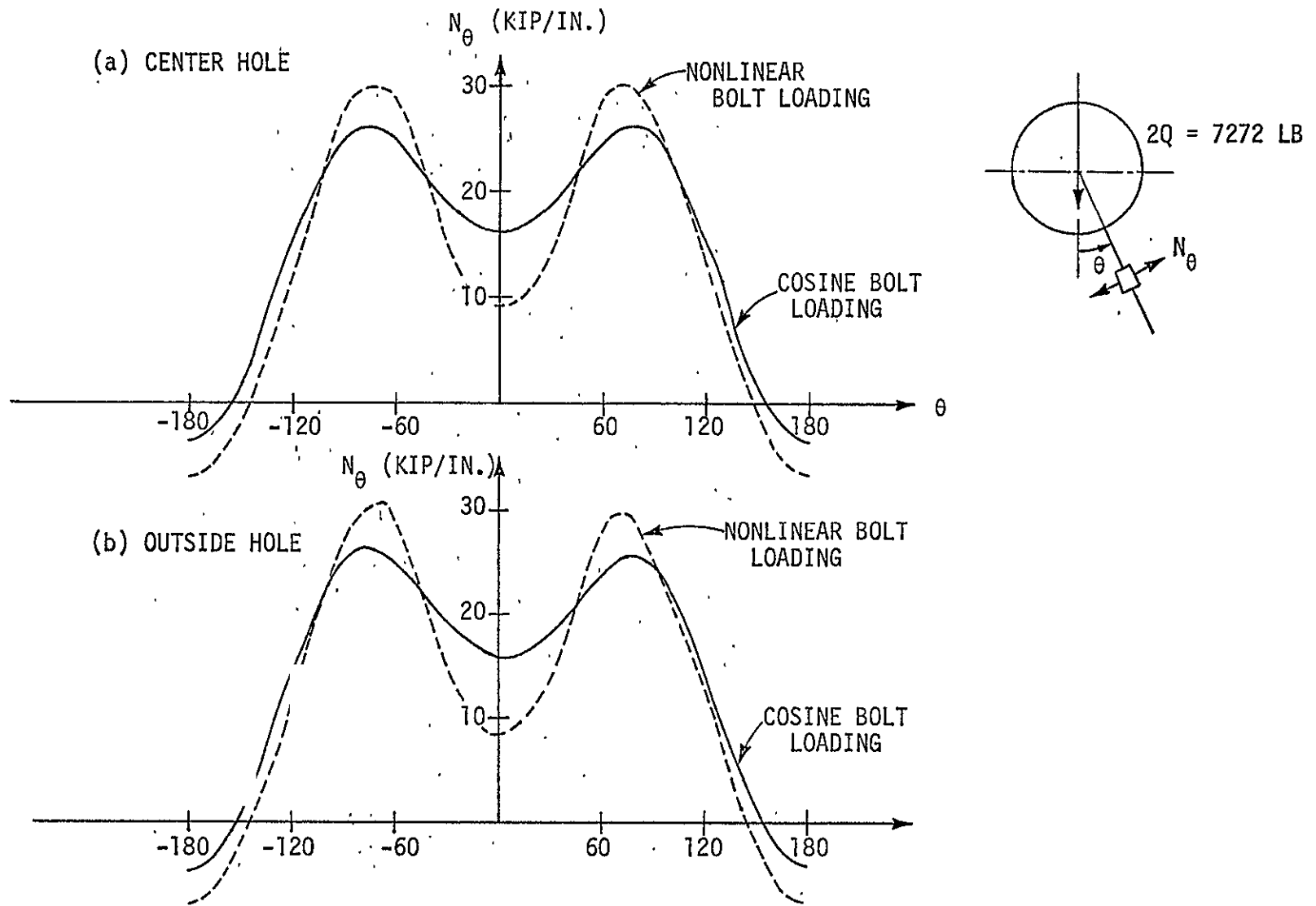


Figure 7. Circumferential Membrane Force Distributions at the Bolt Holes.

