DYNAMIC INTERACTIVE MODELLING
IN ELASTICITY: PROBLEMS

by

Boonrut Tantraphol

A Thesis Submitted to the Faculty of the
DEPARTMENT OF CIVIL ENGINEERING
AND ENGINEERING MECHANICS

In Partial Fulfillment of the Requirements
For the Degree of

MASTER OF SCIENCE
WITH A MAJOR IN CIVIL ENGINEERING

In the Graduate College

THE UNIVERSITY OF ARIZONA

1972
STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgement of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: B. Jantraphel

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

H. A. KAMEL
Professor of Aerospace and Mechanical Engineering

May 12, 1972
ACKNOWLEDGEMENT

The author wishes to express his sincere gratitude to Dr. H. A. Kamel for his valuable guidance, suggestions and technical assistance which greatly aided in the development and presentation of this thesis. The author is also indebted to Dr. Kamel for the use of the FELP2 program employed in this study.

The author is also grateful to his parents for their continuous encouragement and support.

Sincere appreciation is extended to the author's wife, Vatana, for her encouragement and assistance in the preparation of the manuscript.

In addition, the author appreciates the assistance rendered by Mr. Adisai Vongsevaisayavan in drafting some of the figures.

For the memory of her love, devotion and constant inspiration to her family, the author dedicates this thesis to his beloved late mother.
## TABLE OF CONTENTS

- **LIST OF ILLUSTRATIONS** ........................................ vi
- **ABSTRACT** ........................................................... viii

### CHAPTER

1. **INTRODUCTION** ................................................. 1

2. **THE DERIVATION OF THE LINEAR STRAIN TRIM6 ELEMENT** .... 9
   - Corner Displacements of the Triangle ................................ 9
   - Strain-displacement Matrix \( T_{e,p} \) ................................ 15
   - The Stiffness Matrix \( k \) ........................................ 19

3. **DEVELOPMENT OF THE TRIM5 AND TRIM4 ELEMENTS** ......... 23
   - Transformation from TRIM6 to TRIM5 ............................... 23
   - Relationship Between \( P_6 \) and \( P_5 \) .......................... 26
   - Relationship Between \( k_6 \) and \( k_5 \) .......................... 27
   - Displacements and Stresses ........................................ 28
   - The TRIM4 Element ................................................ 29
   - Relationship Between \( P_6 \) and \( P_4 \) .......................... 32
   - Relationship Between \( k_6 \) and \( k_4 \) .......................... 33
   - Displacements and Stresses ........................................ 33
   - Alternative Approaches ............................................ 34
     - Alternative #1 .................................................. 34
     - Alternative #2 .................................................. 37
     - Alternative #3 .................................................. 41

4. **EXAMPLE PROBLEM** ............................................. 47
   - Example 1 ......................................................... 47
TABLE OF CONTENTS--Continued

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Model #1</td>
<td>48</td>
</tr>
<tr>
<td>Model #2</td>
<td>48</td>
</tr>
<tr>
<td>Model #3</td>
<td>48</td>
</tr>
<tr>
<td>Comparison of results</td>
<td>48</td>
</tr>
<tr>
<td>Example 2</td>
<td>60</td>
</tr>
<tr>
<td>Model #1</td>
<td>60</td>
</tr>
<tr>
<td>Model #2</td>
<td>60</td>
</tr>
<tr>
<td>Model #3</td>
<td>61</td>
</tr>
<tr>
<td>Comparison of Results</td>
<td>61</td>
</tr>
<tr>
<td>5</td>
<td>73</td>
</tr>
<tr>
<td>CONCLUSIONS</td>
<td></td>
</tr>
</tbody>
</table>

APPENDIX

<table>
<thead>
<tr>
<th>APPENDIX</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>75</td>
</tr>
<tr>
<td>MATRIX FORMULATION OF ELEMENT PROPERTIES IN THE FINITE ELEMENT METHOD</td>
<td></td>
</tr>
<tr>
<td>Displacement Interpolation</td>
<td>76</td>
</tr>
<tr>
<td>Strains</td>
<td>77</td>
</tr>
<tr>
<td>Stresses</td>
<td>78</td>
</tr>
<tr>
<td>Equivalent Nodal Forces</td>
<td>79</td>
</tr>
<tr>
<td>B</td>
<td>84</td>
</tr>
<tr>
<td>THE TRIM3 ELEMENT</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>89</td>
</tr>
<tr>
<td>USER'S GUIDE</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>108</td>
</tr>
<tr>
<td>NOMENCLATURE</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>111</td>
</tr>
<tr>
<td>LISTING OF SUBROUTINES USED IN PROGRAM FELP2</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>140</td>
</tr>
<tr>
<td>INTRODUCTION OF INTERMEDIATE BOUNDARY NODES</td>
<td></td>
</tr>
<tr>
<td>REFERENCES</td>
<td>144</td>
</tr>
</tbody>
</table>
## LIST OF ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Some Membrane Elements: (a) Isoparametric Elements and Some Interfacing Elements, (b) and (c) Complete Polynomial and Interfacing Elements</td>
<td>5</td>
</tr>
<tr>
<td>2.1</td>
<td>A TRIM6 Element</td>
<td>10</td>
</tr>
<tr>
<td>2.2</td>
<td>Nodal Displacements of a TRIM6 Element</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>Triangle 1-2-3</td>
<td>21</td>
</tr>
<tr>
<td>3.1</td>
<td>A TRIM5 Element</td>
<td>24</td>
</tr>
<tr>
<td>3.2</td>
<td>Displacement of Node 1 by Unity</td>
<td>24</td>
</tr>
<tr>
<td>3.3</td>
<td>Displacement of Node 3 by Unity</td>
<td>24</td>
</tr>
<tr>
<td>3.4</td>
<td>Displacement of Node 2 by Unity</td>
<td>24</td>
</tr>
<tr>
<td>3.5</td>
<td>A TRIM4 Element</td>
<td>30</td>
</tr>
<tr>
<td>3.6</td>
<td>A TRIM6 Element Expanded from a TRIM4 Element</td>
<td>30</td>
</tr>
<tr>
<td>3.7</td>
<td>A TRIM6 Element Expanded from a TRIM5 Element</td>
<td>42</td>
</tr>
<tr>
<td>4.1</td>
<td>The Loaded Cantilever Plate for Example 1</td>
<td>50</td>
</tr>
<tr>
<td>4.2</td>
<td>Model #1 (Example 1)</td>
<td>51</td>
</tr>
<tr>
<td>4.3</td>
<td>Model #2 (Example 1)</td>
<td>52</td>
</tr>
<tr>
<td>4.4</td>
<td>Model #3 (Example 1)</td>
<td>53</td>
</tr>
<tr>
<td>4.5</td>
<td>$\sigma_x$ and $\sigma_y$ Stress Variation of a Cantilever Plate</td>
<td>54</td>
</tr>
<tr>
<td>4.6</td>
<td>$\tau_{xy}$ and $\sigma$ Stress Variation of a Cantilever Plate</td>
<td>55</td>
</tr>
<tr>
<td>4.7</td>
<td>Model #2 (Example 1) and its Deflected Shape</td>
<td>56</td>
</tr>
</tbody>
</table>
### LIST OF ILLUSTRATIONS--Continued

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.8</td>
<td>Stress Presentation of Model #1 (Example 1)</td>
<td>57</td>
</tr>
<tr>
<td>4.9</td>
<td>Stress Presentation of Model #2 (Example 1)</td>
<td>58</td>
</tr>
<tr>
<td>4.10</td>
<td>Stress Presentation of Model #3 (Example 1)</td>
<td>59</td>
</tr>
<tr>
<td>4.11</td>
<td>Square Plate with Hole</td>
<td>63</td>
</tr>
<tr>
<td>4.12</td>
<td>Model #1 (Example 2)</td>
<td>64</td>
</tr>
<tr>
<td>4.13</td>
<td>Model #2 (Example 2)</td>
<td>65</td>
</tr>
<tr>
<td>4.14</td>
<td>Model #3 (Example 2)</td>
<td>66</td>
</tr>
<tr>
<td>4.15</td>
<td>$\sigma_y$ Stress Distribution</td>
<td>67</td>
</tr>
<tr>
<td>4.16</td>
<td>$\sigma$ Stress Distribution</td>
<td>68</td>
</tr>
<tr>
<td>4.17</td>
<td>$\sigma_x$ Stress Distribution</td>
<td>69</td>
</tr>
<tr>
<td>4.18</td>
<td>$\tau_{xy}$ Stress Distribution</td>
<td>70</td>
</tr>
<tr>
<td>4.19</td>
<td>Stress Presentation of Model #3 (Example 2)</td>
<td>71</td>
</tr>
<tr>
<td>4.20</td>
<td>An Enlarged Portion of Model #3</td>
<td>72</td>
</tr>
<tr>
<td>B.1</td>
<td>A TRIM3 Element</td>
<td>84</td>
</tr>
<tr>
<td>C.1</td>
<td>A Triangle with the Origin at the Centroid</td>
<td>97</td>
</tr>
<tr>
<td>C.2</td>
<td>A Cantilever Plate under Loading</td>
<td>105</td>
</tr>
<tr>
<td>C.3</td>
<td>Undeflected- and Deflected-Shape of the Cantilever Plate</td>
<td>106</td>
</tr>
<tr>
<td>C.4</td>
<td>(a) Stress Presentation of the Plate</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(b) Unscaled and Undeflected Shape of the Plate</td>
<td></td>
</tr>
<tr>
<td>F.1</td>
<td>Flow Chart for Automatic Node Renumbering and Element-type Modification</td>
<td>141</td>
</tr>
<tr>
<td></td>
<td>Subsequent to the Introduction of Intermediate Boundary Nodes</td>
<td></td>
</tr>
<tr>
<td>F.2</td>
<td>Introduction of Intermediate Boundary Nodes</td>
<td>143</td>
</tr>
</tbody>
</table>
ABSTRACT

Dynamic interactive modelling analysis is defined as the analysis in which a problem may be solved in successive steps in which the model is continually improved, and in which the results from one step are used as a basis for obtaining the solution to the next modified model. The usefulness of graphical display, especially when employed in conjunction with the analysis, is also considered.

A linearly varying strain triangular element with six nodes (TRIM6 element) is developed. Triangular elements with five and four nodes (TRIM5 and TRIM4 elements) are also developed to be used as interfacing elements between the linearly varying strain triangular element (TRIM6) and the three-node constant strain triangular element (TRIM3) to ensure inter-element compatibility. Different approaches to obtain the stiffness matrices of interfacing elements are discussed.

Example problems are solved employing all four elements described above. It is concluded that the technique adopted in the analysis is economical.
A Fortran program for solving two-dimensional problems, including plotting subroutines, is also presented along with a user's guide.
CHAPTER 1

INTRODUCTION

With the wide use in business, scientific and engineering applications, large digital computers have become increasingly more sophisticated. Their capacity in terms of core size and availability of backing storage has multiplied from year to year. At the same time, a small revolution has taken place in the design and development of mini- and medium-sized computers which combine speed and modest capital investment. Interactive graphics terminals which may be connected to either a large or a small computer have also been introduced to the market. The efficient performance and the low cost of the smaller size computers make them particularly suitable for use with interactive graphics terminals in engineering analysis. Interactive graphics systems have been used for some time now in the Structural Design Process (Batdorf, Kapur, and Sayer, 1968, and Prince, 1971). With more "intelligent" interactive program packages developed, the problem of man-machine communication is greatly reduced. Such systems will play an increasingly important role in engineering analysis (Kamel, Liu, and White, 1971).
This thesis does not in any way advocate the exclusion of large systems from engineering analysis. Large size computers may be used to produce an overall solution to a problem. The results of such an analysis will be mainly displacements which may be used subsequently by a smaller size computer in obtaining "finer" or more accurate local solutions. Thus, any local region of interest, i.e., areas of higher stress concentration, may be investigated on line in order to get a better insight of the detailed structural behavior.

Whether the results are produced by a large or a small computer, an interactive graphics system may be used to display the structural model, the deflected shape, as well as the state of stress. While analysis packages for large computers are designed for a "single shot" type of solution, software packages for a small computer may be written in such a way that the engineer can examine the plots and interrupt the program at any stage in order to introduce modifications. Intermediate calculations in the modification process can be automated to a great extent so that the engineer is allowed to concentrate on the results and the subsequent model modifications until the results are satisfactory. Utilizing a light pen, the modification of a model can often be accomplished at great ease.
When using the Finite Element Method to solve a structural problem, an engineer uses his judgment to create a model which will best represent the structure. The most economic model to use is that which offers sufficient detail in areas of rapid change, and as few unknowns as possible in areas of little variation of the behavior. This implies a pre-knowledge of the results, which is only possible for relatively simple and routine problems. In real life, it is often difficult to choose a model that best represents the structure and yet can be solved most economically. Therefore, several different models may have to be used. After a model is chosen and results are obtained, it is often the case that particular areas of the model need further analysis, leading to modifications of the model in the form of, for example, an increase in the number of elements and nodes and/or a change in the element type. The technique by which a problem may be solved in successive steps in which the model is continually improved, and in which the results from one step are used as a basis for obtaining the solution to the next modified model is termed "Dynamic Interactive Modelling Analysis."

In a Finite Element solution and, in particular, in an interactive analysis, different areas of a model often require different degrees of refinement. There are several element types providing various degrees of
sophistication which may be used for the purpose. For example, one may consider isoparametric elements, higher order complete polynomial and restricted polynomial elements, some of which are shown in Figures 1.1a and 1.1b. It is the primary aim of this thesis to provide the tools necessary to enable mixing of elements of different order within the same analysis without violating the inter-element compatibility. To be more specific, the thesis is developed to allow for the mixing of TRIM3 and TRIM6 elements with the use of interface elements such as the TRIM₅ and TRIM₄ elements (triangular membrane elements¹ with 5 and 4 nodes, respectively - see Figure 1.1c) with the stress on interactive analysis. The concept of using TRIM₅ and TRIM₄ elements as interfaces between TRIM3 and TRIM6 elements was investigated in principle by David Yeung as an assignment in a graduate class conducted by Prof. H. A. Kamel in 1970. The Program HOE (Higher Order Elements) developed by Donald Liu, another graduate student of Prof. H. A. Kamel, to compute the stiffness matrix (\(k\)) of a TRIM₆ element was modified in 1971 to compute a TRIM₅ element, and equilibrium checks were satisfactorily performed on the stiffness matrix obtained. In a recent

¹ Since the elements mentioned throughout this paper are membrane elements, the terms "element" and "membrane element" are used interchangeably.
Fig. 1.1 Some Membrane Elements: (a) Isoparametric Elements and Some Interfacing Elements, (b) and (c) Complete Polynomial and Interfacing Elements
book by John L. Meek (1971) another approach (see Chapter 3) for obtaining the stiffness matrices \( k \) of TRIM5 and TRIM4 elements is shown. However, none of these works has combined such elements (TRIM6, TRIM5, TRIM4 and TRIM3) in solving a problem.

As an example of how an interactive analysis may be performed in a plane stress problem, TRIM3 (a triangular membrane element having 3 nodes - see Figure 1.1b) may be used to get the overall picture of the stress distribution within the structure. It is well known that, whereas, this simple element requires less computation time, it is less accurate than the more sophisticated elements (except in the case of a constant- or nearly constant-stress field for which it yields the exact- or a sufficiently accurate-solution). After studying the results, it is often desirable to refine the solution, say, by changing some of the elements from TRIM3 to other higher order types such as TRIM6 or TRIM10 (membrane elements with 6 and 10 nodes, respectively - see Figure 1.1b). The change may be accomplished locally in the area of interest, with interfacing elements used between the local area and the rest of the model.

The thesis also includes a certain amount of experimentation in the area of graphical display. A computer program (see Appendix D) for plotting the model,
the deflected shape, either separately or superimposed on top of the original model - as well as a state of stress parameter based on the von Mises Criterion. The use of the von Mises Criterion (as suggested by Key and Beisinger, 1971) to represent the state of stress at a node or within an element gives an indication of the factor of safety at the point. An alternative method for displaying the state of stress is to plot vectors for $\sigma_x$, $\sigma_y$, and $\tau_{xy}$ separately for each node. The plotting subprogram has been written in a generalized manner so that it can handle all previously-described elements (TRIM3, TRIM4, TRIM5, and TRIM6) as well as a parallelogram membrane element (PARAM) which is also available in the FELP2\textsuperscript{2} program.

Another aspect of interest is the local analysis capability of FELP2. It is possible to recompute a certain region using a finer mesh subsequent to an analysis. The plotting subprogram is capable of automatically enlarging the local area for more accurate display.

In changing element types during the process of a dynamic modelling analysis, the program can be improved so that the addition of a new node can be accomplished by a reference to the element interface to be modified. The

2. FELP2 is a Finite Element Program for solving two-dimensional problems developed by Professor H. A. Kamel at The University of Arizona's Aerospace and Mechanical Engineering Department.
program will then automatically assign the new node an internal number, and change the types of the elements affected. In assigning the new node numbers, a process of internal renumbering of all nodes may be automatically performed to optimize the band width (see Appendix F).

The program for this thesis was developed for the CDC-6400 computer at the University Computer Center, although the final implementation of the procedure is envisaged for the AME Department's medium-sized PDP-15 computer.

The two computers have two important features in common. They both accept the FORTRAN IV language and have plotting capabilities that can be programmed in a similar manner. The program can be adapted to the PDP-15 without much difficulty. The PDP-15's interactive graphics console, however, is necessary so that the objective of this research is totally achieved.
CHAPTER 2

THE DERIVATION OF THE LINEAR STRAIN TRIM6 ELEMENT

TRIM6 is a triangular membrane element with six nodal points, three of which are located on the three corners while the other three are located along the three sides (see Figure 2.1).

Corner Displacements of the Triangle

The displacements of the six corner nodes are defined by the column matrix.

\[
P_{(12x1)} = \begin{bmatrix}
  u_1 \\
  v_1 \\
  u_2 \\
  v_2 \\
  u_3 \\
  v_3 \\
  u_4 \\
  v_4 \\
  u_5 \\
  v_5 \\
  u_6 \\
  v_6 
\end{bmatrix} \tag{2-1}
\]
Fig. 2.1 A TRIM6 Element

(a) Displacement of a Corner Node
(b) Displacement of a Side Node

Fig. 2.2 Nodal Displacements of a TRIM6 Element
The displacement variation within the triangle is governed by a complete second order polynomial of $x$ and $y$. Figure 2.2 shows two types of displacement shapes, one associated with a corner node and the other with a side node.

Any one of the displacement functions for the element may be written as:

$$f = a_0 + a_x x + a_y y + a_{xx} x^2 + a_{xy} x y + a_{yy} y^2$$  \hspace{0.5cm} (2-2)

The six unknown coefficients $a_0$ to $a_{yy}$ may be determined from the boundary conditions at the six nodes.

If a node, for example, node 1, is moved by unity in a certain direction, say the $x$ direction, while the other 5 nodes remain stationary, a set of six equations may be used to determine the unknown coefficients:

$$a_{01} + a_{xx} x_1 + a_{yy} y_1 + a_{xx} x_1^2 + a_{xy} x_1 y_1 + a_{yy} y_1^2 = 1$$
$$a_{01} + a_{xx} x_2 + a_{yy} y_2 + a_{xx} x_2^2 + a_{xy} x_2 y_2 + a_{yy} y_2^2 = 0$$
$$a_{01} + a_{xx} x_3 + a_{yy} y_3 + a_{xx} x_3^2 + a_{xy} x_3 y_3 + a_{yy} y_3^2 = 0$$
$$a_{01} + a_{xx} x_4 + a_{yy} y_4 + a_{xx} x_4^2 + a_{xy} x_4 y_4 + a_{yy} y_4^2 = 0$$
$$a_{01} + a_{xx} x_5 + a_{yy} y_5 + a_{xx} x_5^2 + a_{xy} x_5 y_5 + a_{yy} y_5^2 = 0$$
$$a_{01} + a_{xx} x_6 + a_{yy} y_6 + a_{xx} x_6^2 + a_{xy} x_6 y_6 + a_{yy} y_6^2 = 0$$  \hspace{0.5cm} (2-3)
A similar set of equations is associated with the movement of each of the six nodes of the TRIM6 element in either the x or y direction. The six equation sets for either x or y may be written jointly as:

\[
\begin{bmatrix}
1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 \\
1 & x_2 & y_2 & x_2^2 & x_2 y_2 & y_2^2 \\
1 & x_3 & y_3 & x_3^2 & x_3 y_3 & y_3^2 \\
1 & x_4 & y_4 & x_4^2 & x_4 y_4 & y_4^2 \\
1 & x_5 & y_5 & x_5^2 & x_5 y_5 & y_5^2 \\
1 & x_6 & y_6 & x_6^2 & x_6 y_6 & y_6^2 \\
\end{bmatrix}
\begin{bmatrix}
a_{01} & a_{02} & a_{03} & a_{04} & a_{05} & a_{06} \\
a_{x1} & a_{x2} & a_{x3} & a_{x4} & a_{x5} & a_{x6} \\
a_{y1} & a_{y2} & a_{y3} & a_{y4} & a_{y5} & a_{y6} \\
a_{xx1} & a_{xx2} & a_{xx3} & a_{xx4} & a_{xx5} & a_{xx6} \\
a_{xy1} & a_{xy2} & a_{xy3} & a_{xy4} & a_{xy5} & a_{xy6} \\
a_{yy1} & a_{yy2} & a_{yy3} & a_{yy4} & a_{yy5} & a_{yy6} \\
\end{bmatrix} = I^{(6x6)}
\]

or

\[X \cdot A = I.\]

The \(X\) matrix in Equation (2-4a) is readily computed from the x and y coordinates of the six nodal points of the element. With the right-hand side of the equation being an identity matrix \(I\), the matrix \(A\) is simply the inverse of the matrix \(X\)

\[A = X^{-1}.\]
The expression for the matrix $X^{-1}$ has been obtained in closed form (Veubeke, 1965, and Argyris, 1965) in the simple case of mid-side nodes. In the more general case, $A$ may be found numerically. The matrix $A$ can be used for the interpolation of the $u$- and $v$-displacements, or indeed in any other interpolation related to the element such as temperature or body force distribution.

