View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by NASA Technical Reports Server

NASA Contractor Report 189050

A Novel Approach in Formulation of Special Transition Elements: Mesh Interface Elements

Nesrin Sarigul
Ohio State University
Columbus, Ohio

November 1991

Prepared for
Lewis Research Center
Under Grant NAG3-790

National Aeronautics and
Space Administration

(NASA-CR-189050) A NOVEL APPROACH IN FORMULATION OF SPECIAL TRANSITION ELEMENTS: MESH INTERFACE ELEMENTS Final Report, Jan. 1989 (Ohio State Univ.) 64 p CSCL 20K
N92-20954
Unclas
G3/39 0079767

TABLE OF CONTENTS

PREFACE

This report describes the status of work being performed on a Novel Approach in Formulation of Special Transition Elements research, (Grant NAG 3-790). The research is being monitored by Dr. C.C. Chamis of the NASA Lewis Research Center. The graduate research assistants for this term were Mr. O. Odabas and Mr. M. Yahiaoui. Mr. Odabas and Mr. Yahiaoui, for whom I am the dissertation advisor, are both Ph.D. degree students in the Aeronautical and Astronautical Engineering Department.

Nesrin Sarigul

I. INTRODUCTION

The objective of this research program is in the development of more accurate and efficient advanced methods for solution of singular problems encountered in various branches of mechanics. The research program can be categorized under three levels. First two levels involve with the formulation of new class of elements called "Mesh Interface Elements" (MIE) to connect meshes of traditional elements either in three dimensions or in three and two dimensions. The finite element formulations are based on the boolean sum and blending operators. This report describes the results obtained from the first two levels of the program. It may be noted that, at present, the second level of the program is being conducted under NAG 3-790.

In today's advanced aircraft and space structure applications, steep temperature and/or stress gradients are commonly encountered. The analysis methods need to incorporate these steep gradients into the solution efficiently and accurately. Mesh Interface Elements are being formulated and tested in this research to account for the steep gradient effects. At present, the heat transfer and structural analysis problems are being formulated from uncoupled theory point of view.

The status report, first, summarizes the general formulation for heat transfer and structural analysis by including the newly introduced varying material properties at material nodal points of the elements concept. Then the formulation of mesh interface elements are detailed. On the computational efficiency side, a hidden-symbolic computation concept developed by the author is given. Verification examples are included from heat transfer and structural analysis problems. Appendix includes listings of the computer modules that are developed for this purpose.

II. FINITE ELEMENT FORMULATION FOR HEAT TRANSFER AND STRUCTURAL ANALYSIS

Thermal effects induced by aerodynamic heating on advanced aircraft and spacecraft systems of current technology requires special analysis procedures in order to design these structural components to fulfill the specific mission requirements. The steep temperature and/or stress gradients and unusual advanced geometry and material concepts are being major items to deal with. In this research, the steep gradients problems are tackled by utilizing the Mesh Interface Elements and variable material properties at material nodal points elements concept.

II.a. Heat Transfer Analysis

The heat conduction equation in three dimension is

$$\rho c_p \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left[k_{xx} \frac{\partial T}{\partial x} + k_{xy} \frac{\partial T}{\partial y} + k_{xz} \frac{\partial T}{\partial z}\right] + \frac{\partial}{\partial y} \left[k_{yx} \frac{\partial}{\partial x} + k_{yy} \frac{\partial T}{\partial y} + k_{yz} \frac{\partial T}{\partial z}\right]$$

$$+ \frac{\partial}{\partial z} \left[k_{zx} \frac{\partial T}{\partial x} + k_{zy} \frac{\partial T}{\partial y} + k_{22} \frac{\partial T}{\partial z}\right] + f$$

where ρ is the mass density of the material, c_p is the heat capacity, T is the absolute temperature, $k_{xx}, \ldots k_{22}$ are the heat conduction coefficients and f is the heat source term.

The equation will be solved subjected to the boundary conditions in terms of prescribed either the temperature and/or temperature gradients.

For homogeneous anisotropic solids the equation becomes

$$\rho c_p \frac{\partial T}{\partial t} = k_{xx} \frac{\partial^2 T}{\partial x^2} + k_{yy} \frac{\partial^2 T}{\partial y^2} + k_{22} \frac{\partial^2 T}{\partial z^2} + (k_{xy} + k_{yx}) \frac{\partial^2 T}{\partial x \partial y} + (k_{yz} + k_{zy}) \frac{\partial^2 T}{\partial y \partial z}$$

$$+ (k_{xy} + k_{zx} \frac{\partial^2 T}{\partial z \partial x}) + f$$

in case of isotropic material we have

$$\rho c_p \dot{T} - k \left(\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2}\right) + f$$

for the steady-state conditions, we obtain the Poisson's equation as

$$k \;\; {}^2 T + f = 0$$

where

$$\nabla^2 = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)$$

By utilizing the method of weighted residual on the heat conduction equation, we obtain

$$- \int_v \rho c_p W \frac{\partial \bar{T}}{\partial T} \, dV + \int_v Wk(\;\; . \;\; \bar{T}) \, dV + \int_v W f dV = 0$$

where W is the weighting function and T is the approximated temperature field and V is the volume of the structure. By utilizing the Green-Gauss theorem on the first term of the equation above, we obtain

$$\int_v K. \;\; W. \;\; T \, dV = \int_v W. f dV + \int_s n. k. \;\; T dS$$

where S is the surface of the domain.