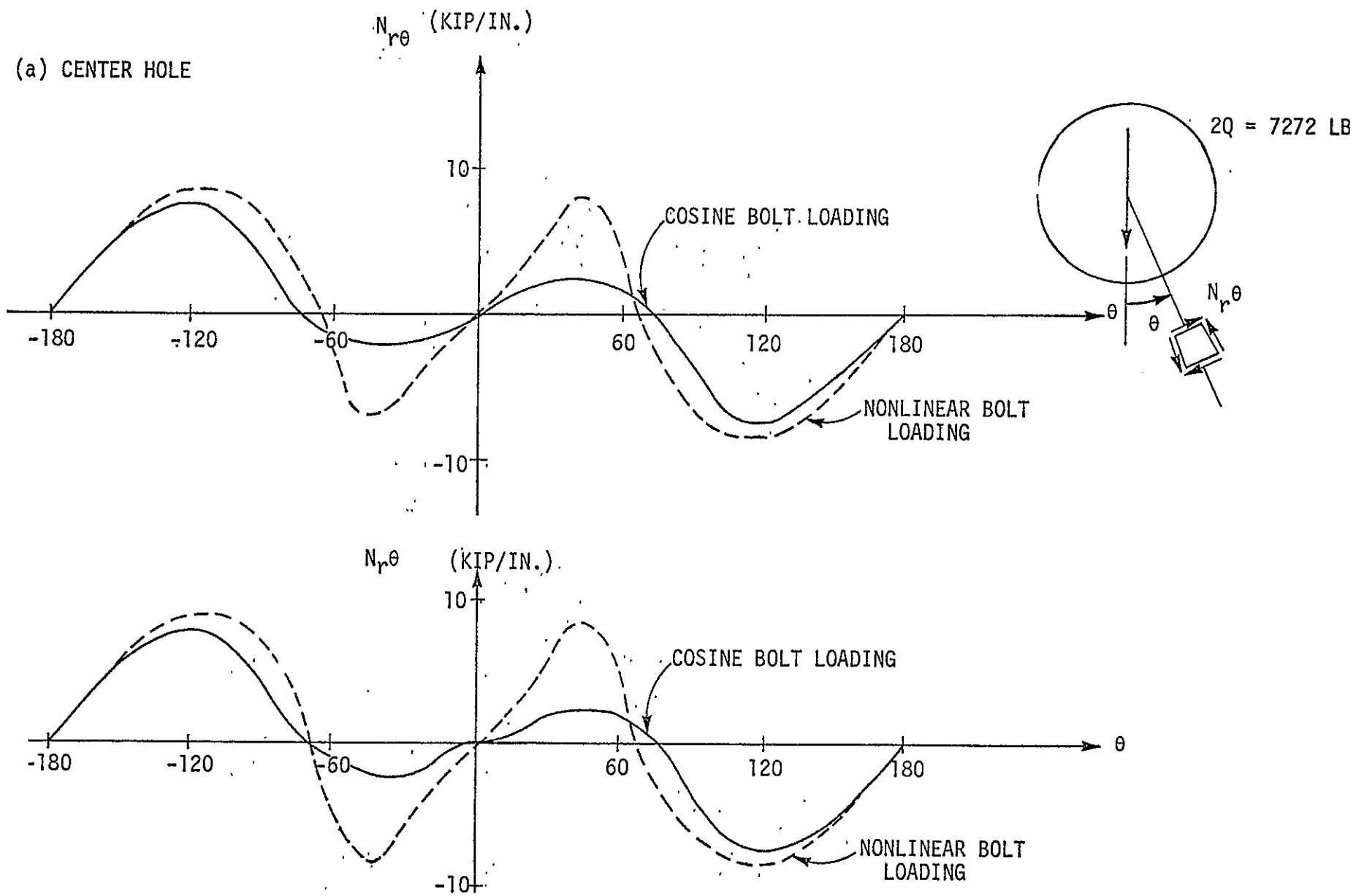


Figure 8. Membrane Shearing Force Distributions at the Bolt Hole.