One may express the matrix $A$ as

$$A = \begin{bmatrix}
   a_{01} & a_{02} & a_{03} & a_{04} & a_{05} & a_{06} \\
   a_{x1} & a_{x2} & a_{x3} & a_{x4} & a_{x5} & a_{x6} \\
   a_{y1} & a_{y2} & a_{y3} & a_{y4} & a_{y5} & a_{y6} \\
   a_{xx1} & a_{xx2} & a_{xx3} & a_{xx4} & a_{xx5} & a_{xx6} \\
   a_{xy1} & a_{xy2} & a_{xy3} & a_{xy4} & a_{xy5} & a_{xy6} \\
   a_{yy1} & a_{yy2} & a_{yy3} & a_{yy4} & a_{yy5} & a_{yy6}
\end{bmatrix}.$$ 

One can readily see that the first row of the matrix gives the constant coefficients, the second row gives the coefficients of $x$, the third row gives the coefficients of $y$, etc.

The transformation matrix for the displacements [see Equation (A-1)] can be written as:
\[ T_{u,p} = \left[ A_0 + A_x + A_y + A_{xx} + A_{xy} + A_{yy} \right] \]

(2-6)

The matrices \( A_0, A_x, A_y, \ldots, A_{yy} \) are constructed by rearranging elements of the matrix \( A \). For example:

\[ A_0 = [A_{01} A_{02} A_{03} A_{04} A_{05} A_{06}] \]

\[ (2\times12) (2\times2) (2\times2) (2\times2) (2\times2) (2\times2) (2\times2) \]

where

\[ A_{01} = \begin{bmatrix} a_{01} & 0 \\ 0 & a_{01} \end{bmatrix} \]

\[ (2\times2) \]

\[ A_x = [A_{x1} A_{x2} A_{x3} A_{x4} A_{x5} A_{x6}] \]

\[ (2\times12) (2\times2) (2\times2) (2\times2) (2\times2) (2\times2) (2\times2) \]

where

\[ A_{x1} = \begin{bmatrix} a_{x1} & 0 \\ 0 & a_{x1} \end{bmatrix} \]

\[ (2\times2) \]

\[ A_{yy} = [A_{yy1} A_{yy2} A_{yy3} A_{yy4} A_{yy5} A_{yy6}] \]

\[ (2\times12) (2\times2) (2\times2) (2\times2) (2\times2) (2\times2) (2\times2) \]
where

\[
\frac{A_{yyi}}{(2x2)} = \begin{bmatrix}
a_{yyi} & 0 \\
0 & a_{yyi}
\end{bmatrix} \quad (2-7)
\]

Strain-displacement Matrix \( (T_{e,p}) \)

Re-writing Equation \((A-8)^1\)

\[
T_{e,p} = D_{e,u} \mathbf{T}_{u,p} \quad (A-8)
\]

and from Equation \((A-7)\)

\[
D_{e,u} = \begin{bmatrix}
\frac{\partial}{\partial x} & 0 \\
0 & \frac{\partial}{\partial y} \\
\frac{\partial}{\partial y} & \frac{\partial}{\partial x}
\end{bmatrix} \quad (A-7)
\]

\[
= \begin{bmatrix}
\frac{\partial}{\partial x} & 0 \\
0 & 0 \\
0 & \frac{\partial}{\partial x}
\end{bmatrix} + \begin{bmatrix}
0 & 0 \\
0 & \frac{\partial}{\partial y} \\
\frac{\partial}{\partial y} & 0
\end{bmatrix} \quad (2-8)
\]

\[
= \begin{bmatrix}
1 & 0 \\
0 & 0 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
0 & 0 \\
\frac{\partial}{\partial x} & 0 \cdot 1 \\
0 \cdot 1 & \frac{\partial}{\partial y}
\end{bmatrix} \quad (2-9)
\]

\[
= D_x \cdot \frac{\partial}{\partial x} + D_y \cdot \frac{\partial}{\partial y} \quad (2-10)
\]

1. The A in the equation refers to Appendix A.
where \( \mathbf{D}_x = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \) and \( \mathbf{D}_y = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \) (2-11)

It is noted that both \( \mathbf{D}_x \) and \( \mathbf{D}_y \) are constant matrices.

Substituting Equations (2-10) and (2-6) into Equation (A-8)

\[
T_{e,p} = (\mathbf{D}_x \partial / \partial x + \mathbf{D}_y \partial / \partial y)(\mathbf{A}_0 + \mathbf{A}_x x + \mathbf{A}_y y + \mathbf{A}_{xx} x^2 + \mathbf{A}_{xy} xy + \mathbf{A}_{yy} y^2)
\]

\[
= \mathbf{D}_x (\mathbf{A}_x + 2\mathbf{A}_{xx} x + \mathbf{A}_{xy} y) + \mathbf{D}_y (\mathbf{A}_y + 2\mathbf{A}_{yy} y + \mathbf{A}_{xy} x)
\]

\[
T_{e,p} = (\mathbf{D}_x \mathbf{A}_x + \mathbf{D}_y \mathbf{A}_y) + (2 \mathbf{D}_x \mathbf{A}_{xx} + \mathbf{D}_y \mathbf{A}_{xy}) + (2 \mathbf{D}_y \mathbf{A}_{yy} + \mathbf{D}_x \mathbf{A}_{xy}) y
\]

\[
= \mathbf{B}_0 + \mathbf{B}_x x + \mathbf{B}_y y
\] (2-12)

\[
\mathbf{B}_0 = (\mathbf{D}_x \mathbf{A}_x + \mathbf{D}_y \mathbf{A}_y)
\] (3x12) (3x2)(2x12) (3x2)(2x12) (2-13)
\[
\begin{bmatrix}
1 & 0 \\
0 & 1 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
\begin{bmatrix}
ax_1 & 0 & ax_2 & 0 & ax_3 & 0 & ax_4 & 0 & ax_5 & 0 & ax_6 & 0 \\
0 & 0 & ax_1 & 0 & ax_2 & 0 & ax_3 & 0 & ax_4 & 0 & ax_5 & 0 & ax_6
\end{bmatrix}
+ \\
0 & 0 & 0 & 0 & ay_1 & 0 & ay_2 & 0 & ay_3 & 0 & ay_4 & 0 & ay_5 & 0 & ay_6
\end{bmatrix}
\begin{bmatrix}
ay_1 & ax_1 & ay_2 & ax_2 & ay_3 & ax_3 & ay_4 & ax_4 & ay_5 & ax_5 & ay_6 & ax_6 \\
ax_1 & 0 & ax_2 & 0 & ax_3 & 0 & ax_4 & 0 & ax_5 & 0 & ax_6 & 0
\end{bmatrix}
= [B_{01} & B_{02} & B_{03} & B_{04} & B_{05} & B_{06}] \tag{2-14}
\]

where

\[
B_{01} = \begin{bmatrix} ax_1 & 0 \\ 0 & ay_1 \\ ay_1 & ax_1 \end{bmatrix} \quad \text{(3x2)}
\]
Similarly

\[
\frac{B_x}{(3\times12)} = (2D_{xx}A_{xx} + D_{xy}A_{xy}) \tag{2-15}
\]

\[
= \begin{bmatrix}
2a_{xx1} & 0 & 2a_{xx2} & \cdots & 2a_{xx6} & 0 \\
0 & a_{xy1} & 0 & \cdots & 0 & a_{xy6} \\
a_{xy1} & 2a_{xx1} & a_{xy2} & \cdots & a_{xy6} & 2a_{xx6}
\end{bmatrix} \tag{2-16}
\]

\[
= [B_{x1} \ B_{x2} \ B_{x3} \ B_{x4} \ B_{x5} \ B_{x6}] \tag{2-17}
\]

where

\[
\frac{B_{x1}}{(3\times2)} = \begin{bmatrix}
2a_{xx1} & 0 \\
0 & a_{xy1} \\
a_{xy1} & 2a_{xx1}
\end{bmatrix}
\]

Again, similarly

\[
\frac{B_y}{(3\times12)} = (2D_{yy}A_{yy} + D_{xy}A_{xy}) \tag{2-18}
\]

\[
= \begin{bmatrix}
2a_{yy1} & 0 & 2a_{yy2} & 0 & \cdots & 2a_{yy6} & 0 \\
0 & a_{xy1} & 0 & a_{xy2} & \cdots & 0 & a_{xy6} \\
a_{xy1} & 2a_{yy1} & a_{xy2} & 2a_{yy2} & \cdots & a_{xy6} & 2a_{yy6}
\end{bmatrix} \tag{2-19}
\]

\[
= [B_{y1} \ B_{y2} \ B_{y3} \ B_{y4} \ B_{y5} \ B_{y6}] \tag{2-19}
\]
where

\[
\begin{bmatrix}
2a_{yyi} & 0 \\
0 & a_{xxi} \\
a_{xxi} & 2a_{yyi}
\end{bmatrix}
\]  

The Stiffness Matrix \( k \)

Rewriting Equation (A-31)

\[
k = \int_T T_{e,p}^t E T_{e,p} dV.
\]  \( (2-20) \)

For an element having a constant thickness \( t \),

\[
k = t \int_A T_{e,p}^t \bar{\rho} E T_{e,p} \bar{\rho} dA
\]  \( (2-21) \)

where

\( E \) is the material stiffness matrix or the Generalized Young's Modulus,

\( t \) is the thickness of the element, and

\( A \) is the area of the triangle.
Substituting the matrix $T_{e,p}$ from Equation (2-12) into Equation (2-21)

\[
k = t \int_A (B_0^t + B_x^t x + B_y^t y)(E B_0 + E B_x x + E B_y y) \, dx \, dy
\]

\[
= t \int_A (B_0^t E B_0 + B_x^t E B_x x \int_A x \, dx + B_y^t E B_y y \int_A x^2 \, dx + B_x^t E B_x x \int_A y \, dy + B_y^t E B_y y \int_A y^2 \, dy
\]

\[
+ B_0^t E B_x \int_A x \, dx \, dy + B_x^t E B_x \int_A x \, dx \, dy
\]

\[
+ B_0^t E B_y \int_A y \, dy \, dy + B_y^t E B_y \int_A y \, dy \, dy
\]

\[
+ B_x^t E B_y \int_A x y \, dx \, dy + B_y^t E B_x \int_A x y \, dy \, dy
\]

\[
= t \int_A (B_0^t E B_0 + B_x^t E B_x + B_y^t E B_y) \, dx \, dy
\]

or

\[
k = t[B_0^t E B_0 + B_x^t E B_x + B_y^t E B_y]
\]

\[
+ (B_0^t E B_x + B_x^t E B_0) I_y
\]

\[
+ (B_0^t E B_y + B_y^t E B_0) I_x
\]

\[
+ (B_x^t E B_y + B_y^t E B_x) I_{xy}
\]

\[
(2-23)
\]
For a triangle, with \((x_1, y_1), (x_2, y_2)\) and \((x_3, y_3)\) being the coordinates of the three corners (1, 2 and 3) with respect to the centroid of the triangle (see Fig. 2.3), the values of the integrals are given by

\[
A = \iiint_A dx
dy = \text{area of the triangle}
\]

\[
I_x = \iiint_A y
dx
dy = 0
\]

\[
I_y = \iiint_A x
dx
dy = 0
\]

\[
I_{xx} = \iiint_A y^2
dx
dy = \frac{A}{12}(y_1^2 + y_2^2 + y_3^2)
\]

\[
I_{yy} = \iiint_A x^2
dx
dy = \frac{A}{12}(x_1^2 + x_2^2 + x_3^2)
\]

\[
I_{xy} = \iiint_A xy
dx
dy = \frac{A}{12}(x_1y_1 + x_2y_2 + x_3y_3)
\]
Thus, if the (local) origin is chosen at the centroid of the triangle, one then obtains

\[ k = t [ B_0^t E B_0 A + \frac{B_t^t}{B_x} E B_x I_{yy} + \frac{B_t^t}{B_y} E B_y I_{xx} + \]

\[ (\frac{B_t^t}{B_x} E B_y + \frac{B_t^t}{B_y} E B_x) I_{xy} ]. \]
CHAPTER 3

DEVELOPMENT OF THE TRIM5 AND TRIM4 ELEMENTS

The TRIM5 element is defined as a triangular element having five nodes with two sides possessing intermediate nodes (compatible with TRIM6 elements) while the third side has none (compatible with a TRIM3 element) (see Fig. 3.1).

The method employed in this effort for the computation of the characteristics of a TRIM5 element is based on the introduction of the sixth node to the third side thus producing a TRIM6 element. After the properties of the TRIM6 element have been established, the condition that the displacement of the sixth node is linearly dependent on those of nodes 1 and 3 is enforced. The third side will therefore always remain straight, whereas, the first two may assume a second order deflected shape.

Transformation from TRIM6 to TRIM5

For convenience, let it be assumed that the added sixth node is located at the middle of side 1-3, whereas, the fourth and fifth nodes are located at the middle of sides 1-2 and 2-3, respectively. It can be seen that if
Fig. 3.1 A TRIM 5 Element
Fig. 3.2 Displacement of Node 1 by Unity
Fig. 3.3 Displacement of Node 3 by Unity
Fig. 3.4 Displacement of Node 2 by Unity
each of the original five nodes displaces, one at a time, only the displacements of nodes 1 and 3 cause a displacement at node 6. Since side 1-3 remains straight after deformation, a unit displacement of node 1 in, say, the x direction will cause a displacement of node 6 by 1/2 in the same direction (see Fig. 3.2). In the same manner, the displacement of node 3 by unity will displace node 6 by 1/2 (see Fig. 3.3).

The displacement of any other node (nodes 2, 4 and 5) will not affect the displacement of node 6. Figure 3.4 shows, as an example, node 2 being displaced by unity while node 6 remains stationary.

Therefore, the u-displacement (displacement in the x-direction) of node 6 is the average of the u-displacements of nodes 1 and 3.

\[ u_6 = \frac{1}{2}(u_1 + u_3) \]  

(3-1)

Similarly, for the v-displacement (displacement in the y-direction)

\[ v_6 = \frac{1}{2}(v_1 + v_3) \]  

(3-2)

Thus, the element nodal displacement matrix \( p_5 \) of a TRIM5 element is related to those of the expanded TRIM6 element \( p_6 \) through the relationship
\[
\begin{bmatrix}
    u_1 \\
v_1 \\
u_2 \\
v_2 \\
u_3 \\
v_3 \\
u_4 \\
v_4 \\
u_5 \\
v_5 \\
u_6 \\
v_6
\end{bmatrix} = 
\begin{bmatrix}
    u_1 \\
v_1 \\
u_2 \\
v_2 \\
u_3 \\
v_3 \\
u_4 \\
v_4 \\
u_5 \\
v_5 \\
u_6 \\
v_6
\end{bmatrix} = 
\begin{bmatrix}
    1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
    0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
    0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
    0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\end{bmatrix}
\begin{bmatrix}
    u_1 \\
v_1 \\
u_2 \\
v_2 \\
u_3 \\
v_3 \\
u_4 \\
v_4 \\
u_5 \\
v_5 \\
u_6 \\
v_6
\end{bmatrix}
\]

or

\[
P_6 = T_{6,5} P_5
\]

(3-3b)

(12x1) (12x10)(10x1)

Relationship Between $P_6$ and $P_5$

The external work done by forces acting on the TRIM6 element

\[
= \frac{1}{2} P^t P_6
\]

(3-4)

Utilizing the relationship between $P_6$ and $P_5$ in Equation (3-3b)
\[
\frac{1}{2} P_6^t P_6 = \frac{1}{2} P_5^t T_{6,5}^t P_6 \quad (3-5)
\]

From the Principle of Conservation of Energy (P.C.E.), comparing Equation (3-5) with the external work done on the TRIM5 element \((\frac{1}{2} P_5^t P_5)\), it is seen that

\[
P_5 = T_{6,5}^t P_6 \quad (3-6)
\]

Relationship Between \(k_6\) and \(k_5\)

Substituting \(P_6 = k_6 P_6\) into Equation (3-6)

\[
P_5 = T_{6,5}^t k_6 P_6.
\]

Substituting Equation (3-3b) into the above equation

\[
P_5 = T_{6,5}^t k_6 T_{6,5} P_5. \quad (3-7)
\]

By examining Equation (3-7), it is obvious that the stiffness matrix of the TRIM5 element is given by

\[
k_5 = T_{6,5}^t k_6 T_{6,5} \quad (3-8)
\]

\(\begin{pmatrix} 10x12 \end{pmatrix} \begin{pmatrix} 10x12 \end{pmatrix} \begin{pmatrix} 12x12 \end{pmatrix} \begin{pmatrix} 12x10 \end{pmatrix}\)

\(k_5\) can be efficiently obtained from \(k_6\) using two nested DO loops (see SUBROUTINE SKE in Appendix C).
**Displacements and Stresses**

In calculating the displacements within a TRIM5 element the corner displacements of the five nodes are extracted from the structural displacement matrix \( \mathbf{r} \). The displacements of the sixth node are introduced by averaging the displacements of nodes 1 and 3 and the element is subsequently treated as a TRIM6 element as described previously.

Displacements for a point within the element is, from Equation (A-1),

\[
\mathbf{u} = \mathbf{T}_{u,p} \mathbf{p}_6 \\
= \mathbf{T}_{u,p} \mathbf{T}_{6,5} \mathbf{p}_5
\]

The stresses can be obtained by substituting Equation (A-6) into Equation (A-9)

\[
\mathbf{S} = \mathbf{E} \mathbf{T}_{e,p} \mathbf{p}
\]

where \( \mathbf{E} \) is the material stiffness matrix (generalized Young's modulus),
\( \mathbf{T}_{e,p} \) is the strain displacement matrix of the expanded TRIM6 element [see Equation (2-12)],

1. See Appendix A.
and \( p_6 \) is the corner displacement matrix of the TRIM6 element [see Equation (2-1)].

Substituting Equation (3-3b) into Equation (3-11) to obtain the stress matrix as

\[
S = E \, T_e, p_6 T_e, p_5. \tag{3-12}
\]

The TRIM4 Element

The TRIM4 element is defined as an element having four nodes with one side possessing an intermediate node (compatible with a TRIM6 element) while the other two sides have none (compatible with TRIM3 elements) (see Fig. 3.5).

The calculation of the properties for a TRIM4 element is performed in a manner similar to that of a TRIM5 element, i.e., by introducing additional nodes (nodes 5 and 6) to sides 2-3 and 3-1, thereby producing an expanded TRIM6 element (see Fig. 3.6). The displacements of each introduced nodes are again linearly interpolated from the two end nodes on the same side. Therefore, the \( u \)- and \( v \)-displacements of node 5 may be obtained by averaging the relevant displacements of nodes 2 and 3. Likewise, the displacements of node 6 are computed by averaging those of nodes 1 and 3.
Fig. 3.5 A TRIM4 Element

Fig. 3.6 A TRIM6 Element Expanded from a TRIM4 Element
\[ u_5 = \frac{1}{2}(u_2 + u_3) \]  
\[ v_5 = \frac{1}{2}(v_2 + v_3) \]  
\[ u_6 = \frac{1}{2}(u_1 + u_3) \]  
\[ v_6 = \frac{1}{2}(v_1 + v_3) \]  

(3-13a)  
(3-13b)

Thus, the element nodal displacement matrix of a TRIM4 \((p^4)\) is related to that of the expanded TRIM6 element \((p^6)\) through the relationship

\[
\begin{bmatrix}
    u_1 \\
    v_1 \\
    u_2 \\
    v_2 \\
    u_3 \\
    v_3 \\
    u_4 \\
    v_4 \\
    u_5 = \frac{1}{2}u_2 + \frac{1}{2}u_3 \\
    v_5 = \frac{1}{2}v_2 + \frac{1}{2}v_3 \\
    u_6 = \frac{1}{2}u_1 + \frac{1}{2}u_3 \\
    v_6 = \frac{1}{2}v_1 + \frac{1}{2}v_3
\end{bmatrix}
= 
\begin{bmatrix}
    1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
    0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
    0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
    0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
    0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
    0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
    .5 & 0 & 0 & 0 & .5 & 0 & 0 & 0 \\
    .5 & 0 & 0 & 0 & .5 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
    u_1 \\
    v_1 \\
    u_2 \\
    v_2 \\
    u_3 \\
    v_3 \\
    u_4 \\
    v_4
\end{bmatrix}
\]  

(3-14a)
or

\[ p_6 = T_{6,4} p_4 \]  

(3-14b)

where the transformation matrix \( T_{6,4} \) is a 12x8 matrix

\[
T_{6,4} = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
.5 & 0 & 0 & 0 & .5 & 0 & 0 & 0 \\
.5 & 0 & 0 & 0 & .5 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & .5 & 0 & 0 \\
0 & .5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & .5 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

(3-15)

Relationship Between \( p_6 \) and \( p_4 \)

The external work done by forces acting on the TRIM6 element

\[
= \frac{1}{2} p_6^t p_6
\]

(3-16)
Utilizing the relationship between $p_6$ and $p_4$ in Equation (3-14b),

$$\frac{1}{2} p_6^T p_6 = \frac{1}{2} p_4^T T_{6,4}^T P_6. \quad (3-17)$$

Comparing Equation (3-17) with the external work done on a TRIM4 element ($\frac{1}{2} p_4^T p_4$) and applying the Principle of Conservation of Energy, it is seen that

$$p_4 = T_{6,4}^T p_6 \quad (3-18)$$

**Relationship Between $k_6$ and $k_4$**

Substituting $p_6 = k_6 p_6$ and Equation (3-14b) into Equation (3-18) one obtains

$$p_4 = T_{6,4}^T k_6 T_{6,4} p_4 \quad (3-19)$$

By examining Equation (3-19), the stiffness matrix of the TRIM4 element ($k_4$) may be found to be

$$k_4 = T_{6,4}^T k_6 T_{6,4}. \quad (3-20)$$

**Displacements and Stresses**

The calculation of the displacements within a TRIM4 element can be performed in a manner similar to the TRIM5 element case, i.e., to treat the element as a TRIM6 element by introducing nodes 5 and 6.
The displacements for any point within the element may then be expressed as

\[ u = T_u, p_6, P_6 \]

\[ = T_u, p_6, T_6, 4P_4 \]  

(3-21)

The stresses can be derived as follows:

\[ S = E \cdot T_e, p_6, P_6 \]  

(3-22)

where \( E \) is the material stiffness matrix,

\( T_e, p_6 \) is the strain-displacement transformation matrix of the expanded TRIM6 element [see Equation (2-12)],

and \( P_6 \) is the corner displacement matrix of the TRIM6 element [see Equation (2-1)].