The discretized equations for NE number of subdomains may be given as

$$\sum_{e=1}^{NE} \int_{V_e} k. \;\; W. \;\; T dV_e = \sum_{e=1}^{NE} \int_{V_e} W. f. dV_e + \sum_{e=1}^{M} \int_{S_e} n. \;\; \dot{T} dS$$

where V_e and S_e are the element volume and surface mean, respectively.

By utilizing the shape functions used in approximating the temperature field to be the weighting functions, we achieve Galerlin type formulation. At an element level the equations are

$$[K_c]^e\{T\}^e = \{Q\}^e + \{Q_s\}^e$$

where $[K_c]^e$ is the finite element conduction matrix, $\{Q\}^e$ is the finite element load vector and $\{Q_s\}^e$ is the heat source vector.

It may be noted that material properties are assumed known at material nodal points within an element as it is detailed in the following section.

The present numerical applications deal with the steady state equations and isotropic material behavior for the time being.

II.b. Structural Analysis

The principle of virtual work will be employed,

$$\int_V (\delta\epsilon)^T \sigma dV = \int_V (\delta\Delta)^T b dV + \int_s (\delta\Delta)^T s ds + \sum_{i=1}^{N} (\delta\Delta)^T P_i$$

where $(\delta\epsilon)^T$ is the virtual strain vector, $(\delta\Delta)^T$ is the virtual displacement vector, b is the body force vector and s is the prescribed surface tractions vector and P_i indicate the point loads.

For an anisotropic material by utilizing the generalized Hooke's law the equation above can be written as

$$\int_V (\delta\epsilon)^T.A.\epsilon\, dV = \int_V (\delta\epsilon)^T.A.\epsilon._{init} dV - \int_V (\delta\epsilon)^T \sigma_{init} dV + \int_V (\delta\Delta)^T.b dV + \int_s (\delta\Delta)^T.s.dS$$
$$+ \sum_{i=1}^{N} (\delta\Delta)^T P_i$$

However, by discretizing the region to be analyzed into an NE number of finite elements, we obtain for one element (e)

$$\int_{V_e} (\delta\epsilon)^T.A.\epsilon dV = \int_{V_e} (\delta\epsilon)^T.A.\ \epsilon_{init} dV - \int_{V_e} (\delta\epsilon)^T \sigma_{init} dV + \int_{V_e} (\delta\Delta)^T.b.dV + \int_{S_e} (\delta\Delta)^T.s.dS$$

$$+ \left(\sum_{i=1}^{N} (\delta\Delta)^T P_i\right)^e$$

where V_e and S_e denote the volume and the boundary surface of the element.

The field variable Δ, displacement field, will be approximated in terms of the nodal displacements times the shape functions. In addition, the material properties are assumed to be known at "material nodal points," NM as introduced in this research, hence, we have

$$A = \sum_{i=1}^{NM} NMAT_i . A_i$$

where $NMAT_i$ are the "material shape functions." It may be noted that it is possible to use material shape functions to be the same as the field variable shape functions. This approach, newly introduced here, will increase solution efficiency for problems involving large gradients.

A summary of the equations takes the form of

$$M.\bar{\Delta} + K.\Delta = P$$

It may be noted that unlike the traditional formulations, in calculation of the stiffness and mass matrix material properties are considered to be known at material nodal points. Therefore, the stiffness matrix, for example, will be calculated as

$$K^e = \int_{vc} B^T \left(\sum_{i=1}^{NM} NMAT_i . A_i\right) . B \; dV$$

Application problems include three-dimensional mesh interface elements under different loading conditions. The results obtained from test examples are reported in numerical examples section.

III. FORMULATION OF MESH INTERFACE ELEMENTS

III.a. General Concepts on Element Formulation

Formulation of the "Mesh Interface Elements" (MIE) are based on the boolean sum. In dealing with analysis aspects in three dimensions, it is known that the higher order elements are too costly to use throughout the domain. It may be noted that, we need these higher order elements only at certain local areas of the domain. The rest of the domain can be modeled by utilizing lower order elements. In this research, a series of Mesh Interface Elements are being developed to connect meshes of different types of elements.

The boolean summation in three dimensions is given as

$$P[F] = P_r \oplus (P_s \oplus P_t)\ [F]$$

where P_r, P_s and P_t are the projector operators, and F is the field variable. The boolean summation yields with an approximation on the field variable F so that along the boundaries of the domain the continuity requirements are met in an exact manner.

III.b. Three-Dimensional Mesh Interface Elements

In order to demonstrate the power of new Mesh Interface Elements, the most commonly used three-dimensional element meshes are considered. These are the meshes of 8-noded and 20-noded elements. A three-dimensional Mesh Interface Element then is formulated by utilizing the boolean sum and the projector operators.