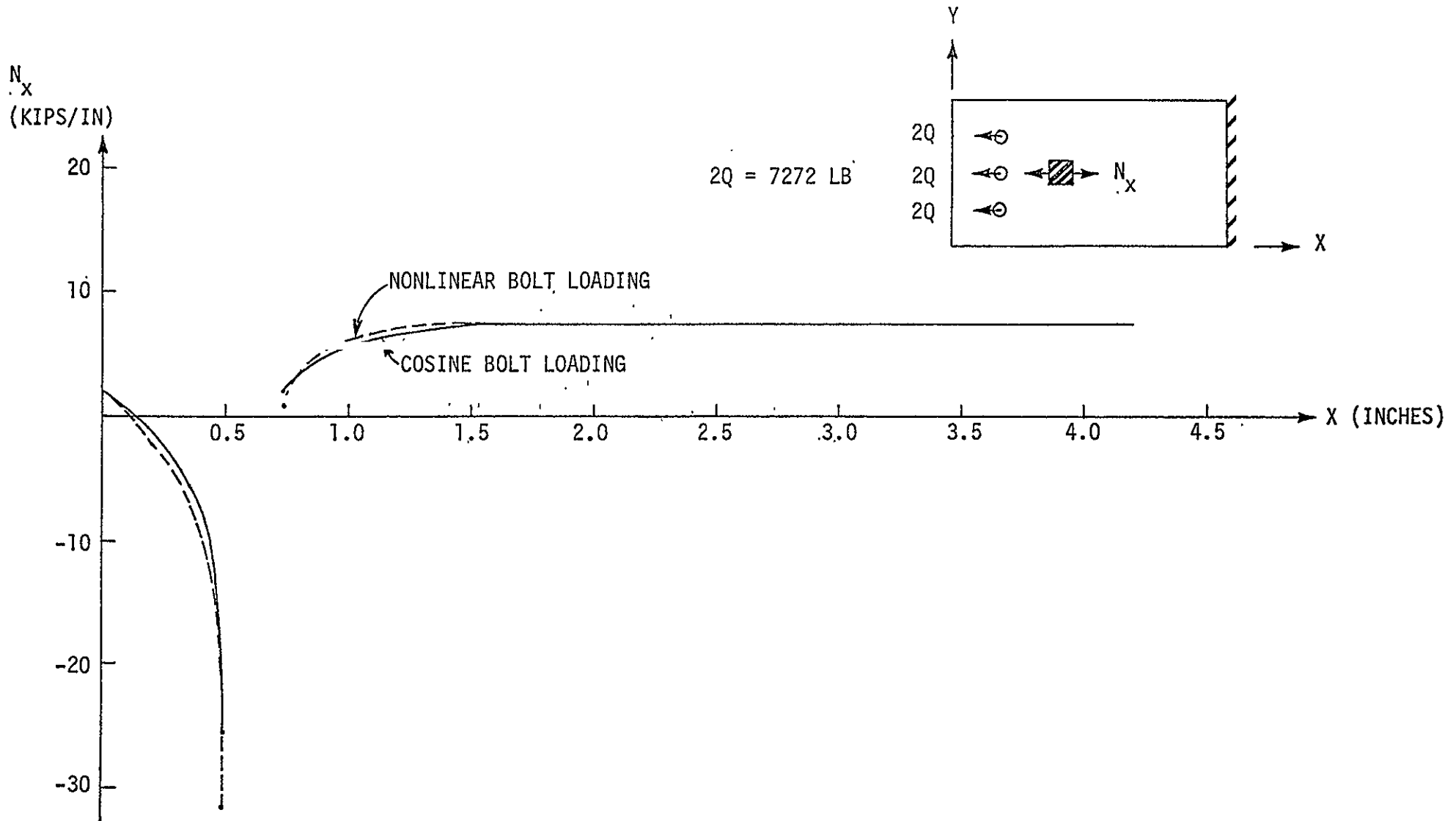
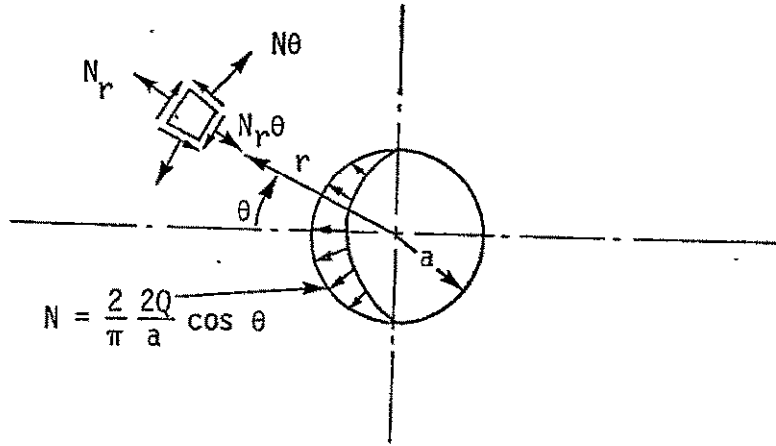