Substituting Equation (3-14b) into Equation (3-22) to obtain the stress matrix of a TRIM4 element

\[ S = E \cdot T_e, p_6, T_6, 4P_4 \]

Alternative Approaches

Alternative #1

A similar approach was mentioned by Meek (1971) in obtaining the stiffness matrix of either a TRIM5, TRIM4 or TRIM3 element by extracting from the matrix \( k_e \) of size
12x12 which, in turn, is obtained from the stiffness matrix of the TRIM6 element \( (k_6) \) through the transformation

\[
 k_g = T^t k_6 T. \tag{3-23}
\]

The matrix \( T \) to be applied to Equation (3-23) depends on the element type. For example, for a TRIM6 element

\[
 T = I \tag{12x12} \tag{3-24}
\]

For TRIM5 and TRIM3 elements, the \( T \) matrices are, respectively,

\[
 T = \begin{bmatrix}
 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
 .5 & 0 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
 0 & .5 & 0 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix} \tag{3-25}
and

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
.5 & 0 & .5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & .5 & 0 & .5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & .5 & 0 & .5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
.5 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & .5 & 0 & 0 & .5 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\end{bmatrix}
\]

(3-26)

After substituting the matrix \( T \) into Equation (3-23), the correct stiffness matrix is obtained by extracting the first 10x10, 8x8 or 6x6 submatrix from \( k_g \) for a TRIM5, TRIM4 or TRIM3 element, respectively.

The stiffness matrices from this approach are essentially identical to those obtained by the approach described earlier in this paper. However, it should be noted that the approach suggested by Meek requires the generation of another 12x12 matrix \( k_g \) [see Equation (3-23)], after obtaining the 12x12 stiffness matrix of the TRIM6
element, from which the stiffness matrices for other triangular elements are extracted. In contrast, the method described earlier in this paper yields directly, from the stiffness matrix of the TRIM6 element, the stiffness matrices for the TRIM5 and TRIM4 elements which are of the sizes 10x10 and 8x8, respectively. In another word, Meek's method could have been simplified by discarding the last 2 columns of the matrix $T$ for the TRIM5 element case and, similarly, the last 4 columns for the TRIM4 element case in order to obtain the same result with less computation time.

The stiffness matrix for the TRIM3 element, however, will be more efficiently computed by using linear interpolation functions to obtain directly a 6x6 matrix as treated in Appendix B of this paper and by Zienkiewicz (1971) rather than to extract a 6x6 submatrix from the $12x12 \frac{k}{g}$ matrix as expressed in Equation (3-23).

Alternative #2

Another approach for obtaining the stiffness matrices of the TRIM5 and TRIM4 elements will now be presented. Since the concepts are applicable to both TRIM5 and TRIM4 elements, only the derivation of the TRIM5 element will be discussed. The argument can be easily extended to the TRIM4 element case.
Instead of obtaining the stiffness matrix of a TRIM6 element from which the stiffness matrix of the TRIM5 element is computed, the desired (10x10) stiffness matrix may be computed directly.

One rewrites Equation (3-3) as

$$\mathbf{P}_6 = T_{6,5} \mathbf{P}_5$$

(3-27)

where

$$T_{6,5} = \begin{bmatrix}
I & 0 & 0 & 0 & 0 \\
0 & I & 0 & 0 & 0 \\
0 & 0 & I & 0 & 0 \\
0 & 0 & 0 & I & 0 \\
0 & 0 & 0 & 0 & I
\end{bmatrix}$$

(3-28)

with $I$ being a 2x2 identity submatrix, and $0$ is a 2x2 null submatrix.

For the TRIM6 case [see Equation (A-6)]

$$\mathbf{e}_{(3\times 1)} = T_{e,p_6} \mathbf{P}_6$$

(3-29)

(3x12) (12x1)

Substituting Equation (3-3) into the above equation

$$\mathbf{e}_{(3\times 1)} = T_{e,p_6} T_{6,5} \mathbf{P}_5$$

(3-30)

(3x12) (12x10) (10x1)
where
\[ T_{e,p_5} = T_{e,p_6} T_{6,5} \] (3-32)

Substituting Equation (2-12) into the above equation

\[ T_{e,p_5} = (B_0 + B_x + B_y) T_{6,5} \] (3-33)

\[ = [B^*_0 + B^*_x + B^*_y] \] (3-34)

where (see Equations 2-14 and 3-33)

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0.51 & 0.51 & 0 & 0 & 0
\end{bmatrix}
\] (12x10)

(3-35)

\[ = [(B_{01} + \frac{1}{2}B_{06}) B_{02} (B_{03} + \frac{1}{2}B_{06}) B_{04} B_{05}] \] (3-36)
In Equation (3-35), $I$ is a 2x2 identity sub-matrix and $0$ is a 2x2 null submatrix.

Similarly [see Equations (2-17), (2-19) and (3-33)],

$$
\begin{bmatrix}
B_x^* \\
B_y^*
\end{bmatrix} = \begin{bmatrix}
(B_{x1} + \frac{1}{2}B_{x6}) & B_{x2} & (B_{x3} + \frac{1}{2}B_{x6}) & B_{x4} & B_{x5} \\
(B_{y1} + \frac{1}{2}B_{y6}) & B_{y2} & (B_{y3} + \frac{1}{2}B_{y6}) & B_{y4} & B_{y5}
\end{bmatrix}
$$

(3x10)

and

$$
\begin{bmatrix}
B_x^* \\
B_y^*
\end{bmatrix} = \begin{bmatrix}
(B_{y1} + \frac{1}{2}B_{y6}) & B_{y2} & (B_{y3} + \frac{1}{2}B_{y6}) & B_{y4} & B_{y5} \\
(B_{x1} + \frac{1}{2}B_{x6}) & B_{x2} & (B_{x3} + \frac{1}{2}B_{x6}) & B_{x4} & B_{x5}
\end{bmatrix}
$$

(3x10)

The matrix $T_{e,p_5}$ from Equation (3-34) can readily be inserted into Equation (A-31) to obtain the stiffness matrix of the TRIM5 element

$$
k_5 = \int_{v} T_{e,p_5}^T E_T e_{e,p_5} dv
$$

(10x10) (10x3) (3x3) (3x10)

Equation (3-39) may finally be written in form of Equation (2-26), utilizing the nomenclature defined in Chapter 2, as

$$
k_5 = - t \left[ B_0^T E B_0 A + B_x^T E B_x I_{yy} + B_y^T E B_y I_{xx} + (B_x^T E B_x + B_y^T E B_y) I_{xy} \right]
$$

(3-40)
Alternative #3

In the previous approaches, a sixth point is added to a TRIM5 element to change it into a TRIM6 element and the existing computation scheme of the TRIM6 element is utilized. In the first two approaches, the stiffness matrix of the TRIM5 element is obtained by kinematic condensation from the stiffness matrix of the TRIM6 element. In the second alternative approach, the stiffness matrix of the TRIM5 element is computed from the matrix $T_{e,p_5}$ which, in turn, is obtained by kinematic condensation of the strain-displacement transformation matrix of the TRIM6 element $(T_{e,p_6})$. The main reason for treating the TRIM5 element as a kinematically condensed TRIM6 element is obvious as one can easily modify the existing TRIM6 program to yield the scheme for the TRIM5 element.

It will be shown here that the TRIM5 element may be computed directly from the beginning in a manner similar to that of the TRIM6 element.
Fig. 3.7 A TRIM6 Element Expanded from a TRIM5 Element

As opposed to allowing the movement of 6 nodal points to obtain Equation (2-3), only 5 nodal points may be allowed to move independently while the sixth point remains dependent on the movement of points 3 and 1 (see Fig. 3.7). In place of Equation (2-3) one then obtains the corresponding set of equations as

\[
\begin{align*}
    a_{01} + a_{x1}x_1 + a_{y1}y_1 + a_{xx1}x_1^2 + a_{xy1}x_1y_1 + a_{yy1}y_1^2 &= 1 \\
    a_{01} + a_{x1}x_2 + a_{y1}y_2 + a_{xx1}x_2^2 + a_{xy1}x_2y_2 + a_{yy1}y_2^2 &= 0 \\
    a_{01} + a_{x1}x_3 + a_{y1}y_3 + a_{xx1}x_3^2 + a_{xy1}x_3y_3 + a_{yy1}y_3^2 &= 0 \\
    a_{01} + a_{x1}x_4 + a_{y1}y_4 + a_{xx1}x_4^2 + a_{xy1}x_4y_4 + a_{yy1}y_4^2 &= 0 \\
    a_{01} + a_{x1}x_5 + a_{y1}y_5 + a_{xx1}x_5^2 + a_{xy1}x_5y_5 + a_{yy1}y_5^2 &= 0 \\
    a_{01} + a_{x1}x_6 + a_{y1}y_6 + a_{xx1}x_6^2 + a_{xy1}x_6y_6 + a_{yy1}y_6^2 &= \frac{1}{2}
\end{align*}
\]  

(3-41)
Four similar sets of equations associated with the movement of nodes 2, 3, 4 and 5 in either the x and y direction may be written jointly with Equation (3-41) to yield

\[
\begin{bmatrix}
1 & x_1 & y_1 & x_1^2 & x_1 y_1 & y_1^2 \\
1 & x_2 & y_2 & x_2^2 & x_2 y_2 & y_2^2 \\
1 & x_3 & y_3 & x_3^2 & x_3 y_3 & y_3^2 \\
1 & x_4 & y_4 & x_4^2 & x_4 y_4 & y_4^2 \\
1 & x_5 & y_5 & x_5^2 & x_5 y_5 & y_5^2 \\
1 & x_6 & y_6 & x_6^2 & x_6 y_6 & y_6^2
\end{bmatrix}
\begin{bmatrix}
a_{01} & a_{02} & a_{03} & a_{04} & a_{05} \\
a_{x1} & a_{x2} & a_{x3} & a_{x4} & a_{x5} \\
a_{y1} & a_{y2} & a_{y3} & a_{y4} & a_{y5} \\
a_{xx1} & a_{xx2} & a_{xx3} & a_{xx4} & a_{xx5} \\
a_{xy1} & a_{xy2} & a_{xy3} & a_{xy4} & a_{xy5} \\
a_{yy1} & a_{yy2} & a_{yy3} & a_{yy4} & a_{yy5}
\end{bmatrix}
= \begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
.5 & 0 & .5 & 0 & 0
\end{bmatrix}
\]

or

\[X \cdot A = I*\]
It is noted that $I^*$ is not an identity matrix in this case. The matrix $A$ may again be solved numerically.

Similar to the treatment in Chapter 2, the matrices $A_0, A_x, A_y, \ldots, A_{yy}$ are constructed from the matrix $A$. However, the matrices in this case are of size $2x10$. For example

$$A_0 = \begin{bmatrix} a_{01} & 0 \\ 0 & a_{01} \end{bmatrix}, \quad (3-43b)$$

$$A_{yy} = \begin{bmatrix} a_{yy1} & a_{yy2} & a_{yy3} & a_{yy4} & a_{yy5} \\ a_{yy1} & a_{yy2} & a_{yy3} & a_{yy4} & a_{yy5} \end{bmatrix}, \quad (3-44a)$$

The strain-displacement matrix $T_{e,p_5}$ may now be expressed as

$$T_{e,p_5} = \begin{bmatrix} B^* + B^*x + B^*y \end{bmatrix} \quad (3-45)$$
where the matrices $B_x^*$, $B_y^*$ and $B_y^*$ are of the size 3x10 and may be found to be

\[
B_0^* = \begin{bmatrix}
  a_{x1} & 0 & a_{x2} & 0 & a_{x3} & 0 & a_{x4} & 0 & a_{x5} & 0 \\
  0 & a_{y1} & 0 & a_{y2} & 0 & a_{y3} & 0 & a_{y4} & 0 & a_{y5} \\
  a_{y1} & a_{x1} & a_{y2} & a_{x2} & a_{y3} & a_{x3} & a_{y4} & a_{x4} & a_{y5} & a_{x5}
\end{bmatrix},
\]

(3.46a)

\[
B_x^* = \\
\begin{bmatrix}
  2a_{xx1} & 0 & 2a_{xx2} & 0 & 2a_{xx3} & 0 & 2a_{xx4} & 0 & 2a_{xx5} & 0 \\
  0 & a_{xy1} & 0 & a_{xy2} & 0 & a_{xy3} & 0 & a_{xy4} & 0 & a_{xy5} \\
  a_{xy1} & 2a_{xx1} & a_{xy2} & 2a_{xx2} & a_{xy3} & 2a_{xx3} & a_{xy4} & 2a_{xx4} & a_{xy5} & 2a_{xx5}
\end{bmatrix},
\]

(3.46b)

and

\[
B_y^* = \\
\begin{bmatrix}
  2a_{yy1} & 0 & 2a_{yy2} & 0 & 2a_{yy3} & 0 & 2a_{yy4} & 0 & 2a_{yy5} & 0 \\
  0 & a_{xy1} & 0 & a_{xy2} & 0 & a_{xy3} & 0 & a_{xy4} & 0 & a_{xy5} \\
  a_{xy1} & 2a_{yy1} & a_{xy2} & 2a_{yy2} & a_{xy3} & 2a_{yy3} & a_{xy4} & 2a_{yy4} & a_{xy5} & 2a_{yy5}
\end{bmatrix},
\]

(3.46c)
It is interesting to note that Equation (3-45) is identical to Equation (3-34) from alternative approach #2. Moreover, Equations (3-46a), (3-46b), and (3-46c) are essentially equivalent to Equations (3-36), (3-37), and (3-38), respectively. Thus, one can apply Equations (3-39) and (3-40) to calculate the stiffness matrix of the TRIM5 element.

This approach should be the most efficient among the four approaches described herein since the computation of the matrix $A$ in Equation (3-42) requires less time than that in Equation (2-4a). The computation time needed for the matrix $T_{e,p5}$ [see Equation (3-45)] is also less as the matrices $B^*, B^*_{x}$ and $B^*_{y}$ are smaller in size compared to those in the first three approaches. Furthermore, the kinematic condensation of the stiffness matrix is no longer necessary.
EXAMPLE PROBLEM

Two example problems have been solved using mixtures of TRIM6, TRIM5, TRIM4 and TRIM3 elements. Example 1 compares results from three finite element models representing the same plate subjected to uniformly distributed load (Fig. 4.1). It is a trivial case intended mainly to assure that the constant strain case can be represented by the various elements. Example 2 shows solutions to a stress concentration problem using first a pure TRIM3 model followed by two other models introducing different numbers of higher order elements in an attempt to show the superiority of results when higher order elements are used in the area of high stress concentration. Results obtained are compared with results by the "classical solution" (Timoshenko and Goodier, 1951).

Example 1

A cantilever plate of the size 4x2x1 inches is loaded by a uniform load of 1 lb. per sq. in. on side CD as shown in Fig. 4.1. Side AB is built in.
Model #1

Model #1 has 32 TRIM3 elements with 25 nodal points as shown in Fig. 4.2.*

Model #2

In model #2, 8 TRIM6 elements are used in the same arrangement as in example 1 (see Fig. 4.3).*

Model #3

For model #3, a mixture of TRIM3, TRIM4, TRIM5 and TRIM6 elements is employed. There are 17 nodal points (Fig. 4.4).

Comparison of results

The values of the stresses $\sigma_x$, $\sigma_y$, $\tau_{xy}$ and $\overline{\sigma}$ along the clamped edge of the plate are plotted for all three models in Figs. 4.5 and 4.6. The stress $\overline{\sigma}$ is the stress computed from the von Mises Criterion (Desai and Abel, 1972) which, in the plane stress case, can be expressed as

$$\overline{\sigma} = \sqrt{0.5[(\sigma_x - \sigma_y)^2 + \sigma_y^2 + \sigma_x^2 + 6\tau_{xy}]} \quad (4-1)$$

Results from the three models are in agreement.

Figure 4.7* shows a plot of the deflected shape of the plate superimposed on the undeflected figure.

*Figures marked with an asterisk (*) are plotted by the CALCOMP Plotter at the University Computer Center. However, rollers shown in Figs. 4.2, 4.3, 4.7 and 4.14 are hand drawn.
In Figures 4.8*, 4.9* and 4.10* the stresses at each node are plotted for the three models using the same scale of 1.28. These figures are self-explanatory.
Fig. 4.1 The Loaded Cantilever Plate for Example 1
Fig. 4.2 Model #1 (Example 1)
Fig. 4.3 Model #2 (Example 1)
Fig. 4.4 Model #3 (Example 1)
Fig. 4.5 $\sigma_x$ and $\sigma_y$ Stress Variation of a Cantilever Plate
Fig. 4.6 $\tau_{xy}$ and $\sigma$ Stress Variation of a Cantilever Plate
Fig. 4.7 Model #2 (Example 1) and its Deflected Shape
Fig. 4.8 Stress Presentation of Model #1 (Example 1)
Fig. 4.9 Stress Presentation of Model #2
(Example 1)
Fig. 4.10 Stress Presentation of Model #3
(Example 1)
Example 2

A square plate of size 10x10 inches with a hole, 2 inches in diameter, at the center, is loaded with a uniformly distributed load of 1 lb./in. as shown in Fig. 4.11. The thickness of the plate is 1 inch.

Due to the symmetry of both the plate and the loading condition, only a quarter of the structure is taken for the calculation of displacements and stresses. Furthermore, with the v-displacements along the x-axis and u-displacements along the y-axis being zero, the plate can be modeled with the use of rollers as shown in Figs. 4.12, 4.13 and 4.14.*

Model #1

Model #1 is made of 62 TRIM3 elements with 43 nodal points (Fig. 4.12).

Model #2

Ten intermediate nodes are added to elements near the hole where a high stress concentration has been found from the previous solution. There are 53 nodes with the same number of elements (see Fig. 4.13).
Model #3

More intermediate nodes are added to the second model, resulting in 85 nodes, as shown in Fig. 4.14.*

Comparison of Results

Figures 4.15, 4.16, 4.17 and 4.18 compare the stress distributions along the x-axis from the three models with those from the classical solution (Timoshenko and Goodier, 1951). The highest stresses are found to be $\sigma_y$ and $\bar{\sigma}$ at the edge of the hole. Again, $\bar{\sigma}$ is the stress obtained from Equation (4-1).

There is an improvement of results due to the introduction of higher order elements near the high stress concentration region. For example, at the edge of the hole where $\sigma_y$ is highest, the stresses $\sigma_y$ from models #1, #2 and #3 deviate 13.3%, 8% and 0.7%, respectively, from the corresponding value obtained by the classical solution. Similarly, the stresses $\bar{\sigma}$ from the three models differ by 16.6%, 8% and 0.0%, respectively, from the value obtained by the classical solution at the same point. The graphs for $\sigma_x$ and $\tau_{xy}$ are not of great interest in the problem since the stress values are relatively low.
In Fig. 4.19, the stresses at all nodes, for model #3, are plotted as vectors using the scaling factor 0.416. Figure 4.20 displays an enlarged portion near the hole of model #3.
Fig. 4.11 Square Plate with Hole
Fig. 4.12 Model #1 (Example 2)
Fig. 4.13 Model #2 (Example 2)
Fig. 4.14 Model #3 (Example 2)
Fig. 4.15 $\sigma_y$ Stress Distribution
Fig. 4.16 $\sigma$ Stress Distribution
Fig. 4.17 $\sigma_x$ Stress Distribution
Fig. 4.18 $\tau_{xy}$ Stress Distribution
Fig. 4.19 Stress Presentation of Model #3 (Example 2)
Fig. 4.20 An Enlarged Portion of Model #3
The TRIM5 and TRIM4 elements developed are used to solve example problems and the results obtained are satisfactory. In example 1, model #1, utilizing only TRIM3 elements, yields stress values almost identical to those obtained from models #2 and #3 which incorporate elements of higher order. This is because the strain throughout the plate is approximately constant. Therefore, the constant-strain TRIM3 elements used can furnish sufficiently accurate results. However, for the varying-strain case, as in example 2, TRIM3 elements do not yield satisfactory results especially in the area of high stress concentration.