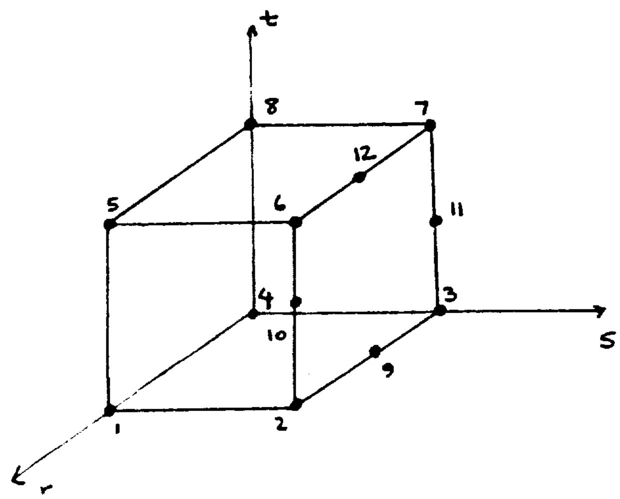


Figure 1. Three-Dimensional 12-noded Mesh Interface Element (12-MIE)

The formulation domain is [0,1] x [0,1] x [0,1]. This element will connect to an 8-noded solid element from (r,o,t) surface and to a 20-noded solid element from (r,1,t) surface. In addition, the field variable for this research, either the temperature field or the displacement field, is required to be continuous on the boundary surfaces of the element.

In order to construct the element to meet the constraints above, the projector operators become,

$$P_r[F] = rF(1,s,t) + (1-r)F(o,s,t)$$

$$P_s[F] = sF(r,1,t) + (1-s)F(r,o,t)$$

$$P_t[F] = tF(r,s,1) + (1-t)F(r,s,o)$$

The boolean sum then takes the form of

$$P[F] = (P_r + P_s + P_t - P_rP_s - P_sP_t - P_rP_t + P_rP_sP_t)[F]$$

Now, by selecting the surface interpolation functions, F(1,s,t)... F(r,s,o) to satisfy the continuity requirements we obtain the approximation function as

$$F \simeq \bar{F} = P[F]$$

or

$$\begin{aligned} \bar{F} = & (r-rs)(1-t)F_1 + (2rst^2 - 2r^2st + 2r^2s - rst - rs)F_2 \\ & + (-2rst^2 - 2r^2st + 2r^2s + 2st^2 + 5rst - 3st - 3rs + s)F_3 \\ & + (1-s-r+rs)(1-t)F_7 + (rt-rst)F_5 + (2rst^2 + 2sr^2t - 3rst)F_6 \\ & + (2r^2st - 2rst^2 + 2st^2 - rst - st)F_7 + (t-rt)(1-s)F_8 \\ & + (4rs-4sr^2)(1-t)F_9 + (4rst-4rst^2)F_{10} + (s-rs)(4+-4t^2)F_{11} \\ & + (4rst-4r^2st)F_{12} \end{aligned}$$

where $F_1, \ldots F_{12}$ are the nodal values of the field variable.

The material properties are considered to be known at nodal points of the element and

$$[A] = \sum_{i=1}^{NM} NMAT_i A_i$$

where $NMAT_i$ are the material shape functions.

The stiffness matrix is obtained and computer modules are included in the Appendix.

This element is tested for both heat transfer and structural analysis applications and performs well as a Mesh Interface Element.

III.b.2 Three-Two Dimensional Mesh Interface Elements

These elements are at present being tested and the results, together with the detailed formulation, will be reported at the end of the second level research.

IV. COMPUTER MODULES

Mesh Interface Elements are coded to solve heat transfer and structural analysis problems. Listing of these computer modules are included in the Appendix.

IV.a. Description of Computer Modules

ELEM Constructs the element stiffness matrix for 8-, 12- and 20-noded three-dimensional elements.

ELEM08 Calculates the derivatives of the shape functions of the 8-noded element.

ELEM12 Calculates the derivatives of the shape functions of the 12-noded mesh interface element.

ELEM20 Calculates the derivatives of the shape functions of the 20-noded three-dimensional element.

SHAP08 Calculates the shape functions of the 8-noded element.

SHAP12 Calculates the shape functions of the 12-noded mesh interface element.

SHAP20 Calculates the shape functions of the 20-noded element.