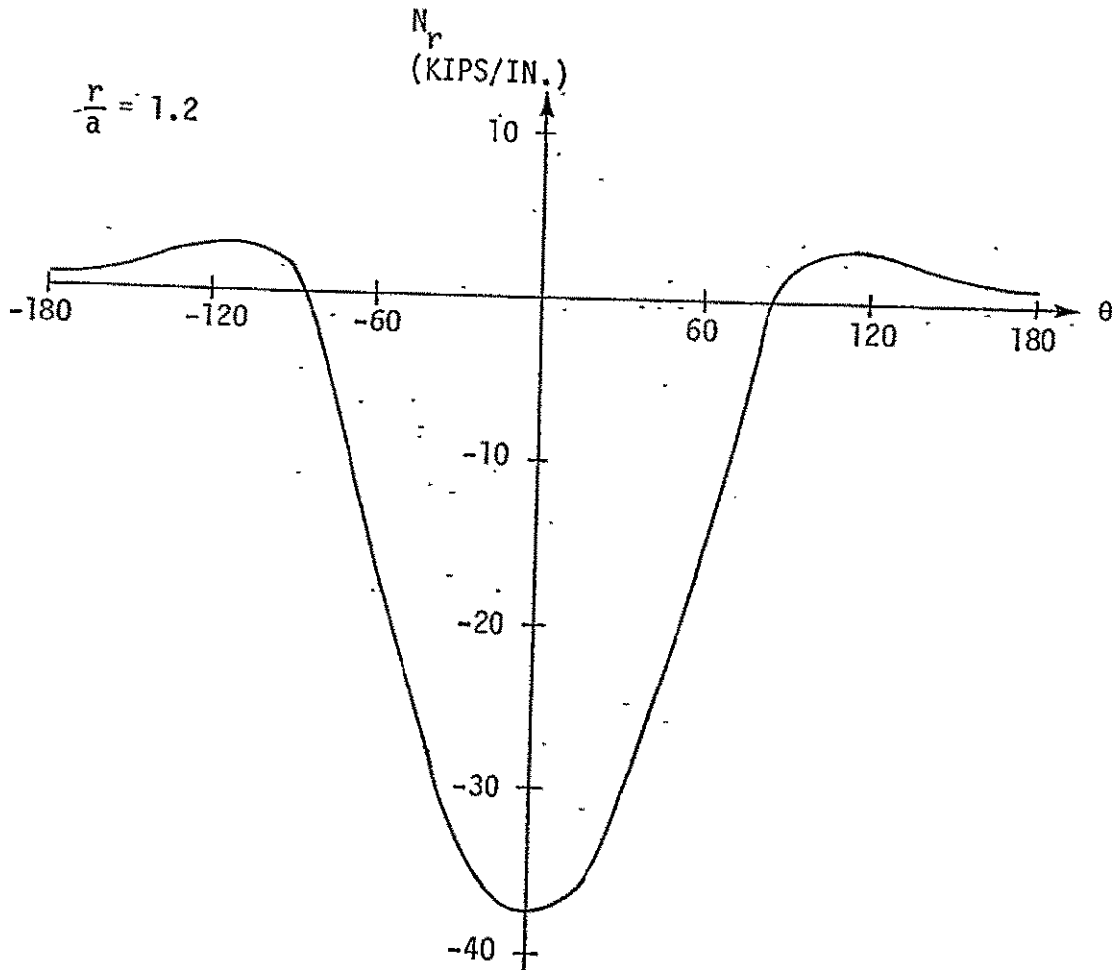