In general, the employment of higher order elements does improve the results as indicated in example 2. The interfacing elements, TRIM5 and TRIM4, may be used in the future to change TRIM3 elements to TRIM6 elements, during a dynamic interactive modelling process (see Chapter 1), while compatibility at interfaces between different element types is maintained. Otherwise, a modified model made exclusively of TRIM6 elements would
have to be used if compatibility is to be satisfied along the interfaces of all elements. Thus, the family of TRIM6, TRIM5, TRIM4, and TRIM3 elements when used together permits one to perform a dynamic interactive analysis more economically. As mentioned before, the change of element types by the addition of a node may be performed automatically (see Chapter 1 and Appendix F).

The capability of plotting a model, its deflected shape, the state of stress as well as enlarging a local area for more accurate display offers a convenient means of investigating data obtained from solutions.

The author believes that Dynamic Interactive Modelling Analysis, supplemented by appropriate graphical display facilities, will play an important role in structural analysis in the future, particularly as applied to engineering design.
APPENDIX A

MATRIX FORMULATION OF ELEMENT PROPERTIES
IN THE FINITE ELEMENT METHOD

This Appendix is intended to offer a brief review, familiarizing readers of this paper with the matrix formulation used (Kamel and Liu, 1972, and Zienkiewicz, 1971).

In the Finite Element Method of Structural Analysis, a structure is divided into a "finite" number of subregions (or elements) each of which has a finite number of degrees of freedom. There are generally two approaches, the Force and Displacement Methods, which may be employed. Only the Displacement Method which has gained much popularity in recent years and is used in this study will be discussed here. In this method, the displacements of nodal points (or nodes) chosen at selected locations are the basic unknown parameters of the problem. These points may be located on the intersection of (or along) element boundaries. They may also be defined within the elements themselves as in the TRIM10 element.¹

¹ A triangular membrane element with 10 nodes, nine of which are on the boundaries, while the tenth is inside the element.
**Displacement Interpolation**

The state of displacement within each element is uniquely defined by a set of functions. Depending on element type, the functions may be complete or restricted polynomials. As an example, the \( u \)- and \( v \)-displacements within a TRIM3 element are obtained through superposition of three basic linear polynomials.\(^2\) The chosen functions uniquely relate the state of the displacements within each element to the element nodal displacements so that

\[
\mathbf{u} = \mathbf{T}_{u,p} \mathbf{p} \tag{A-1}
\]

where \( \mathbf{u} \) is the displacement function matrix of an internal point within an element

\[
\mathbf{u} = \begin{bmatrix} u(x,y) \\ v(x,y) \end{bmatrix} \tag{A-2}
\]

The two displacement functions \( u \) and \( v \) represent the displacement pattern in the \( x \) and \( y \) directions, respectively.

\( \mathbf{T}_{u,p} \) is a displacement interpolation matrix obtained from the chosen displacement interpolation functions and \( \mathbf{p} \) is the element node displacement matrix.

---

2. See Appendix B for detailed derivation of the TRIM3 element.
Strains

The strains within an element are obtained from the displacement functions and can therefore be related to the nodal displacements.

\[ \mathbf{e} = \mathbf{D}_{e,u} \mathbf{u} \]  
\[ = \mathbf{D}_{e,u} \mathbf{T}_{u,p} \mathbf{p} \]  
\[ = \mathbf{T}_{e,p} \mathbf{p} \]

where \( \mathbf{e} \) is the element strain matrix
\( \mathbf{D}_{e,u} \) is the Differential Operator matrix

\[ \mathbf{D}_{e,u} = \begin{bmatrix} \partial / \partial x & 0 \\ 0 & \partial / \partial y \\ \partial / \partial y & \partial / \partial x \end{bmatrix} \]
$T_{e,p}$ is the Strain-Nodal displacement interpolation matrix

$$T_{e,p} = D_{e,u} T_{u,p} \quad (A-8)$$

**Stresses**

Stresses at any point within the element (as well as on the boundaries) can be related to the strains in the linear elastic case using the material properties

$$\bar{S} = \bar{E} \bar{e} \quad (A-9)$$

where $\bar{S}$ is the Stress matrix of the element

$$\bar{S} = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{bmatrix}, \quad (A-10)$$

$E$ is the Material Stiffness matrix (or Generalized Young's Modulus)

$$E = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{(1-\nu)}{2} \end{bmatrix}, \quad (A-11)$$

and $\bar{e}$ is the Strain Matrix of the element.
\[ \text{Equivalent Nodal Forces} \]

We consider the particular case where all forms of energy, e.g., thermal energy, initial strains, etc., are absent except for the work done on the structure by external forces.

Let \( U_V, U_s \) define the distributed body and surface forces (loads) acting on a unit volume of the elastic material and a unit surface area, respectively,

\[
\begin{align*}
\mathbf{U}_V &= \begin{bmatrix} U_V(x,y) \\ V_V(x,y) \end{bmatrix} = \begin{bmatrix} U_V \\ V_V \end{bmatrix} \\
\mathbf{U}_s &= \begin{bmatrix} U_s(x,y) \\ V_s(x,y) \end{bmatrix} = \begin{bmatrix} U_s \\ V_s \end{bmatrix}
\end{align*}
\]

The body forces \( \mathbf{U}_V \) and \( \mathbf{U}_s \) must correspond to the \( u- \) and \( v- \)displacements in direction.

Let the matrix \( \mathbf{P} \) represent the element nodal forces whose components correspond to those of the nodal displacement matrix \( \mathbf{p} \) in both order and direction

\[
\mathbf{P} = \begin{bmatrix} U_1 \\ V_1 \\ U_2 \\ V_2 \\ \vdots \end{bmatrix}
\]  
(A-13)
The work done by body forces \( \mathbf{U}_v \) subject to displacements \( \mathbf{u} \), W.D.\( _v \), can be obtained by
\[
W.D.\_v = \frac{1}{2} \int_V \mathbf{u}^T \mathbf{U}_v \, dV \tag{A-14}
\]
where \( V \) is the total volume of the material acted upon by the body forces \( \mathbf{U}_v \).

Substitute Equation (A-1) into Equation (A-14) and utilize the transposition rules in matrix algebra
\[
W.D.\_v = \frac{1}{2} \int_V \mathbf{p}^T \mathbf{T}_s \mathbf{U}_v \, dV \tag{A-15}
\]
Since the corner displacement matrix \( \mathbf{p} \) is a constant matrix
\[
W.D.\_v = \frac{1}{2} \int_V \mathbf{p}^T \mathbf{U}_v \, dV \tag{A-16}
\]
Substitute the expression \( \int_V \mathbf{T}_s \mathbf{U}_v \, dV \) by \( \mathbf{P}_v \).
Thus, the work done by the body forces can be written as
\[
W.D.\_v = \frac{1}{2} \mathbf{P}_v^T \mathbf{P}_v \tag{A-17}
\]
\( \mathbf{P}_v \) is called the equivalent nodal loads due to body forces.

Similarly, if \( \mathbf{P}_s \) denotes the equivalent nodal forces due to surface forces \( \mathbf{U}_s \) acting on the surface \( S \),

3. From matrix algebra, if \( \mathbf{A} = \mathbf{B} \mathbf{C} \), then \( \mathbf{A}^T = \mathbf{C}^T \mathbf{B}^T \).
\( p_s \) may be found to be

\[
p_s = \int_S T_u^t p_s^u dS \quad (A-18)
\]

and the work done by the surface forces \( p_s \), \( W.D. \), is equal to

\[
W.D. = \frac{1}{2} p^t \int_S T_u^t p_s^u dS \quad (A-19)
\]

\[
= \frac{1}{2} p^t p_s. \quad (A-20)
\]

The equivalent nodal body and surface forces \((p_v \text{ and } p_s)\) are summed algebraically to yield the total equivalent nodal forces \(p\), i.e.,

\[
p = p_v + p_s \quad (A-21a)
\]

\[
= \int_V T_v^t p_v dV + \int_S T_u^t p_s^u dS. \quad (A-21b)
\]

The work done by the body and surface forces, \( W.D. \), can then be represented as

\[
W.D. = \frac{1}{2} p^t p.
\]

The work done on the structure is stored in the form of strain energy (Principle of Conservation of energy). Thus

\[
\frac{1}{2} p^t p = \frac{1}{2} \int_V S^t e dV
\]
\[ \mathbf{p}^T \mathbf{p} = \int_V \mathbf{S}^T \mathbf{e} \, dV \]  \hspace{1cm} (A-22)

where \( \frac{1}{2} \mathbf{S}^T \mathbf{e} \, dV \) is the strain energy due to the stresses \( \mathbf{S} \) and strains \( \mathbf{e} \) stored in the element of volume \( V \).

For small displacements (i.e., linear elasticity) the equivalent nodal forces \( \mathbf{p} \) are linearly dependent on \( \mathbf{p} \).
\( \mathbf{P} \) and \( \mathbf{p} \) are related by a matrix called the element stiffness matrix (\( \mathbf{k} \))

\[ \mathbf{P} = \mathbf{k} \mathbf{p}. \] \hspace{1cm} (A-23)

The element stiffness matrix is usually obtained from an expression to be derived next [see Equation (A-31)].

Rewriting Equation (A-22)

\[ \mathbf{P}^T \mathbf{P} = \int_V \mathbf{S}^T \mathbf{e} \, dV \] \hspace{1cm} (A-24)

Substituting Equation (A-9) into the above equation and remembering that \( \mathbf{E} \) is a symmetric matrix

\[ \mathbf{P}^T \mathbf{P} = \int_V \mathbf{e}^T \mathbf{E} \mathbf{e} \, dV \] \hspace{1cm} (A-25)

\[ = \int_V \mathbf{e}^T \mathbf{E} \mathbf{e} \, dV \] \hspace{1cm} (A-26)
Substituting Equation (A-6) into the above equation

\[ P^t P = \int_V p^t T_e^t p_E T_e p \, dV \]  \hspace{1cm} (A-27)

Since \( p \), the element nodal displacement matrix, is a constant matrix,

\[ P^t P = p^t \left[ \int_V T_e^t p_E T_e p \, dV \right] p \]  \hspace{1cm} (A-28)

Equation (A-28) is valid for any value of \( p \); thus

\[ P^t = p^t \left[ \int_V T_e^t p_E T_e p \, dV \right] \]  \hspace{1cm} (A-29)

or

\[ P = \left[ \int_V T_e^t p_E T_e p \, dV \right] p \]  \hspace{1cm} (A-30)

Comparing Equations (A-30) and (A-23)

\[ k = \int_V T_e^t p_E T_e p \, dV \]  \hspace{1cm} (A-31)

Equation (A-31) is the expression used to calculate an element stiffness matrix \( k \).
APPENDIX B

THE TRIM3 ELEMENT

The TRIM3 element (Kamel and Liu, 1972, and Zienkiewicz, 1971) is a constant strain triangular membrane element with three nodal points. In FELP2, the corner points must be numbered counterclockwise.

Fig. B.1 A TRIM3 Element

1. See Chapter 1.
Consider first a displacement vector for the u-displacements (displacements in X-direction) of the three corner points of a TRIM3 element.

\[ \mathbf{p}_u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \]  \hspace{1cm} (B-1)

The movement \( u \) of an arbitrary point \( q(x,y) \) can be written in the form of Equation (A-1) as

\[ [u] = T_{u,p} \mathbf{p}_u \]  \hspace{1cm} (B-2)

The interpolation matrix \( T_{u,p} \) for this case can be represented by

\[ T_{u,p} = [f_{u1} \quad f_{u2} \quad f_{u3}] \]  \hspace{1cm} (B-3)

where \( f_{u1}, f_{u2}, \) and \( f_{u3} \) are functions having the value of unity at corner points 1, 2 and 3, respectively. At other corner points these functions possess the value of zero. As an example, the function \( f_{u1} \) may be defined as

\[ f_{u1} = \frac{A_{q23}}{A} \]  \hspace{1cm} (B-4)

where \( A_{q23} \) is the area defined by points \( q, 2 \) and 3, \( A \) is the area defined by points 1, 2 and 3 (see Fig. B-1).
Using similar arguments for \( f_{u2} \) and \( f_{u3} \), \( T_{u,p_u} \) can then be written as

\[
T_{u,p_u} = \frac{1}{A} [A_{q23} \ A_{q31} \ A_{q12}].
\]

Thus, the displacement \([u]\) can be expressed as

\[
[u] = \frac{1}{A} [A_{q23} \ A_{q31} \ A_{q12}] \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}.
\] (B-5)

It can be shown that the interpolation matrix \( T_{u,p_v} \) for the \( v \)-displacement \([v]\) is identical to \( T_{u,p_u} \). That is

\[
T_{u,p_v} = \frac{1}{A} [A_{q23} \ A_{q31} \ A_{q12}]
\]

and \([v]\) may be obtained by

\[
[v] = \frac{1}{A} [A_{q23} \ A_{q31} \ A_{q12}] \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}.
\]

The movement of point \( q(x,y) \), in a general two-dimensional case, is [see Equations (A-1) and (A-2)]

\[
u = \begin{bmatrix} u(x,y) \\ v(x,y) \end{bmatrix} = T_{u,p} \quad \text{ (B-6)}
\]
where \( p \) is defined as \[ \text{see Equation (A-3)} \]
\[
p^t = [u_1 \ v_1 \ u_2 \ v_2 \ u_3 \ v_3]
\]
and
\[
T_{u,p} = \frac{1}{A} \begin{bmatrix}
A_{q23} \ 0 & A_{q31} \ 0 & A_{q12} \ 0 \\
0 & A_{q23} \ 0 & A_{q31} \ 0 & A_{q12}
\end{bmatrix} . \quad (B-7)
\]

The Strain-nodal displacement interpolation matrix \[ \text{see Equation (A-8)} \] may be found to be
\[
T_{e,p} = \begin{bmatrix}
\partial/\partial x \ 0 \\
0 & \partial/\partial y \\
\partial/\partial y & \partial/\partial x
\end{bmatrix} \frac{1}{A} \begin{bmatrix}
A_{q23} \ 0 & A_{q31} \ 0 & A_{q12} \ 0 \\
0 & A_{q23} \ 0 & A_{q31} \ 0 & A_{q12}
\end{bmatrix} . \quad (B-8)
\]

Equation (B-8) will eventually yield
\[
T_{e,p} = \frac{1}{2A} \begin{bmatrix}
-y_{32} \ 0 & -y_{13} \ 0 & -y_{21} \ 0 \\
0 & x_{32} \ 0 & x_{13} \ 0 & x_{21} \\
x_{32}-y_{32} & x_{13}-y_{13} & x_{21}-y_{21}
\end{bmatrix} . \quad (B-9)
\]
where \( x_{ij} = x_i-x_j \) and \( y_{ij} = y_i-y_j \). \quad (B-10)

The strain at any point of the triangle may be obtained by \[ \text{see Equation (A-6)} \]
\[
e = T_{e,p}p.
\]
Since the matrix $T_{e,p}$ is a constant matrix, the strains throughout a TRIM3 element are constant. As a result, the stresses, obtained by multiplying the strain matrix with a constant matrix $E$ [see Equations (A-9) and (A-11)], are also constant throughout the element.

The element stiffness matrix may be obtained by [see Equation (A-31)]

$$k = \int_V T_{e,p}^t E T_{e,p} \, dV$$

$$= At T_{e,p}^t E T_{e,p}$$

where $T_{e,p}$ is given by Equation (B-9), $A$ and $t$ are the area and thickness of the element, respectively.
APPENDIX C

USER'S GUIDE

The FELP2 Program is designed to solve two-dimensional elasticity problems using several types of finite membrane elements (TRIM6, TRIM5, TRIM4, TRIM3, and PARAM). A guide showing how to use the various subroutines of the system follows.

Subroutine ATEB(AL, A, B, YG, P, S, SS, MS) computes the expression \( a \cdot A^T E B = SS \) and stores the result in the square array \( SS \) where:

- \( AL = \) location of scalar \( a \)
- \( A = \) Array \( A \) of size 3x13
- \( B = \) Array \( B \) of size 3x13
- \( YG, P, S = \) values in the 3x13 matrix \( E \) as shown

\[
E = \begin{bmatrix}
YG & P & 0 \\
P & YG & 0 \\
0 & 0 & S
\end{bmatrix}
\]

- \( MS = \) dimension of the square array \( SS \), not smaller than 12.

Subroutine AVS computes the von Mises Stress Criterion \( \bar{\sigma} \) from \( \sigma_x, \sigma_y \) and \( \tau_{xy} \) for all the nodal points. For the plane stress case, the stress \( \bar{\sigma} \) can be expressed as
\[ \bar{\sigma} = \sqrt{0.5[(\sigma_x - \sigma_y)^2 + \sigma_y^2 + \sigma_x^2 + 6\tau_{xy}^2]} \]

Subroutine **CLEAR** initializes (clears) arrays \( \text{NGPT} \) and \( \text{NELT} \) in preparation for a new problem where:

- \( \text{NGPT} \) = Total number of nodal points
- \( \text{NELT} \) = Total number of elements

Entry point **CLEAR** initializes arrays \( \text{FAC}, \text{FACS} \) and \( \text{SF} \) where:

- \( \text{FAC} \) = Scaling factor used by all subsequent plots after subroutine **PMOD** is called
- \( \text{FACS} \) = Scaling factor used for plotting stress vectors in subroutine **PSTR**
- \( \text{SF} \) = Displacement scaling factor used in subroutine **PDMOD**.

Subroutine **CLEM** \((A, \text{MA}, \text{NA}, M, N)\) initializes (clears) matrix \( A \) of size \( M \times N \) stored in array \( A \), size \( \text{MAxNA} \).

Subroutine **CONG** \((A, \text{MA}, \text{NA}, M, N, B, \text{MB}, \text{NB}, C, \text{MC}, \text{NC})\) computes the expression \( A^t B A = C \), where

- \( A \) is a matrix of size \( M \times N \) stored in array \( A \), dimension \( \text{MAxNA} \) and
- \( B \), a matrix of size \( M \times M \) stored in array \( B \), dimension \( \text{MBxNB} \).
- Matrix \( C \) of size \( N \times N \) is stored in array \( C \), dimension \( \text{MCxNC} \).

Subroutine **CSTRE** initializes arrays \( \text{INPS}, \text{SVON} \) which will be used in subroutine **STRE**, where:
INPS is a one-dimensional array, size 80, used for counting the number of elements sharing each node.

TSNP is a 3x80 array containing stresses $\sigma_x$, $\sigma_y$ and $\tau_{xy}$ (which are stored in rows 1, 2 and 3, respectively) for each nodal point.

Subroutine DEFL computes deflections using Gaussian elimination method and prints out both the $u$- and $v$-displacements for each nodal point.

Subroutine ELAM $(YG,P,S)$ defines the generalized material stiffness matrix

$$E_{(3x3)} = \begin{bmatrix}
\frac{YG}{1-P^2} & \frac{YG\cdot P}{1-P^2} & 0 \\
\frac{YG\cdot P}{1-P^2} & \frac{YG}{1-P^2} & 0 \\
0 & 0 & S
\end{bmatrix}$$

where $YG,P,S$ are Young's Modulus, Poisson's Ratio and Shear Modulus for the material used, respectively.

Subroutine FRDM $(L)$ changes the freedom pattern of node $L$ where entry points SUPU and SUPV are used for suppressing the $u$- and $v$-displacements, respectively.

Subroutine GENPT $(N,X,Y)$ assigns the coordinates $(X,Y)$ to node $N$. 
Subroutine **LINF** (I1, I2, XD, YD, D, SL, SMM) determines the direction cosines (SL, SMM) of the line joining nodes I1 and I2 in the x-y plane. D is the length, XD and YD are the projections on the x and y axes of the line, respectively.

**Subroutine LISTS.** Subroutine LISTS computes and prints out the total number of degrees of freedom (NUNKT) and the half-band-width of the structural master stiffness. It also lists the total number of nodal points (NGPT), the total number of elements (NEIT), the coordinates of each nodal point, as well as the type and properties of each element.

**Subroutine LOAD (I, V).** Entry point INCBU increases the load at node I in the x direction by the amount V. Similarly, entry point INCBV increases the load at node I in the y direction by the amount V.

**Subroutine MAXMIN (VALUE, NPTS, NV, XU, XL, XR, YU, YL, YR)** finds the maximum (XU and YU) and minimum (XL and YL) values for the two rows of the 2xNV array VALUE. XR is the range of values for the first row given by XU = XL. YR is the range of values for the second row given by YU=YL. NPTS is the number of columns to be processed.

**Subroutine MINV (AA, AC, MS).** Matrix AC is the inverse of matrix AA, both are of the sizes MxM. MS is the dimension of arrays AA and AC.
Subroutine MMM (A,M1,N1,M,N,B,M2,N2,K,C,M3,N3).
Post-multiplies matrix A of size MxN by matrix B of size NxK, resulting in matrix C at size MxK. The dimensions of arrays A, B and C are M1xN1, M2xN2 and M3xN3, respectively.

Subroutine MTMM (A,M1,N1,MA,NA,B,M2,N2,MB,C,NC,D, M3,N3) computes the expression $A^t B C$ and stores it in array D where the array dimensions of A, B, C and D are M1xN1, M2xN2, M1xN1 and M3xN3, respectively. The matrix sizes of A, B, C and D are MAxNA, MAxNB, NBxNC and NAXNC, respectively.