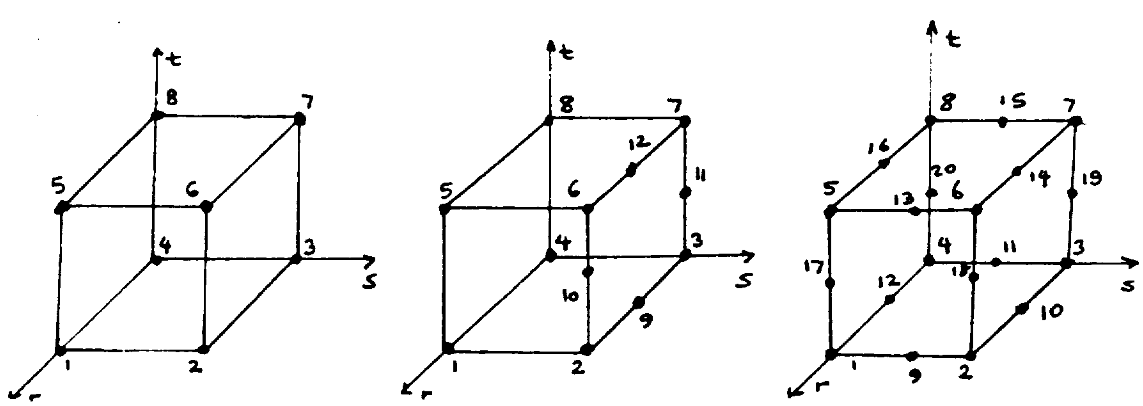


Figure 2. Element Numbering Scheme For Three-Dimensional Elements

In addition to the modules given on the previous page, a general purpose assembly module is structured in order to include two- and three-dimensional elements together with the mesh interface elements. Further, a "hidden symbolic computation" scheme is developed in order to compute the integrals, resulting from the element formulations, exactly. A library of these modules are:

POLDIF Differentiates a given polynomial

POLMLT Multiplies two polynomials

POLADD Adds two polynomials

POLINT Integrates a given polynomial

POLIEV Evaluates an integral at its upper and lower limits

It may be noted that the hidden symbolic computation scheme works well except that it is a bit time consuming at this stage to convert everything in this form.

At present, due to the time limitations, and also due to the major objectives of the research being rather different from the hidden-symbolic computation concept, this approach is set aside for the time being.

IV.b. List of Input Variables

NNODES: Number of nodes

NELEM: Number of elements

NBOUND: Number of boundary surfaces

COOR (I,J): x,y,z coordinates of the Ith node

NCONE(I,1): Element type for the Ith element

(NCONEC(I,J), J=2, NCONEC(I,1)+1): Nodal connectivities

FORCE (I): Element forces

NBKIND(I,1), NBKIND(I,2), NBKIND(I,3): Surface number, number of nodes, element number for the Ith boundary surface

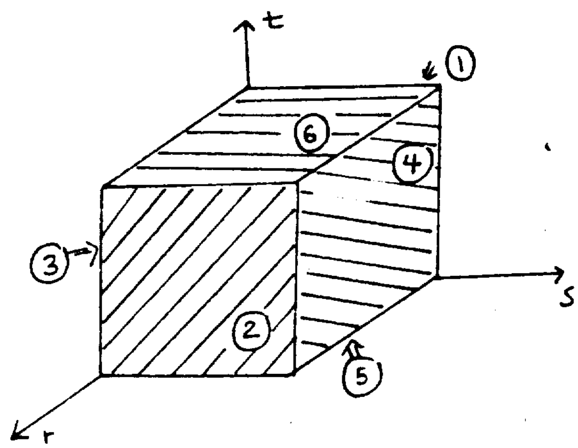


Figure 3. Surface Number for Three-Dimensional Elements

CBOUN (I,1), CBOUN (I,2), CBOUN (I,3): Prescribed heat flux convection coefficient outside temperature

COEF: Conduction coefficient

NTEMP: Number of nodes with prescribed temperatures

ITEMP, TEMP(ITEMP): Node number, corresponding temperature

IV.c. Computational Considerations

The computations are being performed on a VAX 11/780 computer. Larger scale applications will be solved at NASA Lewis Cray-XMP. It may be noted that since Cray uses the VAX 11/780 type computers as a front end, Cray runs will be performed readily.

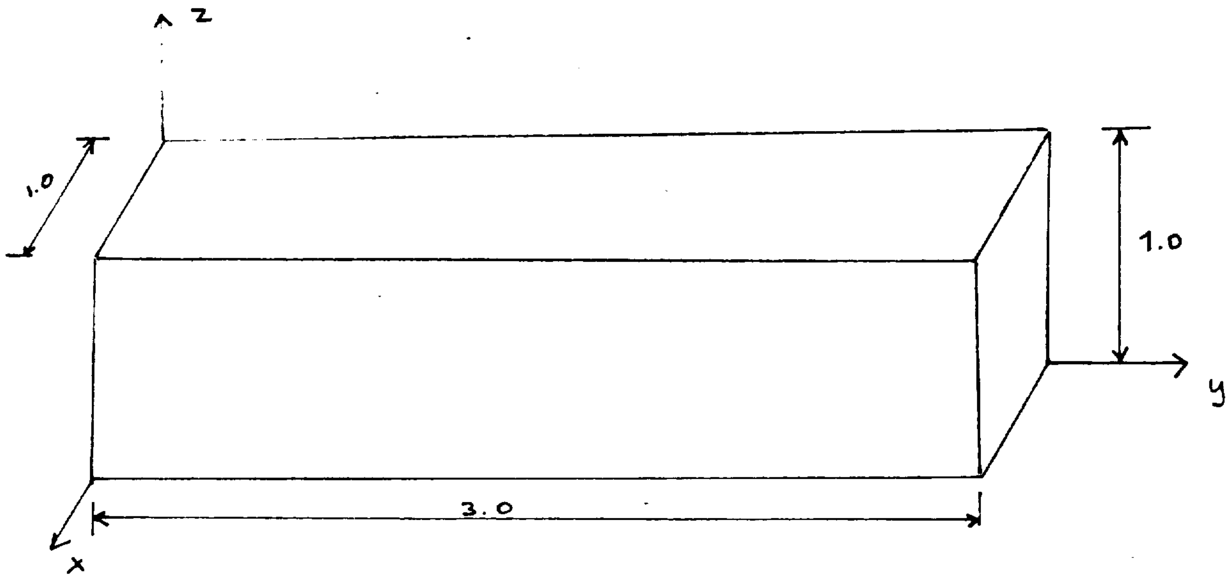
V. Numerical Verifications and Concluding Remarks