Figure 9. Longitudinal Membrane Force Distribution Along Specimen Centerline.

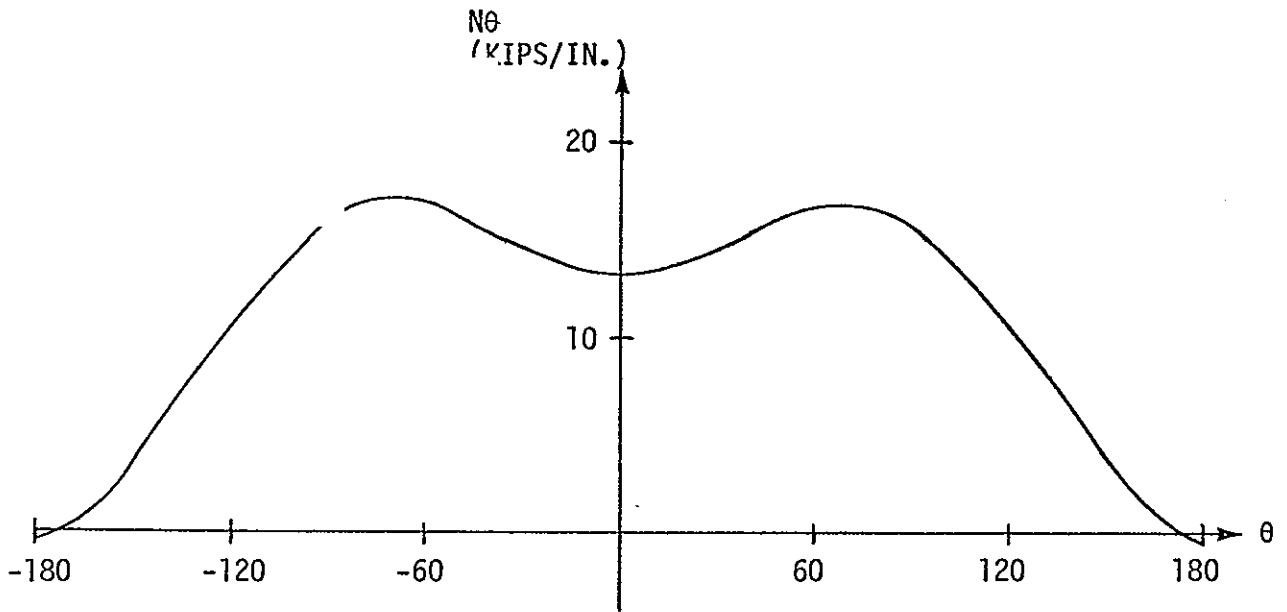


(a) INFINITE ISOTROPIC MEDIUM WITH COSINE BOLT LOADING

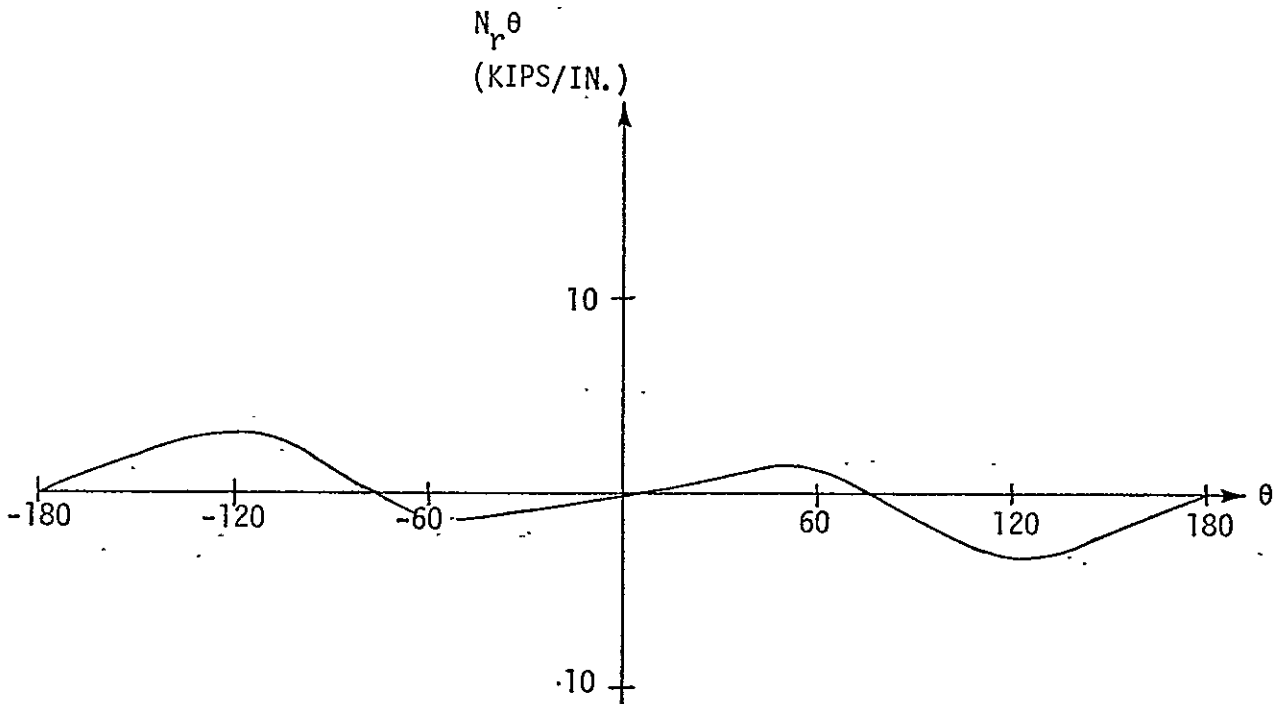


(b) RADIAL MEMBRANE FORCE DISTRIBUTION FOR $r/a = 1.2$.