Subroutine MXMNI (VALUE, NPTS, NV, XU, XL, XR) has a similar function to subroutine MAXMIN except that the array VALUE in this case is a one-dimensional array of size NV.

Subroutine PAGE skips a page in the printout.
Entry point SLASH4 skips 4 lines in the printout.
Subroutine PARAM (I,J,K,L,T) defines a parallelogram element, thickness T, connecting nodes I, J, K and L.

Subroutine PDMOD. This subroutine obtains the maximum nodal displacement values in both x and y directions. It automatically finds a displacement scaling factor so that the scaled maximum displacement is 0.2 inches when plotted unless the scaling was predefined. This factor is used in plotting the deflected model.
Subroutine PLOTDM (NP1, NP2, NP3, NP4, NP5, NP6, INP, IET).

This subroutine plots the deflected model of a TRIM6, TRIM5, TRIM4, TRIM3 or PARAM element. Both INP and IET are equal to 3 for any of the TRIM elements (TRIM6, TRIM5, TRIM4, and TRIM3) and are equal to 4 for the PARAM element. NP1, ..., NP6 are the 6 element nodal points numbered counterclockwise. If a node is non-existent, the number is given as zero.

Subroutine PLOTM (NP1, NP2, NP3, NP4, NP5, NP6, INP, IET) plots the model of a TRIM6, TRIM5, TRIM4, TRIM3 or PARAM element. As in subroutine PLOTDM, INP and IET are equal to 3 for any of the TRIM elements (TRIM6, TRIM5, TRIM4, and TRIM3) and are equal to 4 for the PARAM element. NP1, ..., NP6 are the 6 element nodal points numbered counterclockwise. If a node is non-existent, the number is given as zero.

Subroutine PMOD (XT, YT, SL, Y1, YT1, XZ) prepares for subsequent model plotting. The two input values, XT and YT, are the minimum abcissa and ordinate values for the structural model to be plotted. Subroutine PMOD will determine the maximum and minimum values for the model as well as the height (HTY1) and width (WIDX) of both the scaled and unscaled models. It will then select an appropriate origin (which is at the distance -X1, -Y1 from XT, YT) and plot a pair of axes (X and Y) with tic marks to be used as reference for the model to be plotted. A
scaling factor (FAC) is computed and its value printed out. However, if a particular scaling factor is preferred, FAC may be defined (with a value other than zero) prior to the subroutine call. All the subsequent plots will use the same scaling factor FAC. XZ is a computed value which may be used as the minimum abcissa (XT) in the next call to subroutine PMOD.

**Subroutine PPAR (FIA,SE,D12)** prepares for stiffness and stress computation of a PARAM element.

**Subroutine PRIBK** prints out the non-zero values from array BK containing the structure master stiffness matrix.

**Subroutine PRILD** prints out all nodal forces (loads) in both x and y directions.

**Subroutine PSTR** determines the largest nodal stress value obtained by using the von Mises Criterion (see subroutine AVS). Unless a scaling factor (FACS) for plotting the stress vector is predefined, with a non-zero value, the subroutine computes the scaling factor which will yield a maximum scaled stress vector whose length is 1.5 inches. The stress vector is drawn at a 60 degrees inclination to the X-axis.

**Subroutine PTRIM (A)** prepares for stiffness and stress computation of a TRIM3 element.
Subroutine PTRIMH (A,ZIYY,ZIXX,ZIXY,MS) prepares for stiffness and stress computation of a higher order triangular element (TRIM6, TRIM5 or TRIM4). It computes A, ZIYY, ZIXX, ZIXY where:

A = Area of the element

\[ ZIYY = \int_{A} x^2 \, dx \, dy = \frac{A}{12} (x_1^2 + x_2^2 + x_3^2) \]

\[ ZIXX = \int_{A} y^2 \, dx \, dy = \frac{A}{12} (y_1^2 + y_2^2 + y_3^2) \]

\[ ZIXY = \int_{A} xy \, dx \, dy = \frac{A}{12} (x_1y_1 + x_2y_2 + x_3y_3). \]

Points 1, 2 and 3 define a triangle. The origin for the x and y axes is taken at the centroid of the triangle as shown in Fig. C.1.

MS = Number of nodal points of the element. MS is equal to 4, 5 or 6, depending on whether the element is a TRIM4, TRIM5, or TRIM6 element.
Subroutine SMM \((S,A,MA,NA,M,N,B)\) multiplies matrix \(A\), whose size is \(M \times N\), by a scalar \(S\) and stores the results in \(B\). The array sizes of both \(A\) and \(B\) are \(M \times N\).

Subroutine SKE computes the element stiffness matrix for any of the element types used in the FELP2 Program (e.g., TRIM6, TRIM5, TRIM4, TRIM3, and PARAM and ROD2) and stores the values in array \(SK\).

Subroutine STIFF computes and assembles element stiffnesses into the structure master stiffness.

Subroutine STRE calculates element stresses for all elements used in the FELP2 Program.

Subroutine STRESS computes stresses for all the elements of the structure.

Fig. C.1 A Triangle with the Origin at the Centroid
Subroutine TRIM3 (I,J,K,T) defines a membrane triangular element thickness, T, connecting nodes I, J and K.

Subroutine TRIMH (II,12,13,14,15,16,MS,T). This subroutine defines a higher order membrane triangular element connecting all or some of nodal points (II, ...., I6). The element type is given by MS and the element thickness by T. II, I2 and I3 define the 3 corner points of the element. I4, I5 and I6 define the intermediate nodes on sides II-I2, I2-I3, and I3-II, respectively. For a TRIM6 element, MS equals 6 and nodes II, ...., I6 are defined. For a TRIM5, MS equals 5 and the first 5 nodes II, ...., I5 are defined, whereas, I6 has the value of zero. In the same manner, for a TRIM4 element, MS equals 4 and the first 4 nodes (II, ...., I4) have non-zero values. It is noted that the numbering of the nodes should be counterclockwise.

Subroutine UNLOAD initializes array BR which contains the values for nodal forces (loads).

An example to demonstrate the use of the program is given. No attempts are made to interpret the results of this example here.

A cantilever plate of the size 2x4 inches is loaded with a uniformly distributed load of 1 lb./in. along the right edge of the plate as shown in Figure C.1.
The thickness of the plate is taken as unity. A Finite Element model using TRIM6, TRIM5, TRIM4 and TRIM3 is shown in Figure C.2.

A Steering Program called Program T3456 showing the sequence of subroutine calls and results from the program is given here as an example.¹

¹ Before calling any plotting subroutine, Subroutine INITIAL must be called to initialize the plot file. A 999 call to the Subroutine PLOT should always be the last call to terminate the file. For detail of the plotting subroutines refer to the University Computer Center User's Manual and the Calcomp Plotter hand-out sheets.
PROGRAM T3456 (INPUT 9 OUTPUT 9 PUNCH TAPE 1)
ITAPE5=INPUT 9 TAPE5=OUTPUT 9 TAPE7=PUNCH
COMMON/ORIG/NET, NU NK T, IHBW, YM, SM, PR, NEL, NCP
1, TGPC(2, 80), LETY(80), LECP(6, 80), TEGD(3, 83)
2, BO(3, 13), BX(3, 13), BY(3, 13), NGPT
3, BK(60, 80), BR(80), SR(80), LBW(80), SK(12, 83)
4, TEP(3, 12), T1(3, 12), T2(3, 12), E(3, 3), E1(3, 3)
5, LCP(6), WD(8), D(12, 12), W1(12, 12)
COMMON/P1/ HEIGHT, DFL1(82), DFL2(82), TGPC1(82)
1, TGPC2(82), FAC, SF, FAC9, SADIC, WIDX
COMMON/TRI/INPS(80), TSNP(3, 80), TGPE(2, 8)
1SVON(80), ZZ
DIMENSION XX(35), YY(35)
CALL INITIAL (0, 1, 0, 3, 0, 0)
PRINT 20001
20001 FORMAT(8P=PLEASE REVERSE PAPER*)
CALL CLEAR
SADIC=80
FAC=1.5
DO 25 I=1, 5
XX(I)=0.
XX(9*I)=2.
IF(I* GT, 4) GOTO 25.
XX(I*5)=1.
IF(I* GT, 3) GOTO 25.
XX(I*14)=4.
25 CONTINUE
Z=0.
DO 27 I=1, 5
IF(I* EQ, 1) GOTO 26.
YY(I*4)=Z.
26 YY(I)=Z.
YY(I*9)=Z.
Z=Z* 5.
27 CONTINUE
YY(15)=0. $ YY(16)=1. $ YY(17)=2. DO 35 I = 1, 17.
35 CALL GENPT(I, XX(I), YY(I)).
DO 23 I=1, NGPT.
TGPC1(I)=TGPC(I)
23 TGPC2(I)=TGPC(2, I)
YG=1. E* 7
CALL ELAM (YG, 0, 0, 0, 0)
T=1.
CALL TRIMH(1, 12, 3, 6, 7, 2, 6, T)
CALL TRIMH(3, 14, 5, 9, 9, 4, 6, T)
CALL TRIMH(10, 12, 1, 11, 6, 0, 5, T)
CALL TRIMH(12, 14, 3, 13, 8, 7, 6, T)
CALL TRIMH(12, 10, 16, 11, 0, 0, 4, T)
CALL TRIMH(14, 12, 17, 13, 0, 0, 9, T)
CALL TRIM3(10,15,16,T)
CALL TRIM3(12,16,17,T)
CALL LISTS
CALL STIFF
CALL UNLOAD
CALL INCBU(15,5)
CALL INCBU(16,5)
CALL INCBU(17,5)
CALL PRILD
DO 51 I=1,5
   CALL SUPU(I)
51   CALL SUPV(I)
   CALL DEFL
   CALL PMOD(0.0,3,0, X1,Y1,HEIGHT,XZ1)
   CALL PLOTM (1,12,3,6,7,2,3,3)
   CALL PLOTM(3,14,5,8,9,4,9,3,3)
   CALL PLOTM (10,12,1,11,6,0,3,3)
   CALL PLOTM(12,14,3,13,8,7,3,3)
   CALL PLOTM (12,10,16,11,0,0,3,3)
   CALL PLOTM (14,12,17,13,0,0,3,3)
   CALL PLOTM(10,15,16,0,0,0,3,3)
   CALL PLOTM(12,16,17,0,0,0,3,3)
   CALL PDMOD
   CALL PLOTDM(1,12,3,5,7,2,3,3)
   CALL PLOTDM(3,14,5,8,9,4,9,3,3)
   CALL PLOTDM(10,12,1,11,6,0,3,3)
   CALL PLOTDM(12,14,3,13,8,7,3,3)
   CALL PLOTDM(12,10,16,11,0,0,3,3)
   CALL PLOTDM(14,12,17,13,0,0,3,3)
   CALL PLOTDM(10,15,16,0,0,0,3,3)
   CALL PLOTDM(12,16,17,0,0,0,3,3)
   CALL CSTR
   CALL STRESS
   CALL AVS
   FAC=1
   CALL PMOD(XZ1,3,0, X1,Y1,HEIGHT,XZ2)
   CALL PLOTM (1,12,3,6,7,2,3,3)
   CALL PLOTM(3,14,5,8,9,4,9,3,3)
   CALL PLOTM (10,12,1,11,6,0,3,3)
   CALL PLOTM(12,14,3,13,8,7,3,3)
   CALL PLOTM (12,10,16,11,0,0,3,3)
   CALL PLOTM(14,12,17,13,0,0,3,3)
   CALL PLOTM(10,15,16,0,0,0,3,3)
   CALL PLOTM(12,16,17,0,0,0,3,3)
   CALL PSTR
   PRINT 20002
20002 FORMAT(*PM=RESTORE PAPER NOW*)
   CALL PLOT(0.0,0.999)
   STOP
END
NO. OF UNKNOWNS 34  HALF BAND WIDTH 24

NO. OF POINTS 17  NO. OF ELEMENTS 8

PTo 1 X 0.00 Y 0.00
PTo 2 X 0.00 Y 0.50
PTo 3 X 0.00 Y 1.00
PTo 4 X 0.00 Y 1.50
PTo 5 X 0.00 Y 2.00
PTo 6 X 1.00 Y 0.50
PTo 7 X 1.00 Y 1.00
PTo 8 X 1.00 Y 1.50
PTo 9 X 1.00 Y 2.00
PTo 10 X 2.00 Y 0.00
PTo 11 X 2.00 Y 0.50
PTo 12 X 2.00 Y 1.00
PTo 13 X 2.00 Y 1.50
PTo 14 X 2.00 Y 2.00
PTo 15 X 4.00 Y 0.00
PTo 16 X 4.00 Y 1.00
PTo 17 X 4.00 Y 2.00

ELEMENTS

<table>
<thead>
<tr>
<th>TRIM6</th>
<th>1</th>
<th>12</th>
<th>3</th>
<th>6</th>
<th>7</th>
<th>2</th>
<th>T</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRIM6</td>
<td>3</td>
<td>14</td>
<td>5</td>
<td>8</td>
<td>9</td>
<td>4</td>
<td>T</td>
<td>1.0</td>
</tr>
<tr>
<td>TRIM5</td>
<td>10</td>
<td>12</td>
<td>1</td>
<td>11</td>
<td>6</td>
<td>T</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>TRIM6</td>
<td>12</td>
<td>14</td>
<td>3</td>
<td>13</td>
<td>8</td>
<td>7</td>
<td>T</td>
<td>1.0</td>
</tr>
<tr>
<td>TRIM4</td>
<td>12</td>
<td>10</td>
<td>16</td>
<td>11</td>
<td>T</td>
<td>1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRIM4</td>
<td>14</td>
<td>12</td>
<td>17</td>
<td>13</td>
<td>T</td>
<td>1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRIM3</td>
<td>10</td>
<td>15</td>
<td>16</td>
<td>T</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TRIM3</td>
<td>12</td>
<td>16</td>
<td>17</td>
<td>T</td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
GRID POINT DEFLECTIONS

<table>
<thead>
<tr>
<th>PT.</th>
<th>NO.</th>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.46934112E+08</td>
<td>9.57961597E+09</td>
</tr>
<tr>
<td>9.46934112E+08</td>
<td>9.57961597E+09</td>
</tr>
<tr>
<td>9.46934112E+08</td>
<td>9.57961597E+09</td>
</tr>
<tr>
<td>9.46934112E+08</td>
<td>9.57961597E+09</td>
</tr>
<tr>
<td>9.46934112E+08</td>
<td>9.57961597E+09</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.03552351E+07</td>
<td>4.68056424E+08</td>
</tr>
<tr>
<td>2.03552351E+07</td>
<td>4.68056424E+08</td>
</tr>
<tr>
<td>2.03552351E+07</td>
<td>4.68056424E+08</td>
</tr>
<tr>
<td>2.03552351E+07</td>
<td>4.68056424E+08</td>
</tr>
<tr>
<td>2.03552351E+07</td>
<td>4.68056424E+08</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.97060893E+07</td>
<td>2.95324947E+08</td>
</tr>
<tr>
<td>1.97060893E+07</td>
<td>2.95324947E+08</td>
</tr>
<tr>
<td>1.97060893E+07</td>
<td>2.95324947E+08</td>
</tr>
<tr>
<td>1.97060893E+07</td>
<td>2.95324947E+08</td>
</tr>
<tr>
<td>1.97060893E+07</td>
<td>2.95324947E+08</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.8790391E+07</td>
<td>5.12405213E+09</td>
</tr>
<tr>
<td>1.8790391E+07</td>
<td>5.12405213E+09</td>
</tr>
<tr>
<td>1.8790391E+07</td>
<td>5.12405213E+09</td>
</tr>
<tr>
<td>1.8790391E+07</td>
<td>5.12405213E+09</td>
</tr>
<tr>
<td>1.8790391E+07</td>
<td>5.12405213E+09</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.83719084E+07</td>
<td>2.32034165E+08</td>
</tr>
<tr>
<td>1.83719084E+07</td>
<td>2.32034165E+08</td>
</tr>
<tr>
<td>1.83719084E+07</td>
<td>2.32034165E+08</td>
</tr>
<tr>
<td>1.83719084E+07</td>
<td>2.32034165E+08</td>
</tr>
<tr>
<td>1.83719084E+07</td>
<td>2.32034165E+08</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.02748077E+07</td>
<td>6.48783155E+08</td>
</tr>
<tr>
<td>4.02748077E+07</td>
<td>6.48783155E+08</td>
</tr>
<tr>
<td>4.02748077E+07</td>
<td>6.48783155E+08</td>
</tr>
<tr>
<td>4.02748077E+07</td>
<td>6.48783155E+08</td>
</tr>
<tr>
<td>4.02748077E+07</td>
<td>6.48783155E+08</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.93126614E+07</td>
<td>3.15815400E+08</td>
</tr>
<tr>
<td>3.93126614E+07</td>
<td>3.15815400E+08</td>
</tr>
<tr>
<td>3.93126614E+07</td>
<td>3.15815400E+08</td>
</tr>
<tr>
<td>3.93126614E+07</td>
<td>3.15815400E+08</td>
</tr>
<tr>
<td>3.93126614E+07</td>
<td>3.15815400E+08</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.83569208E+07</td>
<td>2.39088684E+09</td>
</tr>
<tr>
<td>3.83569208E+07</td>
<td>2.39088684E+09</td>
</tr>
<tr>
<td>3.83569208E+07</td>
<td>2.39088684E+09</td>
</tr>
<tr>
<td>3.83569208E+07</td>
<td>2.39088684E+09</td>
</tr>
<tr>
<td>3.83569208E+07</td>
<td>2.39088684E+09</td>
</tr>
</tbody>
</table>
SUB PDMOD
MAX DEFL IS 4.0274808E+07
MAX DEFL USED IN PLOT IS 2.0000000E+01
SCALING FACTOR IS 4.9658834E+05
VON MISES STRESS

9.43028476E-01
8.93326099E-01
8.69715408E-01
8.85915910E-01
8.73718123E-01
9.89348813E-01
9.58655151E-01
9.21547342E-01
9.21940110E-01
1.00218456E+00
1.02243848E+00
1.03907106E+00
1.01578398E+00
1.01244674E+00
9.96169116E-01
1.00896841E+00
9.96087711E-01

STRESS LENGTH FACTOR LARGEST STRESS=
1.50000000E+00
1.44359713E+00
1.03907106E+00
Fig. C.2 A Cantilever Plate under Loading
Fig. C.3 Undeflected- and Deflected-Shape of the Cantilever Plate
Fig. C.4  (a) Stress Presentation of the Plate  
(b) Unscaled and Undeflected Shape of the Plate
APPENDIX D

NOMENCLATURE

\( A \) = area of a triangle
\( A \) = coefficient matrix of an element
\( A_0, A_x, A_y, A_{xx}, A_{yy} \) = coefficient matrices for \( T_u, p \)
\( B_0, B_x, B_y \) = coefficient matrices for \( T_e, p \)
\( D_{e,u} \) = differential operator matrix relating the strain distribution to the displacement field
\( D_x, D_y \) = constant matrices associated with \( D_{e,u} \)
\( \varepsilon \) = strain matrix
\( E \) = material stiffness matrix (generalized Young's Modulus)
\( f_{u_i} \) = element interpolation function for u-displacement associated with node \( i \)
\( I \) = identity matrix
\( I_x \) = \( \int \int x \, dx \, dy \)
\( I_{xx} \) = \( \int \int x^2 \, dx \, dy \)
\( I_{xy} \) = \( \int \int xy \, dx \, dy \)
\( I_y \) = \( \int \int y \, dx \, dy \)
\( I_{yy} \) = \( \int \int y^2 \, dx \, dy \)
(k) = element stiffness matrix

(k₄, k₅, k₆) = element stiffness matrices of TRIM₄, TRIM₅ and TRIM₆ elements, respectively

(p) = element nodal displacement matrix

(ρ₄, ρ₅, ρ₆) = nodal displacement matrices for elements TRIM₄, TRIM₅ and TRIM₆, respectively

(ρ₆, ρᵥ) = equivalent element nodal force matrices associated with surface and body forces, respectively

(ρ₄, ρ₅, ρ₆) = element nodal force matrices for elements TRIM₄, TRIM₅ and TRIM₆, respectively

(S) = stress matrix of an element

(t) = thickness of an element

(Tₑ,p) = strain and nodal displacement interpolation matrix

(Tₑ,p₆) = strain and nodal displacement interpolation matrix for a TRIM₆ element

(Tᵤ,p) = element displacement interpolation matrix

(T₆,₄) = transformation matrix relating ρ₆ and ρ₄

(T₆,₅) = transformation matrix relating ρ₆ and ρ₅

(u) or \( \mathbf{u}(x,y) \) = displacement function matrix of an internal point within an element
\( U_s \) = distributed surface force matrix per unit area of the surface

\( U_v \) = distributed body force matrix per unit volume of the material

\( W.D._s, W.D._v \)

\( = \) work done by surface forces \( U_s \) and body forces \( U_v \), respectively