Mesh Interface Elements are being tested and the results obtained from three-dimensional mesh interface elements are included in this section. Numerical examples include heat transfer and structural analysis problems by using different mesh sizes and different element types. It may be noted that very good results are obtained when a mesh interface element is used to connect different element meshes together.

In concluding, the research will continue as it is given in the proposal. On the computational side, at this time, we need to utilize the Cray computer at NASA Lewis Research Center.

V.a. Heat Transfer Applications

STEADY STATE HEAT CONDUCTION 3 ELEMENT SOLUTION



$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} = 0$$

Boundary Conditions

$T|_{x=0} = y^2 - 3z^2$

$T|_{y=0} = 2x^2 - 3z^2$

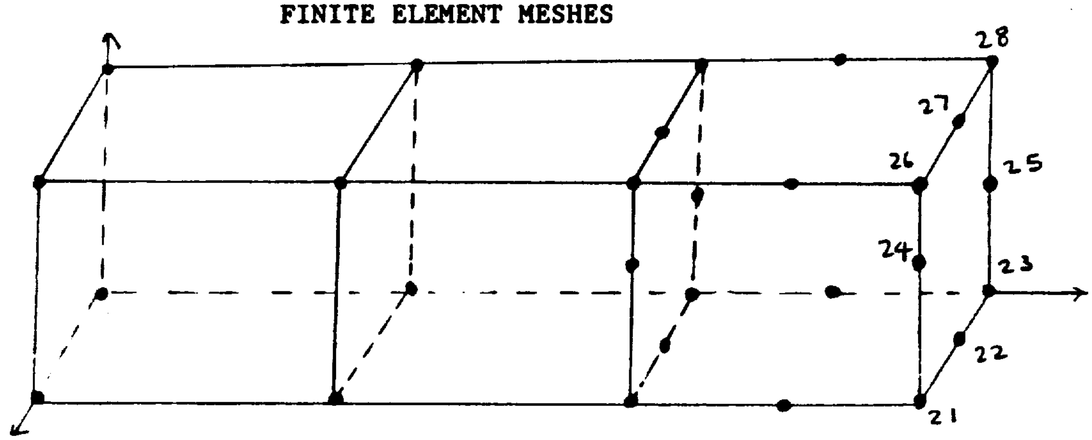
$T|_{z=0} = 2x^2 + y^2$

$\left.\frac{\partial T}{\partial x}\right|_{x=1} = 4$

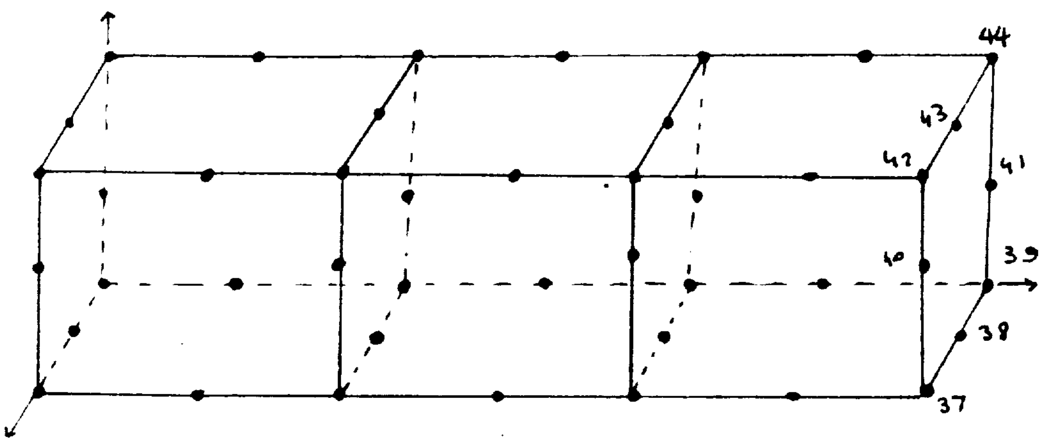
$\left.\frac{\partial T}{\partial y}\right|_{y=3} = 6$