Figure 10. Elasticity Solution for Infinite Isotropic Medium with Cosine Bolt Loading.



(c) CIRCUMFERENTIAL MEMBRANE FORCE DISTRIBUTION FOR $r/a = 1.2$.



(d) MEMBRANE SHEARING FORCE DISTRIBUTION FOR $r/a = 1.2$.

Figure 10 (concluded). Elasticity Solution for Infinite Isotropic Medium with Cosine Bolt Loading.

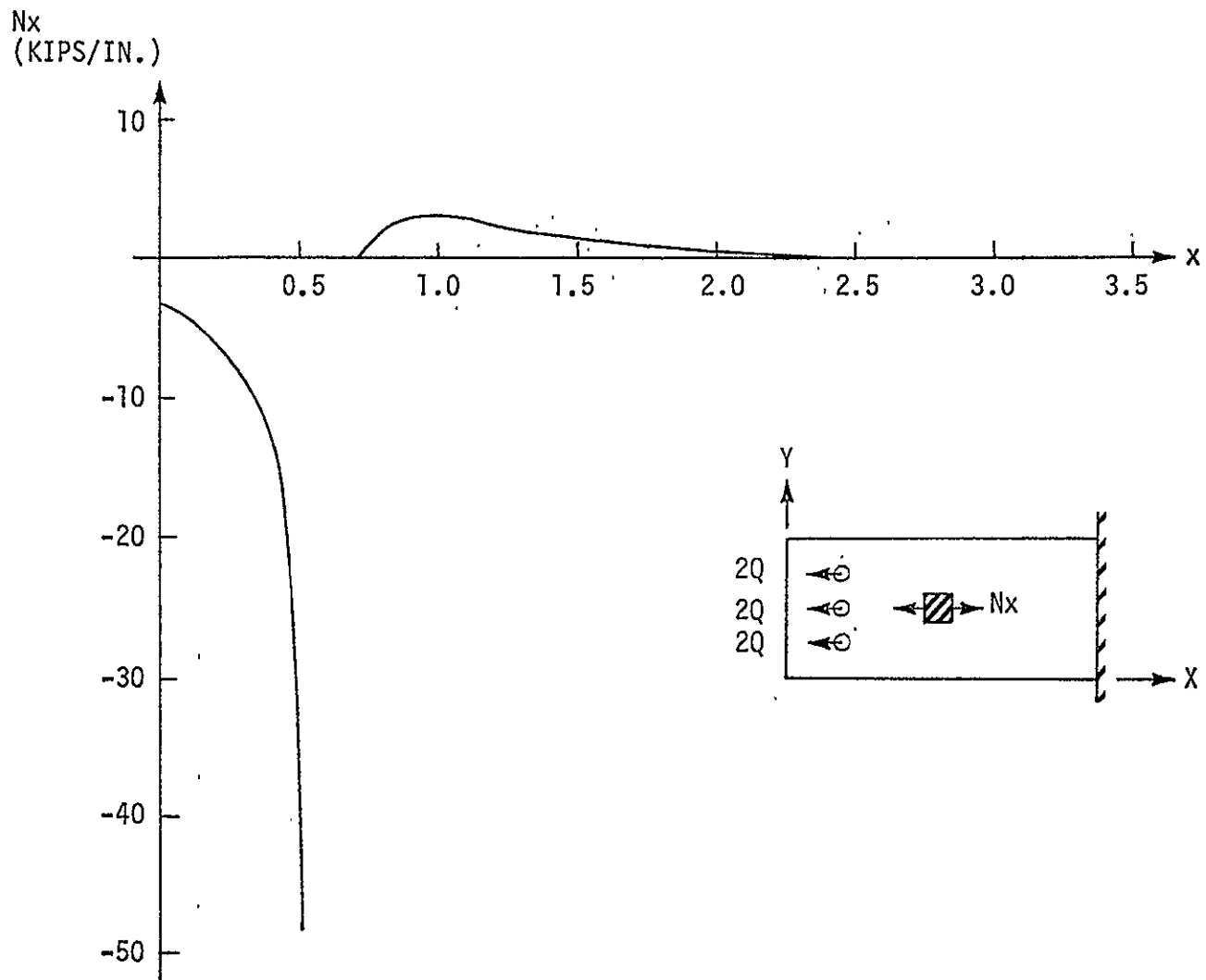


Figure 11. Elasticity Solution for Longitudinal Membrane Force Distribution Along Specimen Centerline.