\( x, y \) = the abcissa and ordinate of an arbitrary point

\( x_i, y_i \) = the abcissa and ordinate of point \( i \)
APPENDIX E

LISTING OF SUBROUTINES USED

IN PROGRAM FELP2
SUBROUTINE ATEB(AL, A, B, YG, P, S, SS, MS)
C: *** MULTIPPLY MATRICES A(TRANSPOSED) * E * B = SS(12*12)
DIMENSION SS(MS, MS), A(3,13), B(3,13)
C: *** A WHEN CALLED IS STILL A (3*12) MATRIX
DO 4 I = 1, 11, 2
   C1 = A(1, I)
   D1 = A(2, I * 1)
DO 4 J = 1, 11, 2
   C2 = B(1, J)
   D2 = B(2, J * 1)
   SS(I, J) = SS(I, J) * AL*(C1 * C2 * YG * D1 * D2 * S)
   SS(I + 1, J) = SS(I + 1, J) * AL*(C2 * D1 * P * C1 * D2 * S)
   SS(I, J + 1) = SS(I, J + 1) * AL*(C1 * D2 * P * C2 * D1 * S)
   SS(I + 1, J + 1) = SS(I + 1, J + 1) * AL*(D1 * D2 * YG * C1 * C2 * S)
4 CONTINUE
RETURN
END

SUBROUTINE AVS
COMMON/ORIG/NELT, NUNKT, IH8W, YM, SM, PR, NEL, NC
1, TGPC(2,80), LETY(80), LECP(6,80), TEGD(3,8)
2, B0(3,13), BX(3,13), BY(3,13), NGPT
3, BK(60,80), BR(80), SR(80), LBW(80), SK(12,13)
4, TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5, LCP(6), WD(8), D(12,12), W1(12,12)
COMMON/TRI/INPS(80), TSNP(3,80), TGPE(2,8)
1, SVON(80), ZZZ
D01 = 1, NGPT
IF(INPS(I) .EQ. 1) G0T04
   Z = INPS(I)
   D02M = 1, 3
2, TSNP(M, I) = TSNP(M, I) / Z
4, A1 = TSNP(1, I) $ A2 = TSNP(2, I) $ A3 = TSNP(3, I)
   SVON(I) = SQRT((A1 - A2)**2 + A2**2 + A3**2 + 6.0*3*2))
1 CONTINUE
PRINT 14
C 14 FORMAT(/8X/PT, NO. NO. OF ELEatal, SIG X*15X
   1*SIG Y, TAU XY, VON MESIS STRESS*)
C PRINT 15(I, INPS(I), (TSNP(M, I), M = 1, 3)
C 1SVON(I) = 1, NGPT)
C 15 FORMAT(/1X/19*I15,5X 4E20,8)
   PRINT 17SVON(I) = 1, NGPT)
   PRINT 17SVON(I) = 1, NGPT)
C 17 FORMAT(13X*VON MISES STRESS* 2(/) (17X*E20,8))
RETURN
END

SUBROUTINE CLEAR
COMMON/ORIG/NELT, NUNKT, IH8W, YM, SM, PR, NEL, NC
COMMON/PR/HEH/PRI(80),TGP/(a2)
1,TGP/(a2),FAC,SF,FAC,SF,SADIC,WIDX

C  ***  PREPARE FOR NEW PROBLEM
C  ***  CLEAR IS STILL THE SAME  CLEAR1 ALSO CLEARS
C  ***  FAC,FACS,SF
PRINT 100
100 FORMAT (1H1,5X,* NEW PROBLEM*,5(/))

1 NGPT=0
NELT =0
RETURN
ENTRY CLEAR1
FAC=0.,$ FACS=0.,$ SF=0.
GO TO 1

END

SUBROUTINE CLEM (A,MA,NA,M,N)
DIMENSION A (MA,NA)
DO 1 I=1,M
DO 1 J=1,N
1 A(I,J) = 0.0
RETURN
END

SUBROUTINE CONG (A,MA,NA,M,N98,MB,NB,C,MC,NC)
DIMENSION A (MA,NA),B(MB,NB),C(MC,NC)
C  ***  FORMS A * T * B * A IN C
DO 1 I=1,N
DO 1 J=1,N
DO 1 K=1,M
DO 1 L=1,M
1 C(I,J) = C(I,J) * A(L,I) * A(K,J) * B(L,K)
RETURN
END

SUBROUTINE CSTRE
COMMON/ORIG/NELT,NUNKT,INHBW,YM,SM,PR,NEL,NCO
1,TGP/(a2),LETY/(a2),LEC/(6,80),TEGD/(3,8A)
2,B0(3,13),BX(3,13),BY(3,13),NGPT
3,BK(60,80),BR(80),SR(80),LBW(80),SK(12,12)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LC/(6),WD(8),D(12,12),W(12,12)
COMMON/TRI/INPS(80),TSNP(3,80),TGP/(2,8)
1SVM(80),ZZZ
C  ***  TO BE CALLED BEFORE CALLING SUB STRESS TO
C  ***  INITIALIZE SOME ARRAYS IN SUBSTRE, TRIM3,4,5,6 ONLY
SUBROUTINE DEFL

C COMPUTE DEFLECTIONS USING GAUSSIAN ELIMINATION
C
COMMON/ORIG/NElt,NUNKT,IHBW,YM,SM,PR,NEL,NGPT
1.TGPC(2,80),LETY(80),LECP(6,80),TEGD(3,80)
2.B0(3,13),BX(3,13),BY(3,13),NGPT
3,BK(60,80),BR(80),SR(80),LBW(80),SK(12,1?)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LCP(6),W0(8),D(12,12),W1(12,12)
COMMON/P1/HEIGHT,DFL1(82),DFL2(82),TGPC1(82)
1,TGPC2(82),FAC,SF,FACS,SADIC,WIDX
DO 10 I=1,NUNKT
10 SR(I) = BR(I)
NM1 =NUNKT-1
DO 1 ID = 1,NM1
C ELEMINATE ENTRIES BENEATH DIAGONAL ID
IDP = ID+1
IL = ID*IHBW+1
IF (IL .GT. NUNKT) IL=NUNKT
DO 2 I=IDP,IL
C ELEMINATE ENTRY K(I,ID) = TEST IF ALREADY ZERO
IF (BK(ID,ID) .EQ. 0.0) GOTO 2
C COEFFICIENT = K(I,ID)/K(ID,ID)
C REPLACE ROW I BY ROW I + ROW I * C
C
DO 3 J=1,IL
3 BK(I,J+1-ID) = BK(I,J+1-ID) + C*BK(ID,J+1-ID)
C
SR(I) =SR(I) +C*SR(ID)
2 CONTINUE
1 CONTINUE
C MATRIX NOW IN UPPER TRIANGULAR FORM. PROCEED
C TO BACK-SUBSTITUTION
DO 4 ID1 = 2,NUNKT
-ID = NUNKT+2-ID1
C ELEMINATE ENTRIES ABOVE K(ID,ID)
IDM = ID - 1
IB = ID+1-IHBW
IF (IB .LT. 1) IB=1
DO 5 I=IB,IDM
C = -BK(I,ID+1-ID)/BK(ID,1)
SR(I) =SR(I) +C*SR(ID)
CONTINUE

DEVIDE BY DIAGONAL ELEMENTS

DO 6 I = 1, NUNKT
   6 SR(I) = SR(I) / BK(I,1)

PRINT DEFORMATIONS

CALL PAGE
CALL SLASH
PRINT 20,

20 FORMAT ( 'GRID POINT DEFLECTIONS' )

PRINT 21

21 FORMAT ( 'GRID POINT DEFLECTIONS' )

RETURN

END

SUBROUTINE ELAM (YG, P, S)

DEFINE ELASTIC MATERIAL

COMMON/ORIG/NELT, NUNKT, IHBDW, YM, SM, PR, NEL, NC

1, TPDC(2, 80), LETY(80), LECP(6, 80), TEGD(3, 80)
2, B0(3, 13), BX(3, 13), BY(3, 13), NGPT
3, BK(60, 80), BR(80), SR(80), LBW(80), SK(12, 15)
4, TEP(3, 12), T1(3, 12), T2(3, 12), E(3, 3), E1(3, 3)
5, LCP(6, 80), WD(8), D(12, 12), W1(12, 12)

YM = YG
PR = P
SM = S

IF (PR .EQ. 0.0) PR = 1.0 / 3.0
IF (SM .EQ. 0.0) SM = YG / (1.0 + PR)

E(1, 1) = YG / (1.0 + PR)
E(2, 2) = E(1, 1)
E(1, 2) = E(1, 1) * PR
E(2, 1) = E(1, 2)
E(1, 3) = 0.0
E(3, 1) = 0.0
E(2, 3) = 0.0
E(3, 2) = 0.0
E(3, 3) = SM
RETURN

END

SUBROUTINE FRDM (L)

CHANGE FREEDOM PATTERN

COMMON/ORIG/NELT, NUNKT, IHBDW, YM, SM, PR, NEL, NC

1, TPDC(2, 80), LETY(80), LECP(6, 80), TEGD(3, 80)
2, B0(3, 13), BX(3, 13), BY(3, 13), NGPT
3, BK(60, 80), BR(80), SR(80), LBW(80), SK(12, 15)
ENTRY SUPU
K = 2*L
1
3 BR(K) = 0.0
DO 2 J = 1, 1*1HBW
2 BK(K, J) = 0.0
I1 = K + 1 - 1*1HBW
IF (I1 .LT. 1) I1 = 1
DO 4 I = 1, K
4 BK(I, K - 1) = 0.0
BK(K, 1) = 0.0
RETURN
ENTRY SUPV
K = 2*L
GO TO 3
END

SUBROUTINE GENPT (Ng, X, Y)
COMMON/ORIG/NEL, NUNKT, 1*1HBW, YM, SM, PR, NEL, NC
1 TGPC(2*80), LETY(80), LECP(6*80), TEGD(3*80)
2 B0(3*12), BX(3*13), BY(3*13), NGPT
3 BK(60*80), BR(80), SR(80), LBW(80), SK(12*12)
4 TEP(3*12), T1(3*12), T2(3*12), E(3*3), E1(3*3)
5 LCP(6), WID(8), D(12*12), W1(12*12)
C: *** GENERATE GRID POINT
NGPT = NGPT + 1
TGPC(1*N) = X
TGPC(2*N) = Y
RETURN
END

SUBROUTINE LINE (I1, I2, XD, YD, D, SL, SMM)
COMMON/ORIG/NEL, NUNKT, 1*1HBW, YM, SM, PR, NEL, NC
1 TGPC(2*80), LETY(80), LECP(6*80), TEGD(3*80)
2 B0(3*12), BX(3*13), BY(3*13), NGPT
3 BK(60*80), BR(80), SR(80), LBW(80), SK(12*12)
4 TEP(3*12), T1(3*12), T2(3*12), E(3*3), E1(3*3)
5 LCP(6), WID(8), D(12*12), W1(12*12)
C: *** ORIGINALLY THIS SUB IS CALLED LINE BUT IS CHANGED
C: *** TO AVOID HAVING AN IDENTICAL NAME WITH THE
C: *** PLOTTER'S SUB LINE, FOR LINE I1*I2 FIND (X2-X1)
C: *** (Y2-Y1) LENGTH, DIR, COSINES
X1 = TGPC(1*I1)
X2 = TGPC(1*I2)
Y1 = TGPC(2*I1)
Y2 = TGPC(2*I2)
XD = X2 - X1
YD = Y2 - Y1
D = SQRT(XD**2 + YD**2)
SL = XD/D
SMM = YD/D
RETURN

SUBROUTINE LISTS
COMMON/ORIG/NEL, NUNKT, IHBW, YM, SM, PR, NEL, NCS
1, TPC(2,80) LETY(80), LEC(6,80), TEGD(3,8)
2, B0(3,13), BX(3,13), RY(3,13), NGPT
3, BK(60,80), BR(80), SR(80), LBW(80), SK(12,12)
4, TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5, LCP(6), WD(8), D(12,12), W1(12,12)

C +++ FINALIZE STRUCTURAL MODEL
DIMENSION LNCP(8)
DATA LNCP/2,3,4,5,6,0,0/
NUNKT = 2 * NGPT

C +++ DETERMINE LOCAL HALF-BAND-WIDTH
DO 1 I = 1, NUNKT
1 LBW(I) = 1
IHBW = 2
DO 2 N = 1, NELT
C NO. OF CORNER POINTS PER ELEMENT
IT = LETY(N)
NCP = LNCP(IT)
DO 5 I = 1, NCP
IS = LECP(I,N)
DO 6 J = 1, NCP
JS = LECP(J,N)
IM = IS
IF (JS LT IM) IM = JS
IM = 2*IM - 1
IBW = 2*IABS(IS-JS) + 2
IF (IBW GT LBW(IM)) GOTO 6
IF (IBW GT IHBW) IHBW = IBW
LBW(IM) = IBW
LBW(IM+1) = IBW-1
6 CONTINUE
5 CONTINUE
2 CONTINUE
CALL PAGE
CALL SLASH6
PRINT 7, NUNKT, IHBW
7 FORMAT(///, 'NO. OF UNKNOWNS *I3',
      1 'HALF BAND WIDTH *I3')
C 7 FORMAT(///, 'NO. OF UNKNOWNS *I4',
C 1 'HALF BAND WIDTH *I4')
PRINT 3, NGPT, NELT
C 3 FORMAT(///, 'NO. OF POINTS *I5,5X',
C 1 'NO. OF ELEMENTS *I5,/')
SUBROUTINE LOAD (I, V)
C *** INCREMENT FORCE IN X DIRECTION BY V AT ST. I
C
COMMON/ORIG/NELT,NUNKT,INKBW,YMSM,PR,NEL,INC
1,TGPC(2*80),LETY(80),LECP(6*80),TEGD(3*80)
2,BK(60,80),BR(80),SR(80),LBW(80),SK(12*12)
3,TEP(3*12),T1(3*12),T2(3*12),E(3*3),E1(3*3)
5,LCP(6),WD(8),D(12*12),W1(12*12)
ENTRY INCBU
BR(2*I-1) = BR(2*I-1)*V
RETURN
ENTRY INCBV
JJ=2*I
SUBROUTINE MAXMIN(VALUE,NPTS,NV,XU,XL,YU,YL,YR)
DIMENSION VALUE(2,NV)
C *** FIND MAX $ MIN OF A (2XNV) ARRAY
XU=VALUE(1,1) $ XL=XU $ YU=VALUE(2,1) $ YL=YU
DO 511 = 2,NPTS
  IF(VALUE(1,1)>XL) XL=VALUE(1,1)
  IF(VALUE(1,1)<XL) XL=VALUE(1,1)
  IF(VALUE(2,1)>YL) YL=VALUE(2,1)
  IF(VALUE(2,1)<YL) YL=VALUE(2,1)
51 CONTINUE
XR=XU-XL $ YR=YU-YL:
RETURN
END

SUBROUTINE MINV(AA,M,A,C,M)
COMMON/ORIG/NELT,NUNKT,IHBW,YM,SM,PR,NEL,NC
17GTEC(10,80),LETY(80),LECP(6,80),TEGD(3,8)
280(3,13),8X(3,13),8Y(3,13),8P,8T
3G8K(60,80),8R(80),8LBW(80),8SK(12,12)
48TEP(3,12),8T(3,12),8T(3,12),8E(3,3),8E(3,3)
58LCP(6),8D(12,12),8D(12,12)
C *** MATRIX INVERSION OF AA (SIZE MxM) AND STORED IN AC,
C *** AA DESTROYED
DIMENSION AA(M,M),AC(M,M)
NB=M
C *** CREATE MATRIX AC=I
DO 20 I=1,M
  DO 21 J=1,M
  21 AC(I,J)=0.0
  20 AC(I,I)=1.0
MO=M-1
DO 4006 I=1,MO
K=I+1
DO 4001 J=K,M
  IF (ABS(AA(J,I))=ABS(AA(I,J))) 4001,4001,4001
  4000 INTERCHANGE ROWS
  4002 DO 4003 N=1,M
  V=AA(I,N)
  AA(I,N)=AA(J,N)
  4003 AA(J,N)=V
  DO 4010 N=1,NB
  V=AC(I,N)
  AC(I,N)=AC(J,N)
  4010 AC(J,N)=V
  4001 CONTINUE
IF (AA(I,i)) 4008, 4012, 4008
4012 PRINT 4013
4013 FORMAT (/52H MATRIX IS SINGULAR )
RETURN
C NULIFY REST OF COLUMN
IF (ABS(AA(I,i)) LT 1.0E-7) PRINT 901, I, AA(I,I)
901 FORMAT (/52H MINV XXXXX I*AA(I,I)=# I5,E20.8)
4008 DO 4011 J=K+M
F=AA(J,i)/AA(I,i)
DO 4004 N=K+M
4004 AA(J,N)=AA(J,N)*F*AA(I,N)
DO 4011 N=1,NB
4011 AC(J,N)=AC(J,N)*F*AC(I,N)
4006 CONTINUE
C BACK SUBSTITUTION
113 JD=M
C ELEMINATE ELEMENTS ABOVE A(JD,JD)
117 I=JD-1
IF (ABS(AA(JD,JD)) LT 1.0E-7) PRINT 912, JD, AA(JD,JD)
912 FORMAT (/52H MINV XXXXX JD*AA(JD,JD)=# I5,E20.8)
115 F=AA(I,JD)/AA(JD,JD)
C MODIFY ROW I
DO 124 J=1,NB
124 AC(I,J)=AC(I,J)*F*AC(JD,J)
I=I+1
IF (I) 116, 116, 115
116 JD=JD-1
IF (JD=1) 118, 118, 117
C DIVIDE BY DIAGONAL ELEMENTS
118 DO 119 I=1,M
DO 119 J=1,NB
119 AC(I,J)=AC(I,J)/AA(I,I)
RETURN
END
SUBROUTINE MMM (A, M1, N1, M2, N2, K, C, M3, N3)
DIMENSION A(M1,N1), B(M2,N2), C(M3,N3)
DO 1 I=1,M
DO 1 J=1,K
S=0.0
DO 2 L=1,N
2 S=S+A(I,L)*B(L,J)
C(I,J)=S
1 CONTINUE
RETURN
END
SUBROUTINE MM1 (A, M1, N1, MA, NA, B, M2, N2, NB, C, NC, D, M3, N3)
DIMENSION A(M1,N1), B(M2,N2), C(M1,N1), D(M3,N3)
DO 1 I=1,NA

DO 1 J=1,NC
S = 0.0
DO 2 K=1,MA
DO 2 L=1,MB
2 S=S+A(K,I)*B(K,L)*C(L,J)
D(I,J)=D(I,J)+S
1 CONTINUE
RETURN
END

SUBROUTINE MXMN1(VALUE,NPTS,NV,XU,XL,XR)
DIMENSION VALUE(NV)
C - - - - - FIND MAX $ MIN OF THE ARRAY VALUE(NV)
XU=VALUE(1) XL=XU
DO 51 I=2,NPTS
IF(VALUE(I).GT.XU) XU=VALUE(I)
IF(VALUE(I).LT.XL) XL=VALUE(I)
51 CONTINUE
XR=XU-XL
RETURN
END

SUBROUTINE PAGE
PRINT 1
1 FORMAT(1H1)
RETURN
ENTRY SLASH4
PRINT 4
4 FORMAT(///)
RETURN
ENTRY SLASH6
PRINT 6
6 FORMAT( //////)
RETURN
END

SUBROUTINE PARAM (I,J,K,L,T)
C - - - - DEFINE PARALLELOGRAM MEMBRANE CRNRS, I,J,K,L
C - - - - THICKNESS T
COMMON/ORIG,NELT,UNKT,IMAN,YM,SM,PR,NEL,NC
1,TGPC(2,80),LETY(80),LECP(6,80),TEGD(3,8)
2,BO(3,13),BX(3,13),BY(3,13),NGPT
3,BK(60,80),BR(80),SR(80),LBR(80),SK(12,12)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LCP(6),WD(8),D(12,12),W1(12,12)
NELT=NELT+1
LETY(NELT) = 3
LECP(1,NELT)=I
LECP(2,NELT)=J
LECP(3,NELT)=K
LECP(4,NELT)=L
TEGRD(1,NELT)=T
RETURN
END

SUBROUTINE PDMOD
COMMON/ORIG/NELT9NUNKT9IHBW9YM9SM9PR9NEL9NC9
19TGPC(2*80)9LETY(80)9LECP(6*80)9TEGRD(3*80)
29B0(3913)9BX(3913)9BY(3913)9NGPT
39BK(60980)9BR(80)9SR(80)9LBW(80)9SK(12912)
49TEP(3912)9T1(3912)9T2(3912)9E(393)9E1(393)
59LCP(6)9WD(8)9D(12912)9W1(12912)
COMMON/PI/HHEIGHT9DFL1(82)9DFL2(82)9TGPC1(82)
19TGPC2(82)9FAC9SF9FACS9SADIC9WIDX
23 DFL1(I)=SR(2*I-1)
23 DFL2(I)=SR(2*I)
CALL MXMN1(DFL1,NGPT982,DFMX,DXYL,DXR)
CALL MXMN1(DFL2,NGPT982,DY,Y,DYR)
IF(DY/-GT000)GO TO 45

Z1=-2
C++ SET MAX.DEFLECTION TO BE PLOT AT -2 IN.
SF= Z1/(ABS(DFMX) )
GO TO 47
45 Z1= SF*DFMX
CALL PAGE
CALL SLASH6
47 PRINT 48,DFMX,Z1,SR
48 FORMAT(* XXX SUB PDMOD XXX MAX DEFL IS*E15,7)
RETURN
END