$\left.\frac{\partial T}{\partial z}\right|_{z=1} = -6$

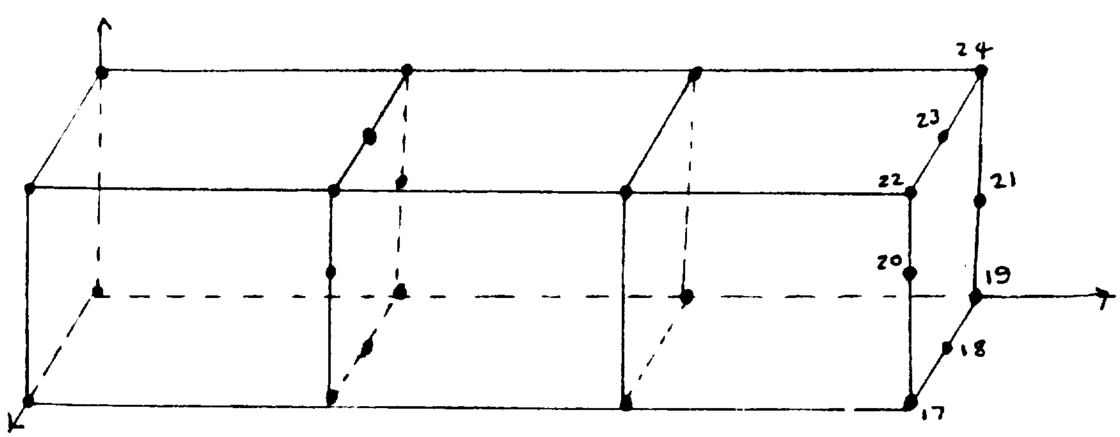
FINITE ELEMENT MESHES



Mesh I. 8-noded/12-noded/20-noded elements

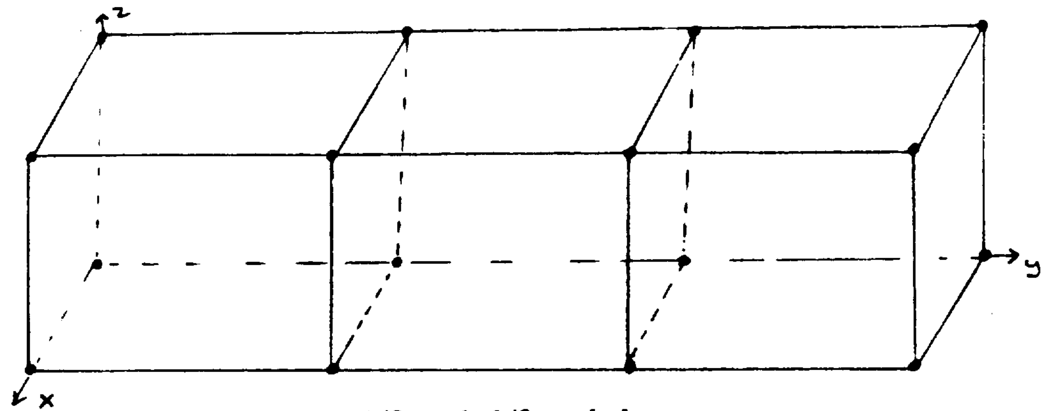


Mesh II. 20-noded/20-noded/20-noded elements



Mesh III. 12-noded/12-noded/12-noded elements

FINITE ELEMENT MESHES (Cont.)



Mesh IV. 8-noded/8-noded/8-noded

Results from each mesh types given in terms of temperature distributions and also % error as compared to the analytical values. Mesh interface element performs well as it is seen from the nodal temperatures.

** TEMPERATURE DISTRIBUTION **   MESH I (8/12/20)

NODE NUMBER	TEMPERATURES
1	0.2000000000D+01
2	0.7210119546D-28
3	-.1000000000D+01
4	-.3000000000D+01
5	0.3000000000D+01
6	0.1000000000D+01
7	0.6489107591D-01
8	-.2000000000D+01
9	0.5983739838D+01
10	0.4500000000D+01
11	0.4000000000D+01
12	0.5378099078D+01
13	0.3250000000D+01
14	0.2907929283D+01
15	0.1308523116D+01
16	0.1000000000D+01
17	0.8250000000D+01
18	0.6250000000D+01
19	0.5224836795D+01
20	0.3250000000D+01
21	0.1100000000D+02
22	0.9500000000D+01
23	0.9000000000D+01
24	0.1018167175D+02
25	0.8250000000D+01
26	0.8010500106D+01
27	0.6556661842D+01
28	0.6000000000D+01

NODE NUMBER	NUMERICAL VALUES	ANALYTICAL VALUES	ERROR
1	0.200000000D+01	0.200000000D+01	0.0000 %
2	0.721011955D-28	0.000000000D+00	0.0000 ***
3	-0.100000000D+01	-0.100000000D+01	0.0000 %
4	-0.300000000D+01	-0.300000000D+01	0.0000 %
5	0.300000000D+01	0.300000000D+01	0.0000 %
6	0.100000000D+01	0.100000000D+01	0.0000 %
7	0.648910759D-01	0.000000000D+00	0.0649 ***
8	-0.200000000D+01	-0.200000000D+01	0.0000 %
9	0.598373984D+01	0.600000000D+01	-0.2710 %
10	0.450000000D+01	0.450000000D+01	0.0000 %
11	0.400000000D+01	0.400000000D+01	0.0000 %
12	0.537809908D+01	0.525000000D+01	2.4400 %
13	0.325000000D+01	0.325000000D+01	0.0000 %
14	0.290792928D+01	0.300000000D+01	-3.0690 %
15	0.130852312D+01	0.150000000D+01	-12.7651 %
16	0.100000000D+01	0.100000000D+01	0.0000 %
17	0.825000000D+01	0.825000000D+01	0.0000 %
18	0.625000000D+01	0.625000000D+01	0.0000 %
19	0.522483679D+01	0.525000000D+01	-0.4793 %
20	0.325000000D+01	0.325000000D+01	0.0000 %
21	0.110000000D+02	0.110000000D+02	0.0000 %
22	0.950000000D+01	0.950000000D+01	0.0000 %
23	0.900000000D+01	0.900000000D+01	0.0000 %
24	0.101816718D+02	0.102500000D+02	-0.6666 %
25	0.825000000D+01	0.825000000D+01	0.0000 %
26	0.801050011D+01	0.800000000D+01	0.1313 %
27	0.655666184D+01	0.650000000D+01	0.8717 %
28	0.600000000D+01	0.600000000D+01	0.0000 %