SUBROUTINE PLOTDM(NP1,NP2,NP3,NP4,NP5,NP6,INP,IET)
C++ THIS IS PLOTDM2
C++ THIS SUB IS MODIFIED SO AS TO TAKE NP4,NP5,NP6 -
C++ ANY OF WHICH CAN BE ZERO
C++ INP=NODAL Pts OF ELE =3 FOR TRIM3,4,5,6)
C++ INP =4 IS FOR PARAM
C++ IET=ELE TYPE =3 FOR TRIM3,4,5,6 AND =4 FOR PARAM
COMMON/ORIG/NELT9NUNKT9IHBW9YM9SM9PR9NEL9NC9
19TGPC(2*80)9LETY(80)9LECP(6*80)9TEGRD(3*80)
29B0(3913)9BX(3913)9BY(3913)9NGPT
39BK(60980)9BR(80)9SR(80)9LBW(80)9SK(12912)
4, TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5, LCP(6), WD(8), D(12,12), W1(12,12)
COMMON/P1/ HEIGHT, DFL1(82), DFL2(82), TGPC1(82)
1, TGPC2(82), FAC, SF, FACS, SADIC, WIDX
C *** PLOT DEFLECTED MODEL FOR ANY ELEMENT
DIMENSION IBCD(1), IP(6)
DATA IBCD/4/
S=SF
DSF=S/FAC
IP(1)=NP1, IP(2)=NP2, IP(3)=NP3
IP(4)=NP4, IP(5)=NP5, IP(6)=NP6
GO TO (90, 90, 30, 40, INP)
30 DO 31 I=1,6
IP$=IP(I)
IF(IPS.EQ.0)GO TO 31
CALL SYMBOL(TGPC(1, IPS) * DSF * DFL1(IPS) * TGPC(5, IPS)
1 * DSF * DFL2(IPS), 0.05, IBCD(1), 0.00, 1)
31 CONTINUE
GO TO 50
40 DO 41 I=1,4
IP$=IP(I)
41 CALL SYMBOL(TGPC(1, IPS) * DSF * DFL1(IPS) * TGPC(5, IPS)
1 * DSF * DFL2(IPS), 0.05, IBCD(1), 0.00, 1)
50 CALL PLOT(TGPC(1, NP1) * DSF * DFL1(NP1) * TGPC(2, NP1)
1 * DSF * DFL2(NP1), 3)
C *** PLOT TRIM ELEMENT BOUNDERIES
DO 55 I=2, IET
IP$=IP(I)
55 CALL PLOT(TGPC(1, IPS) * DSF * DFL1(IPS) * TGPC(5, IPS)
1 * DSF * DFL2(IPS), 2)
CALL PLOT(TGPC(1, NP1) * DSF * DFL1(NP1) * TGPC(2, NP1)
1 * DSF * DFL2(NP1), 2)
90 RETURN
END
SUBROUTINE PLOTM2(NP1, NP2, NP3, NP4, NP5, NP6, INP, IET)
C *** THIS IS PLOTM2
C *** INP=NODAL PTS OF ELE = 3 FOR TRIM3, 4, 5, 6
C *** INP = 4 IS FOR PARAM
C *** IET=ELE TYPE = 3 FOR TRIM3, 4, 5, 6 AND = 4 FOR PARAM
COMMON/ORIG/NELT, NUNKT, IHBW, YM, SM, PR, NEL, NCD
1, TGPC(2, 80), LETY(80), LECF(6,80), TEGD(3,8A)
2, B0(3,13), BX(3,13), BY(3,13), NGPT
3, BK(60,80), BR(80), SR(80), LBW(80), SK(12,12)
4, TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5, LCP(6), WD(8), D(12,12), W1(12,12)
COMMON/P1/ HEIGHT, DFL1(82), DFL2(82), TGPC1(82)
1, TGPC2(82), FAC, SF, FACS, SADIC, WIDX
C *** PLOT MODEL FOR ANY ELEMENTS
C *** IF WANT TO CALL PLOTTING SUB, A CALL INITIAL MUST
C: *** BE CALLED FIRST
C: *** IF NP4 OR NP5 OR NP6 AREN'T NEEDED SIMPLY PUT 0
C: *** IN THEIR PLACES
C: *** HOWEVER, NP1, NP2, NP3 CAN'T BE 0 AS THEY
C: *** DEFINE THE TRIANGLE
D:\DIMENSION IBCD(1) IP(6)
DATA IBCD/2/
IP(1)=NP1$IP(2)=NP2$IP(3)=NP3
IP(4)=NP4$IP(5)=NP5$IP(6)=NP6
C: *** INP=3 IS TRIM, INP=4 IS PARAM
C: *** THE NEXT CALL CARD IS TEMPORARY
CALL SLASH4
GO TO (80, 80, 80, 80, 80, 80, 80, 80, 80, 80, 80)
30 IC=6
C: *** IC=COUNTER FOR TRIM 5, 4, 3
C: *** PLOT NP $ ITS NODAL NUMBER
DO 31 I=1, 6
IPS=IP(I)
IF(IPS.EQ.0)GO TO 35
X=TGPC(1, IPS) $ Y=TGPC(2, IPS)
C CALL NUMBER(X+.05, Y+.05, 0, 0.0, 0, 0, 0, 0)
XD=0.17/FAC $ YD=0.018/FAC
C: *** THIS WILL ENSURE REAL XD $ YD TO BE .17 $ .18
C: *** RESPECTIVELY.
CALL NUMBER(X=XD, Y=YD, 0.06, FLOAT(IPS), 0, 0, 0, 0)
CALL SYMBOL(TGPC(1, IPS), TGPC(2, IPS), 0, 0, 0, 0, 1)
11BCD(1) =0, 0, 0, 0, 0)
GO TO 31
35 IC=IC-1
31 CONTINUE
C: XB=( TGPC(1, NP1)+TGPC(1, NP2)+TGPC(1, NP3))/3.
C: YB=( TGPC(2, NP1)+TGPC(2, NP2)+TGPC(2, NP3))/3.
C CALL SYMBOL(XB=.033, YB=.034, 0.10, 54+IC, 0.0, 0, 1)
GO TO 50
40 IC=4
C: *** IC=4 IS THE COUNTER FOR PARAM
XB=0, $ YB=0.
C: *** PLOT NP $ ITS NODAL NUMBER
DO 41 I=1, 4
IPS=IP(I)
XB=XB+.25*TGPC(1, IPS) $ YB=YB+.25*TGPC(2, IPS)
X=TGPC(1, IPS) $ Y=TGPC(2, IPS)
CALL NUMBER(X+.15, Y+.05, 0, 0.0, 0, 0, 0, 0)
CALL SYMBOL(TGPC(1, IPS), TGPC(2, IPS), 0, 15, 0, 0, 0, 0, 0)
41 CONTINUE
C: *** PLOT TRIANGLE AND ITS CORRESPONDING NO.
C: *** (EITHER 3, 4, 5 OR 6)
HT=0.1
CALL PLOT(TGPC(1,NP1),TGPC(2,NP1),3)

DO 55 I=2,IET
IP$=IP(I)

CALL PLOT(TGPC(1,IPS),TGPC(2,IPS),2)
CALL PLOT(TGPC(1,NP1),TGPC(2,NP1),2)
90 RETURN

SUBROUTINE PMOD(XT,YT,X1,Y1,HTY1,XZ)

C *** FIND SIZE, PLOT AXIS. GIVES VALUES OF FAC ANH HEIGHT

COMMON/ORIG/NELT,NUNKT,IHW,YM,SMPR,NEL,NC
1,TGPC(2,80),LFYT(30),LECP(6,80),TEGD(3,85)
2,BX(3,13),BY(3,13),NGPT
3,BK(60,80),BR(80),SR(80),LBW(80),SK(12,12)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LCP(6),WD(8),(12,12),W1(12,12)
COMMON/P1/HEIGHT,DFL1(82),DFL2(82),TGPC1(82)
1,TGPC2(82),FACS,SAFAC,WIDX
COMMON/TRI/INPS(80),TSNP(3,80),TGPE(2,8)
1,SVON(80),ZZZ

C *** XO,YO ARE INPUT WHILE XI,Y1,HTY1,XZ ARE OUTPUTS

DIMENSION BCD(2)
DATA BCD/6HX AXIS,6HY AXIS/
X0=XT $ Y0=YT
CALL PLOT(X0,-11,-3)
CALL PLOT(Y0,-1,-3)
CALL MAXMIN(TGPC,NGPT,SADIC,XMAX,XMIN,WIDX,
1,YMAX,YMIN,HTY)
IHTX=WIDX
IHTY=HTY
CALL SCALE(TGPC,WIDX,NGPT,100)
CALL SCALE(TGPC,W1,HTY,NGPT,100)

C *** FAC CAN BE PRE-ASSIGNED THE VALUE 1.E. IN

C *** MAIN PROGRAM

C *** FAC,FACS,SAF ARE INITIALIZED IN CLEAR
IF(FAC<GT.0) GO TO 53
HTY1=ABS(HTY)
IF(HTY1<LT.1,0E-7) STOP 1
15 IF(HTY1<GT.10) GO TO 16
21 IF(HTY1<LT.1) GO TO 22
IF(HTY1<GE.4,0) GO TO 44
IF(HTY1<GE.2,0) GO TO 36
HTY=HTY1*5
44 FAC=HTY1/ABS(HTY)
GO TO 54
53 HTY1=FAC*ABS(HTY)
C 54 PRINT 55,FAC,HTY1,HTY
C 55 FORMAT(*FACTOR=*F10,4,*MODEL HEIGHT=*F10,4)
C 1* REAL HT=*F10,4)
CALL FACTOR(FAC)
GO TO 441
HTY1=HTY1/10.
GO TO 15
HTY1=HTY1*10.
GO TO 21
HTY1=HTY1*2.
GO TO 44
X0=0.0
Y0=0.0
IF (XMIN.LT. X0) X0= XMIN
IF (YMIN.LT. Y0) Y0= YMIN
IF (XMIN).NE.70 71=71
71 X1=X0
GO TO 93
IXMIN=XMIN
IF (XMIN< FLOAT(IXMIN)) 74 75 76
74 X1=FLOAT(IXMIN)=1.
GO TO 93
75 X1=XMIN
GO TO 93
76 X1=FLOAT(IXMIN)
93 IF (YMIN) 170 171 171
171 Y1=Y0
GO TO 193
170 IYMIN=YMIN
IF (YMIN< FLOAT(IYMIN)) 174 175 176
174 Y1=FLOAT(IYMIN)=1.
GO TO 193
175 Y1=YMIN
GO TO 193
176 Y1=FLOAT(IYMIN)
C 193 PRINT 195,X1,Y1
C 195 FORMAT(" X1,Y1=F2.0,4-")
193 CALL PLOT(-X1,-Y1)
C *** ORIGIN IS NOW MOVED TO(-X1,-Y1) FROM THE ORIGIN
C *** INITIALLY SET
C *** WIDX & HTY ARE REAL DIMENSION - NOT MODEL HT!
C *** THIS IS CORRECT AS WE HAVE CALLED FAC, THEREFORE
C *** THE DIMENSION PLOTTED WILL BE MODEL HEITH!
CALL AXIS(X1,0.0,BCD(1),-6,WIDX*0.0001,X1)
1TGPC1(NGPT+2),10.0)
CALL AXIS(0.0,Y1,BCD(2),6,HTY,90.0,Y1)
1TGPC2(NGPT+2),10.0)
XZ=X1+WIDX*5.
C *** XZ CAN BE USED AS NEW ORIGIN FOR NEXT PLOT
RETURN
SUBROUTINE PPAR (FLA,SE,D12)
COMMON/ORIG/NELT,NUNKT,IHBW,YM,SM,PR,NEL,NC5
C CHECK FOR PARALLELOGRAM
IF((ABS(SL14-SL)*ABS(SM14-SM)) GT 1.0E-6) GOTO 300
C=SL12*SL14*SM12*SM14
S = SQRT(1.0-C**2)
SE = C/S
I1 = LCP(1)
I4=LCP(4)
YP = (TGPC(2*I4)-TGPC(2*I1))*SL12-(TGPC(1*I4)-TGPC(1*I1))*SM12
FLA = YP/D12
C +++ FORM BASIC MATRICES
CALL CLEM (TEP,3,12,3,8)
CALL CLEM (T1,3,12,3,8)
CALL CLEM (T2,3,12,3,8)
TEP(1,1) = FLA
TEP(3,2) =-FLA
TEP(2,2) = (1.0-FLA*SE)
TEP(3,1) = (1.0-FLA*SE)
TEP(1,3) = FLA
TEP(3,4) = FLA
TEP(2,4) = (1.0+FLA*SE)
TEP(3,3) = (1.0+FLA*SE)
T1(2,2) = 1.0
T1(2,4) =-1.0
T1(3,1) = 1.0
T1(3,3) =-1.0
T2(1,1) = 1.0
T2(1,3) =-1.0
T2(3,2) = 1.0
T2(3,4) =-1.0
T2(2,2) =-2.0*SE
T2(2,4) = 2.0*SE
T2(3,1) =-2.0*SE
T2(3,3) = 2.0*SE
DO 301 I=1,3
DO 301 J=1,4
TEP(I,J+4)=-TEP(I,J)
GOTO 9999
T1(I,J+4) = T1(I,J)
301 T2(I,J+4) = T2(I,J)
CALL CLEM (D=12,12,12,12)
DO 302 I=1,7,2
D(I,I) = SL12
D(I,1) = SM12
D(I+1,I) = SM12
302 D(I+1,I+1) = SL12
RETURN
END

SUBROUTINE PRIBK
C *** PRINT BK (MASTER STIFFNESS)

COMMON/ORIG/NELT,NUNKT,IHBW,YM,SM,PR,NEL,NCD
1,TPGC(2,80),LETY(80),LECP(6,80),TEGD(3,8)
2,BO(3,13),BX(3,13),BY(3,13),NGPT
3,BK(6,80),BR(80),SR(80),LBW(80),SK(12,1)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LCP(6),WD(8),D(12,12),W1(12,12)
PRINT 2,NUNKT
2 FORMAT (///"** MASTER STIFFNESS MATRIX ***
1* TOTAL NUMBER OF UNKNOWNS **I6/**)
DO 4 I=1,NUNKT
DO 4 J=1,IHBW
J=J+1
IF (J.GT.NUNKT) GOTO 4
IF (BK(I,J).EQ.0.0) GOTO 4
PRINT 3,I,J,BK(I,J)
3 FORMAT (2X,21X,5X,E15.8)
4 CONTINUE
RETURN
END

SUBROUTINE PRILD
C ****+ PRINT LOADS

COMMON/ORIG/NELT,NUNKT,IHBW,YM,SM,PR,NEL,NCD
1,TPGC(2,80),LETY(80),LECP(6,80),TEGD(3,8)
2,BO(3,13),BX(3,13),BY(3,13),NGPT
3,BK(6,80),BR(80),SR(80),LBW(80),SK(12,1)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LCP(6),WD(8),D(12,12),W1(12,12)
PRINT 20
20 FORMAT (///"GRID POINT LOADS **)
PRINT 21,(I,8R(2*E1),8R(2*I);I=1,NGPT)
21 FORMAT (* PT, NO, *,I5,5X, U *,E20.8,5X)
RETURN
END

SUBROUTINE PSTLR
C *** THIS IS PSTR4
COMMON/ORG/NELT,NUNKT,IHBW,SYM,SM,PR,NEL,NC
1,TGPC(2,80),LETY(80),LECP(6,80),TEGD(3,89)
2,BO(3,13),BY(3,13),NGPT
3,BK(60,80),BR(60,80),SR(80),LBW(80),SK(12,12)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LCP(6),WD(8),D(12,12),W1(12,12)
COMMON/P1/HEIGHT,DFL1(82),DFL2(82),TGPE(2,8)
1,SVON(80),ZZZ
DIMENSION IBCD(1)
DATA IBCD/61 IN.

C *** PLOT STRESS (USING VON MISES CRITERION)
CALL MXMN1(SVON,NGPT,SADIC,SMAX,SMIN,SR)
IF (ABS(SMIN)<GT 0.8*SMAX) SMAX = SMIN

C *** SF * FAC, FACS ARE INITIALIZED IN SUB CLEAR
IF (FACS<6 To0 8) GO TO AS

C *** THE LARGEST ABSOLUTE VALUE OF
C *** THE ARRAY SVON
S = 1.5

C *** SET MAX, STRESS TO BE PLOTTED AT 1.5 IN.
FACS = S/(ABS(SMAX)*FAC)
GO TO 47

45 S = FACS*FAC*SMAX
47 PRINT 48,S,FACS,SMAX
48 FORMAT(13X*STRESS LENGTH,FACTOR,LARGEST STRESS=*
1(20X,E20.8)

C 48 FORMAT(13X*LENGTH OF STRESS USED IN PLOT,FACTOR,LARGEST STRESS=*E20.8)

441 ANG = ACOS(-1.0)/3.
DO 443 I=1,NGPT
X1 = TGPC(I,I) $ Y1 = TGPC(2,I)
RS = FACS*SVON(I)
RS = ABS(RS)
X2 = X1+RS*COS(ANG) $ Y2 = Y1+RS*SIN(ANG)
CALL PLOT(X1,Y1,3)
CALL PLOT(X2,Y2,2)

443 CONTINUE
RETURN
END

SUBROUTINE PTRIM (A)
COMMON/ORG/NELT,NUNKT,IHBW,SYM,SM,PR,NEL,NC
1,TGPC(2,80),LETY(80),LECP(6,80),TEGD(3,89)
2,BO(3,13),BY(3,13),NGPT
3,BK(60,80),BR(60,80),SR(80),LBW(80),SK(12,12)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5,LCP(6),WD(8),D(12,12),W1(12,12)
I1 = LECP(1,NEL)
I2=LECP(2,NEL)
I3=LECP(3,NEL)
CALL LINF (I2,I3,TEP(2,2),TEP(1,1),D23,SL23,SM23)
CALL LINF (I3,I1,TEP(2,4),TEP(1,3),D31,SL31,SM31)
CALL LINF (I1,I2,TEP(2,6),TEP(1,5),D12,SL12,SM12)
DO 200 J=1,5,2
TEP(1,J)=TEP(1,J)
TEP(2,J)=0.
TEP(3,J)=TEP(2,J+1)
200 TEP(3,J+1)=TEP(1,J)
HP = 5*(D12*D23*D31)
A = SQRT(HP*(HP=D12)*(HP=D23)*(HP=D31))
RETURN
END

SUBROUTINE PTRIMH ( A,ZYXY,ZIXX,ZIYY,MS)
C +++ TGPC(1,1) DEFINED IN SUB GENPT
C +++ LECP(1,NEL) DEFINED IN SUB TRIM6
C +++ MS=4,5,6 FOR TRIM4,5,6
C +++ CALLED BY SKE
C +++ TGPE = TABLE OF GRID PT. COORD) EXTRA ARRAY.
C +++ IN GLOBAL COORD.
COMMON/ORIG/NEL,TNKET,HWV,YM,SM,PR,NEL,NC
1. TGPC(2,80),LEY(80),LECP(6,80),TEGD(3,8)
2. B0(3,13),BX(3,13),BY(3,13),NGPT
3. BR(60,80),SR(80),LBW(80),SK(12,15)
4. TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
5. LCP(6),WD(8),O(12,12),W1(12,12)
COMMON/TRI/INPS(80),TSNP(3,80),TGPE(2,80)
1SVON(80),ZZZ
DIMENSION AA(6,6),A1(6,6),I(6)
DO 7 M=1,MS
7 I(M)=LECP(M,NEL)
I1=I(1) I2=I(2) I3=I(3)
X0=(TGPC(1,1)+TGPC(1,2)+TGPC(1,3))/3.
Y0=(TGPC(2,1)+TGPC(2,2)+TGPC(2,3))/3.
C +++ TGPE(1,M) TGPE(2,M) ARE TEMPORARITY IN
C +++ GLOBAL SYSTEM NOW.
DO 10 M=1,MS
IM=I(M)
TGPE(1,M)=TGPC(1,IM) TGPE(2,M)=TGPC(2,IM)
10 CONTINUE
GO TO (99,99,99,14,15,16),MS
14 I2=I(2) I3=I(3)
TGPE(1,5)=(TGPC(1,I2)+TGPC(1,13))/2.
TGPE(2,5)=(TGPC(2,I2)+TGPC(2,13))/2.
15 I1=I(1) I3=I(3)
TGPE(1,6)=(TGPC(1,13)+TGPC(1,I1))/2.
TGPE(2,6)=(TGPC(2,13)+TGPC(2,I1))/2.
16 CALL CLEM (AA,6,6,6,6,6)
C  +++ TGPE(1,M) $ TGPE(2,M) BECOME LOCAL X, S $ Y, S
C  +++ IN DO 53
   DO 53 M=1,6
      AA(M,2)=TGPE(1,M) =X0
      TGPE(1,M)=AA(M,2)
      AA(M,3)=TGPE(2,M) =Y0
      TGPE(2,M)=AA(M,3)
      AA(M,4)=AA(M,2)*+2
      AA(M,5)=AA(M,2)*AA(M,3)
   53   AA(M,6)=AA(M,3)*+2.
   DO 55 M=1,6
   55   AA(M,1)=1.
C  PRINT 123, ((AA(I,J) , J=1,6) , I=1,6)
C  123 FORMAT(* SUB PRIMHA (AA(I,J) 6X6= 6E18.8)
C  DO 503 M=1,6
C  DO 503 J=1,6
C 503 A2(M,J)=AA(M,J)
   CALL CLEM (A1,6,6,6,6,6,6)
   CALL MINV (AA,6,6)
C  +++ A1 IS THE INVERSE OF AA
C  +++ CHECK WHETHER AA*A1 = I
C  CALL MMM (A2,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6,6)
   CALL CLEM(B0,3,13,3,13)
   CALL CLEM(BX,3,13,3,13)
   CALL CLEM(BY,3,13,3,13)
C  +++ CONSTRUCT THE 8 MATRICES
   DO 80 M=1,6
      II=1+2*(M-1)
      B0(1,II )=A1(2,M)
      B0(1,II+1)=0.
      B0(2,II )=0.
      B0(2,II+1)=A1(3,M)
      B0(3,II )=A1(3,M)
      B0(3,II+1)=A1(2,M)
   80   CONTINUE
   DO 81 M=1,6
      II=1+2*(M-1)
      BX(1,II)=2.*A1(4,M)
      BX(1,II+1)=0.
      BX(2,II)=0.
      BX(2,II+1)=A1(5,M)
      BX(3,II)=A1(5,M)
      BX(3,II+1)=2.*A1(4,M)
   81   CONTINUE
   DO 82 M=1,6
      II=1+2*(M-1)
      BY(1,II )=2.*A1(6,M)
      BY(1,II+1)=0.
      BY(2,II )=0.
BY(2:I+1)=A1(5:M)
BY(3:I )=A1(5:M)
BY(3:I+1)=2*A1(5:M)
82 CONTINUE
CALL LINF (I2,I3,XD23,YD23,D23,SL23,SM23)
CALL LINF (I1,I2,XD31,YD31,D31,SL31,SM31)
CALL LINF (I1,I2,XD12,YD12,D12,SL12,SM12)
HP=5*(D12*D23+D31)
A=SQRT(HP*(HP-D12)*(HP-D23)*(HP-D31))
ZIYY=A*(TGPE(1,1)*2+TGPE(1,2)*2+TGPE(1,3)*2)/12
ZIXX=A*(TGPE(2,1)*2+TGPE(2,2)*2+TGPE(2,3)*2)/12
ZIXY=A*(TGPE(1,1)*2+TGPE(2,1)*2+TGPE(1,2)*2+TGPE(2,2)*
1TGPE(1,3)*TGPE(2,3))/12
99 RETURN
END

SUBROUTINE SMM (S,A,MA,NA,M,N,B)
DIMENSION A(MA,NA),B(MA,NA)
DO 1 I=1,M
DO 1 J=1,N
1 B(I,J) = S*A(I,J)
RETURN
END

SUBROUTINE SKE
COMMON/ORIG/NELT,NUNKT,IHBW,YM,SM,PR,NEL,NCP
1,TGPE(2,80),LETY(80),LECP(6,80),TEGD(3,85)
2,B0(3,13),Bx(3,13),BY(3,13),NGPT
3,BK(60,80),BR(80),SR(80),LBW(80),SK(12,12)
4,TEP(3,12),T1(3,12),T2(3,12),E(3,3),L(3,3)
5,LCP(6),WD(8),D(12,12),W1(12,12)
COMMON/TRI/INPS(80),TSNP(3,80),TGPE(2,8)
1SVON(80),ZZZ
DIMENSION LNCP(8)
DATA LNCP/2,3,4,4,5,6,0,0/
IT=LETY(NEL)
NCP=LNCP(IT)
DO 50 I=1,NCP
50 LCP(I)=LECP(I,NEL)
GO TO (1,2,3,6,6,6,6)
1 CALL PFLA(D)
C=YM*TEGD(1,NEL)/D
DO 100 I=1,4
100 J=1,4
100 SK(I,J) = C*WD(I)*WD(J)
RETURN
2 CALL PTRIM(A)
CALL CLEM(SK,12,12,12,12)
CALL CONG(TEP,3,12,3,6,E,3,3,SK,12,12)
T=TEGD(1,NEL)
C = T*25/A
CALL SMM (C, SK, 12, 12, 8, 8, SK)
RETURN

3 CALL PPAR (FLA, SE, D12)
T = TEGD (1, NEL)
CT = 0.25*T/FLA
CALL SMM (CT, E, 3, 3, 3, 3, E1)
CALL CLEM (W1, 8, 8, 8, 8)
CALL CONG (TEP, 3, 8, 3, 8, E1, 3, 3, W1, 8, 8)
CT = (1+FLA*E*2*SE**2)*T/(12*FLA)
CALL SMM (CT, E, 3, 3, 3, 3, E1)
CALL CONG (T1, 3, 8, 3, 8, E1, 3, 3, W1, 8, 8)
CT = FLA*T/1200
CALL SMM (CT, E, 3, 3, 3, 3, E1)
CALL CONG (T2, 3, 8, 3, 8, E1, 3, 3, W1, 8, 8)
CT = CT*SE
CALL SMM (CT*E, 3, 3, 3, 3, E1)
CALL CONG (TEP, 3, 8, 3, 8, E1, 3, 3, W1, 8, 8)
RETURN

4 DO 43 I=1,12
C DO 43 IS POST MULTIPLYING SK (12X12) BY
C TRANSFORMATION MX (12X8), RESULT = SK (12X8)
SK (I, 3) = SK (I, 3) + 5*SK (I, 9)
SK (I, 5) = SK (I, 5) + 5*SK (I, 9)
SK (I, 1) = SK (I, 1) + 5*SK (I, 11)
SK (I, 2) = SK (I, 2) + 5*SK (I, 12)
SK (I, 6) = SK (I, 6) + 5*SK (I, 12)

43 SK (I, 3) = SK (I, 3) + 5*SK (I, 9)
SK (I, 5) = SK (I, 5) + 5*SK (I, 9)
SK (I, 1) = SK (I, 1) + 5*SK (I, 11)
SK (I, 2) = SK (I, 2) + 5*SK (I, 12)
SK (I, 6) = SK (I, 6) + 5*SK (I, 12)
RETURN

5 DO 53 I=1,12
C DO 53 IS POST MULTIPLYING SK (12X12T) BY
C TRANSFORMATION MX (12X10), RESULT = SK (12X10)
SK (I, 1) = SK (I, 1) + 5*SK (I, 11)
SK (I, 5) = SK (I, 5) + 5*SK (I, 11)
RETURN
SK(I,2) = SK(I,2) + 5*SK(I,12)
53 SK(I,6) = SK(I,6) + 5*SK(I,12)
C *** DO 55 IS PRE-MULTIPLYING SK(I,12X10) BY
C *** TRANSFORMATION MX, TRANSPOSED, RESULT = SK(I,10X10)
DO 55 I = 1, 10
SK(1, I) = SK(1, I) + 5*SK(11, I)
SK(5, I) = SK(5, I) + 5*SK(11, I)
SK(2, I) = SK(2, I) + 5*SK(12, I)
55 SK(6, I) = SK(6, I) + 5*SK(12, I)
RETURN
6 CALL PTRIMH( A\, ZIYY, ZIXX, ZIXY, NCP)
C *** FORM E1 MATRIX BY MULTIPLYING E BY THICKNESS T
C *** THAT IS TO OBTAIN E(1,1) * T, E(1,2) * T, E(3,3) * T
T = TEGD(1, NEL)
YG = E(1,1) * T
P = E(1,2) * T
S = E(3,3) * T
CALL CLEM (SK, 12, 12, 12, 12)
CALL ATEB(A\, B0, B0, Y0, P, S, SK, 12)
CALL ATEB(ZIYY, BX, BX, YG, P, S, SK, 12)
CALL ATEB(ZIIX, BY, BY, YG, P, S, SK, 12)
CALL ATEB(ZIXY, BX, BX, YG, P, S, SK, 12)
CALL ATEB(ZIXY, BY, BY, YG, P, S, SK, 12)
C CALL CHEKS(K(SK, 12, 12)
IF (NCP EQ 6) RETURN
IF (NCP EQ 5) G0 TO 5
IF (NCP EQ 4) G0 TO 4
RETURN
END
SUBROUTINE STIFF
C *** COMPUTE AND ASSEMBLE ELEMENT STIFFNESSES
COMMON/ORIG/NELT, NUNKT, IHBW, YM, SM, PR, NCP, NEL, NCP:
1, T6PC (2, 80) ) , LETY (80) ) , LECF (6, 80) ) ) TEGD (3, 80)
2 \, B0 (3, 13) \, BX (3, 13) \, BY (3, 13) \, NGPT
3 \, BK (60, 80) \, RR (80) \, SR (80) \, LBW (80) \, SK (12, 12)
4 \, TEP (3, 12) \, T1 (3, 12) \, T2 (3, 12) \, E (3, 3) \, E1 (3, 3)
5 \, LCP (6) \, WD (8) \, D (12, 12) \, W1 (12, 12)
C *** MASTER STIFFNESS INITIALIZATION
C PRINT 7, NUNKT, IHBW
C 7 FORMAT ( //, * NO. OF UNKNOWNS * \, I4, *
C 1 * HALF BAND WIDTH * \, I4)
DO 5 I = 1, NUNKT
DO 5 J = 1, IHBW
5 BK(I, J) = 0.0
DO 1 NEL = 1, NELT
C *** ASSEMBLE ELEMENT NEL
CALL SKE
DO 10 I = 1, NCP
IB = 2*LCP(I)-1
IS=2*I=1
DO 10 J=1,NCP
  JB = 2*LCP(J)=1
  JS=2*J=1
DO 11 K=1,2
  KK=IB*K=1
DO 12 L=1,2
  LL=JB+IB+L*K=1
IF (LL .LE. 0) GOTO 12
  BK(KK,LL) = BK(KK,LL)+SK(IS+K=1,JS=L=1)
12 CONTINUE
11 CONTINUE
10 CONTINUE
RETURN
END

SUBROUTINE STRE
C "CALCULATE STRESSES IN ELEMENT NEL"
COMMON/ORIG/NELT,NUNKT,IHBW,YM,SM,PR,NEL,NCP
  TGPC(2,80),LETY(80),LECP(6,80),TEGD(3,8)
  B0(3,13),BY(3,13),NGPT
  BK(60,80),BR(80),SK(80),LW(80),SK(12,12),
  TEP(3,12),T1(3,12),T2(3,12),E(3,3),E1(3,3)
  LCP(6),WD(8),D(12,12),W1(12,12)
COMMON/TRI/INPS(80),TSNP(3,80),TGPE(2,8)
1SVON(80),Z
DIMENSION LNCP(8)
DATA LNCP/2,3,4,4,5,6,0,0/
IT = LETY(NEL)
NCP = LNCP(IT)
DO 50 I=1,NCP
  LCP(I) = LECP(I,NEL)
DO 54 I=1,NCP
  J=LCP(I)
  SK(2*I-1,J) = SR(2*J-1)
54 SK(2*I,J) = SR(2*J)
GO TO (1,2,3,4,5,6),IT
1 CALL PFLA(D)
  A = TEGD(1,NEL)
  DL=0.0
DO 100 I=1,4
  DL=DL+WD(I)*SK(I,1)
  S =DL *YM/D
PRINT 101*I1,I2,S
101 FORMAT (*STRESS IN ROD CONNECTING NODES *,E3,0,
  * AND *,I3,*, IS *, E20,8)
RETURN
2 CALL PTRIM(A)
  T = TEGD(1,NEL)
DO 202 K = 1,3
WD(K) = 0.0
DO 203 I = 1,3
DO 203 J = 1,6
203 WD(K) = WD(K) * E(K, I) * TEP(I, J) * SK(J, I)
202 WD(K) = WD(K) * 5 / A
C DO 225 ICP(I), I = 1,3, (WD(I), I = 1,3)
C 225 FORMAT (5X, 3I5, 3E20.8)
C DO 205 PRINT (LCP(I), I = 1,3), (WD(I), I = 1,3)
C *** TSNP = STRESS AT NP FOR TRIM3, 4, 5, 6 = VALUE <
C *** ARE ACCUMULATED.
C *** INPS IS A COUNTER FOR CORRESPONDING NP STRESSES
C 352 FORMAT (* LOCAL PT, I*, 5X, GLOBAL PT, J*, 5X
C 1*NO OF ELE, INPS(J)*20X, TSNP(M, I), M = 1,3, *)
C DO 354, I = 1, NCP
J = LCP(I)
INPS(J) = INPS(J) + 1
DO 354 M = 1, 3
TSNP(M, J) = TSNP(M, J) * WD(M)
354 CONTINUE
RETURN
3 CALL PPAR (FALSE, DI2)
CALL MMM (D, 8, 8, 8, 8, B, SK, B, 8, 8, 1, W, 8, 8)
X8 = 0.
Y8 = 0.
D12 = D12 * 5
DO 902 K = 1, 4
L = LCP(K)
X8 = X8 + 25 * TGPC(1, L)
Y8 = Y8 + 25 * TGPC(2, L)
902 CONTINUE
DO 903 K = 1, 4
L = LCP(K)
XA = TGPC(1, L) - XR
YA = TGPC(2, L) - YB
DO 904 I = 1, 3
DO 904 J = 1, 8
904 SK(I, J) = TEP(I, J) * D12 * XA * T1(I, J) + YA * T2(I, J)
903 CALL MMM (SK, 8, 8, 3, 8, W, 8, 8, 1, D(1, K), 8, 1)
DO 906 I = 1, 3
DO 906 J = 1, 8
SK(I, J) = TEP(I, J) * D12
906 CONTINUE
CALL MMM (SK, 8, 8, 3, 8, W, 8, 8, 1, D(1, 5), 8, 1)
CALL MMM (E, 3, 3, 3, 3, D, 8, 8, 5, SK, 8, 8)
CT = 0.25 / (D12 * 2 * FLA)
CALL SMM (CT, SK, 8, 8, 3, 5, SK)
PRINT 307, (LCP(I), I = 1, 4)
307 FORMAT (/I6,3I20)  
PRINT 305, ((SK(I,J), J=1,N), I=1,3)  
305 FORMAT (* SXL *,5E20.8/* SYL *,5E20.8/* 
1* TXYL *,5E20.8)  
RETURN  
4 SK(9,1)= (SK(3,1)*SK(5,1))/2.  
SK(10,1)= (SK(4,1)*SK(6,1))/2.  
5 SK(11,1)= (SK(1,1)*SK(5,1))/2.  
SK(12,1)= (SK(2,1)*SK(6,1))/2.  
6 CALL PTRIMH( A, ZIYY, ZIXX, ZIXY, NCP)  
C TO FIND TEP(3X12) FOR THE 6 NODAL PTS.  
C *** KK=1,6 IS FOR THE 6 NPS.  
DO 603 KK= 1,NCP  
CALL' CLEM (TEP, 3,12,3,12)  
DO 604 I = 1,3  
DO 604 J = 1,12  
604 TEP(I,J) = B0(I,J) * BX(I,J) * TGPE(I,KK) * BY(J,J) *  
1 TGPE(2,KK)  
C TO FIND TEP(3X12) FOR POINT AT CENTROID OF TRIANGLE IS  
C *** SIMPLY 90 AS THE CENTROID IS USED AS THE ORIGIN  
C *** FOR THE LOCAL COORDINATE, (THEREFORE X,Y = 0)  
C *** STRESSES FOR THE 6 NODAL PINNALS  
DO 607 K=1,3  
W1(K,KK)=0.0  
606 DO 607 I=1,3  
DO 607 J = 1,12  
607 W1(K,KK) = E(K,I) * TEP(I,J) * SK(J,1) * W1(K,KK)  
603 CONTINUE  
C STRESSES OF CENTROID  
DO 608 K=1,3  
W1(K,7)=0.0  
DO 608 I = 1,3  
DO 608 J = 1,12  
608 W1(K,7) = E(K,I) * B0(I,J) * SK(J,1) * W1(K,7)  
C PRINT 615,NCP, (LCP(I), I=1,NCP)  
C 615 FORMAT (/I6,NCP, *PTS* 1I0,5I18)  
C PRINT 616, NCP  
C 616 FORMAT (I1, 119X *CENTER OF TRIM*, I1/))  
C PRINT 617, (W1(1,J), J=1,NCP)  
C 617 FORMAT (* SXL *, 6E18.6)  
C PRINT 627, W1(1,7)  
C PRINT 618, (W1(2,J), J=1,NCP)  
C 618 FORMAT (* SYL *, 6E18.6)  
C PRINT 627, W1(2,7)  
C PRINT 619, (W1(3,J), J=1,NCP)  
C 619 FORMAT (* TXYL *, 6E18.6)
PRINT 6279 W1(3,7)
C 627 FORMAT('** 114X E10.6')
C *** TSNP = STRESS AT NP FOR TRIM3, 4, 5, 6 = VALUEs ARE ACCUMULATED.
C *** INPS IS A COUNTER FOR CORRESPONDING NP STRESSES
DO 654 KK=1 NCP
   J=LCP(KK)
   INPS(J) = INPS(J) + 1
D0654K
654 TSNP(K,J) = TSNP(K,J) - W1(K,KK)
RETURN,
END
SUBROUTINE STRESS
COMMON/ORIG/NELT, NUNKT, IHBW, YM, SM, PR, NEL, NCP
1, TGP(C,80), LETY(80), LEC(P(6,80), TEGD(3,80)
2, B0(3,13), BX(3,13), BY(3,13), NGPT
3, BK(60,80), BR(80), SR(80), LBW(80), SK(12,13)
4, TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5, LCP(6), WD(8), D(12,12), W1(12,12)
COMMON/TRI/INPS(80), TSNP(3,80), TGP(2,8)
15VON(80), ZZZ
C *** COMPUTE ELEMENT STRESSES
DO 1 NEL=1, NELT
   CALL STRE
1 CONTINUE
99 RETURN
END
SUBROUTINE TRIM3 (I, J, K, T)
C *** DEFINES MEMBRANE TRIANGLE I, J, K THICKNESS T
COMMON/ORIG/NELT, NUNKT, IHBW, YM, SM, PR, NEL, NCP
1, TGP(C,80), LETY(80), LEC(P(6,80), TEGD(3,80)
2, B0(3,13), BX(3,13), BY(3,13), NGPT
3, BK(60,80), BR(80), SR(80), LBW(80), SK(12,13)
4, TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5, LCP(6), WD(8), D(12,12), W1(12,12)
NELT = NELT+1
LETY(NELT) = 2
LECP(1, NELT) = I
LECP(2, NELT) = J
LECP(3, NELT) = K
TEGD(1, NELT) = T
RETURN
END
SUBROUTINE TRIMH (I1, I2, I3, I4, I5, I6, MS, T)
C *** FOR TRIM4 OR TRIM5 PUT I4 OR I5 OR I6 =0 FOR .
C *** PT(S) UNDEFINED.
C *** MS=4,5,6 FOR TRIM4, 5, 6

C  +++++ DEFINE MEMBRANE TRIANGLE I1, I2, I3 WITH 3 EXTRA
C  +++++ NODAL POINTS.
C  +++++ I4, I5, I6 ALONG LINES I1=I2, I2=I3 AND I3=I1
C  +++++ THICKNESS IS T.
C  +++++ LETY(NELT) = 1 IS ROD, =2 IS TRIM3, =3 IS PAGAM
C  +++++ LETY(NELT) = 4 IS TRIM4, =5 IS TRIM5, =6 IS TRIM6
COMMON/ORIG/NELT, NUNKT, IHBW, YM, SM, PR, NEL, NCP
1* TGPC(2,80), LETY(80), LECP(6,80), TEGD(3,8)
2* B0(3,13), BX(3,13), BY(3,13), NGPT
3* BK(60,80), BR(80), SR(80), LBW(80), SK(12,12)
4* TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5* LCP(6), WD(8), D(12,12), W1(12,12)
NELT = NELT+1
LETY(NELT) = MS
LECP(1, NELT) = I1
LECP(2, NELT) = I2
LECP(3, NELT) = I3
LECP(4, NELT) = I4
LECP(5, NELT) = I5
LECP(6, NELT) = I6
TEGD(1, NELT) = T
RETURN
END
SUBROUTINE UNLOAD
C  +++++ INITIALIZE LOAD MATRICE
COMMON/ORIG/NELT, NUNKT, IHBW, YM, SM, PR, NEL, NCP
1* TGPC(2,80), LETY(80), LECP(6,80), TEGD(3,8)
2* B0(3,13), BX(3,13), BY(3,13), NGPT
3* BK(60,80), BR(80), SR(80), LBW(80), SK(12,12)
4* TEP(3,12), T1(3,12), T2(3,12), E(3,3), E1(3,3)
5* LCP(6), WD(8), D(12,12), W1(12,12)
DO 1 I=1, NUNKT
1 BR(I) = 0.0
RETURN
END
APPENDIX F

INTRODUCTION OF INTERMEDIATE BOUNDARY NODES

A general scheme for automatic renumbering of nodes and modification of element types to correspond to the addition of a new node to side i-j during a dynamic modelling analysis is presented in the flow chart in Figure F.1.

In the following example, a sequence of figures resulting from the introduction of new nodes, one at a time, and the subsequent automatic renumbering of nodes are shown (see Fig. F.2). A triangle (Δ) represents the introduction of a new node to the existing model. The number of the new node (NNN) is shown in the next subsequent figure. NNN is computed from the formula $NNN = \frac{(i+j+1)}{2}$ where i and j define the side (or interface) to which the new node is added. The fractional part of NNN is discarded and only the integral part is taken as the number of the new node. Accompanying each figure is the value B, indicating the maximum node number difference occurring in an element in the model. The bandwidth of a model is dependent on the value B.
Figure F.1 Flow Chart for Automatic Node Renumbering and Element-type Modification Subsequent to the Introduction of Intermediate Boundary Nodes
Element number NEL has side i,j?

YES

If node number > NNN increase by 1

Add point NNN to the element

Modify element type

Adjust the order of the node numbers defining the element if necessary

NO

NEL > total number of element?

YES

Increase the total number of nodes by 1

NEL = NEL + 1

If node number > NNN increase by 1

Figure F.1 (Cont'd)
Figure F.2 Introduction of Intermediate Boundary Nodes
REFERENCES


144