MESH I (8/12/20)

** TEMPERATURE DISTRIBUTION **   MESH II (20/20/20)

NODE NUMBER	TEMPERATURES
1	0.2000000000D+01
2	0.5000000000D+00
3	0.0000000000D+00
4	0.1250000000D+01
5	-.7500000000D+00
6	-.1000000000D+01
7	-.2500000000D+01
8	-.3000000000D+01
9	0.2250000000D+01
10	0.2500000000D+00
11	-.7500000000D+00
12	-.2750000000D+01
13	0.3000000000D+01
14	0.1500000000D+01
15	0.1000000000D+01
16	0.2250000000D+01
17	0.2500000000D+00
18	0.5866785466D-14
19	-.1500000000D+01
20	-.2000000000D+01
21	0.4250000000D+01
22	0.2250000000D+01
23	0.1250000000D+01
24	-.7500000000D+00
25	0.6000000000D+01
26	0.4500000000D+01
27	0.4000000000D+01
28	0.5250000000D+01
29	0.3250000000D+01
30	0.3000000000D+01
31	0.1500000000D+01
32	0.1000000000D+01
33	0.8250000000D+01
34	0.6250000000D+01
35	0.5250000000D+01
36	0.3250000000D+01
37	0.1100000000D+02
38	0.9500000000D+01
39	0.9000000000D+01
40	0.1025000000D+02
41	0.8250000000D+01
42	0.8000000000D+01
43	0.6500000000D+01
44	0.6000000000D+01

NODE NUMBER	NUMERICAL VALUES	ANALYTICAL VALUES	ERROR
1	0.200000000D+01	0.200000000D+01	0.0000 %
2	0.500000000D+00	0.500000000D+00	0.0000 %
3	0.000000000D+00	0.000000000D+00	0.0000 ***
4	0.125000000D+01	0.125000000D+01	0.0000 %
5	-0.750000000D+00	-0.750000000D+00	0.0000 %
6	-0.100000000D+01	-0.100000000D+01	0.0000 %
7	-0.250000000D+01	-0.250000000D+01	0.0000 %
8	-0.300000000D+01	-0.300000000D+01	0.0000 %
9	0.225000000D+01	0.225000000D+01	0.0000 %
10	0.250000000D+00	0.250000000D+00	0.0000 %
11	-0.750000000D+00	-0.750000000D+00	0.0000 %
12	-0.275000000D+01	-0.275000000D+01	0.0000 %
13	0.300000000D+01	0.300000000D+01	0.0000 %
14	0.150000000D+01	0.150000000D+01	0.0000 %
15	0.100000000D+01	0.100000000D+01	0.0000 %
16	0.225000000D+01	0.225000000D+01	0.0000 %
17	0.250000000D+00	0.250000000D+00	0.0000 %
18	0.586678547D-14	0.000000000D+00	0.0000 ***
19	-0.150000000D+01	-0.150000000D+01	0.0000 %
20	-0.200000000D+01	-0.200000000D+01	0.0000 %
21	0.425000000D+01	0.425000000D+01	0.0000 %
22	0.225000000D+01	0.225000000D+01	0.0000 %
23	0.125000000D+01	0.125000000D+01	0.0000 %
24	-0.750000000D+00	-0.750000000D+00	0.0000 %
25	0.600000000D+01	0.600000000D+01	0.0000 %
26	0.450000000D+01	0.450000000D+01	0.0000 %
27	0.400000000D+01	0.400000000D+01	0.0000 %
28	0.525000000D+01	0.525000000D+01	0.0000 %
29	0.325000000D+01	0.325000000D+01	0.0000 %
30	0.300000000D+01	0.300000000D+01	0.0000 %
31	0.150000000D+01	0.150000000D+01	0.0000 %
32	0.100000000D+01	0.100000000D+01	0.0000 %
33	0.825000000D+01	0.825000000D+01	0.0000 %
34	0.625000000D+01	0.625000000D+01	0.0000 %
35	0.525000000D+01	0.525000000D+01	0.0000 %
36	0.325000000D+01	0.325000000D+01	0.0000 %
37	0.110000000D+02	0.110000000D+02	0.0000 %
38	0.950000000D+01	0.950000000D+01	0.0000 %
39	0.900000000D+01	0.900000000D+01	0.0000 %
40	0.102500000D+02	0.102500000D+02	0.0000 %
41	0.825000000D+01	0.825000000D+01	0.0000 %
42	0.800000000D+01	0.800000000D+01	0.0000 %
43	0.650000000D+01	0.650000000D+01	0.0000 %
44	0.600000000D+01	0.600000000D+01	0.0000 %

MESH II (20/20/20)

** TEMPERATURE DISTRIBUTION **   MESH III (12/12/12)

NODE NUMBER	TEMPERATURES
1	0.2000000000D+01
2	0.4672171752D-28
3	-.1000000000D+01
4	-.3000000000D+01
5	0.3000000000D+01
6	0.1500000000D+01
7	0.1000000000D+01
8	0.2497546431D+01
9	0.2500000000D+00
10	-.1704954577D+00
11	-.1849798702D+01
12	-.2000000000D+01
13	0.6000000000D+01
14	0.4000000000D+01
15	0.3137206428D+01
16	0.1000000000D+01
17	0.1100000000D+02
18	0.9500000000D+01
19	0.9000000000D+01
20	0.1049921156D+02
21	0.8250000000D+01
22	0.7822844036D+01
23	0.6151866425D+01
24	0.6000000000D+01

NODE NUMBER	NUMERICAL VALUES	ANALYTICAL VALUES	ERROR
1	0.200000000D+01	0.200000000D+01	0.0000 %
2	0.467217175D-28	0.000000000D+00	0.0000 ***
3	-0.100000000D+01	-0.100000000D+01	0.0000 %
4	-0.300000000D+01	-0.300000000D+01	0.0000 %
5	0.300000000D+01	0.300000000D+01	0.0000 %
6	0.150000000D+01	0.150000000D+01	0.0000 %
7	0.100000000D+01	0.100000000D+01	0.0000 %
8	0.249754643D+01	0.225000000D+01	11.0021 %
9	0.250000000D+00	0.250000000D+00	0.0000 %
10	-0.170495458D+00	0.000000000D+00	-0.1705 ***
11	-0.184979870D+01	-0.150000000D+01	23.3199 %
12	-0.200000000D+01	-0.200000000D+01	0.0000 %
13	0.600000000D+01	0.600000000D+01	0.0000 %
14	0.400000000D+01	0.400000000D+01	0.0000 %
15	0.313720643D+01	0.300000000D+01	4.5735 %
16	0.100000000D+01	0.100000000D+01	0.0000 %
17	0.110000000D+02	0.110000000D+02	0.0000 %
18	0.950000000D+01	0.950000000D+01	0.0000 %
19	0.900000000D+01	0.900000000D+01	0.0000 %
20	0.104992116D+02	0.102500000D+02	2.4313 %
21	0.825000000D+01	0.825000000D+01	0.0000 %
22	0.782284404D+01	0.800000000D+01	-2.2144 %
23	0.615186643D+01	0.650000000D+01	-5.3559 %
24	0.600000000D+01	0.600000000D+01	0.0000 %

MESH III (12/12/12)

** TEMPERATURE DISTRIBUTION **   MESH IV (8/6/8)

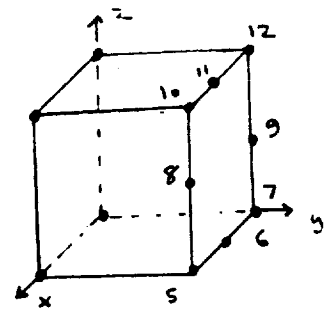
NODE NUMBER	TEMPERATURES
1	0.2000000000D+01
2	0.2104797818D-34
3	-.1000000000D+01
4	-.3000000000D+01
5	0.3000000000D+01
6	0.1000000000D+01
7	0.7559057547D-14
8	-.2000000000D+01
9	0.6000000000D+01
10	0.4000000000D+01
11	0.3000000000D+01
12	0.1000000000D+01
13	0.1100000000D+02
14	0.9000000000D+01
15	0.8000000000D+01
16	0.6000000000D+01

TEMPERATURE DISTRIBUTION - MESH IV (8/8/8)

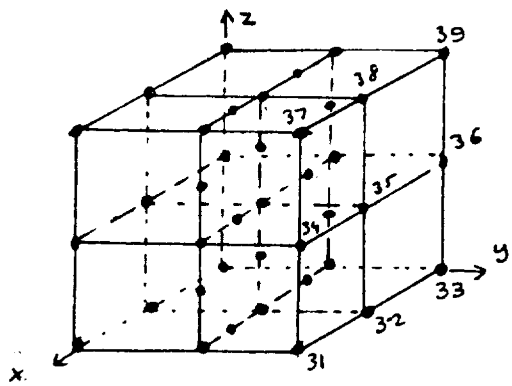
NODE NUMBER	NUMERICAL NUMBER	ANALYTICAL VALUES	ERROR
1	0.2000000000D+01	0.200000000D+01	0.0000 %
2	0.2104797818D-34	0.000000000D+00	0.0000 ***
3	-.1000000000D+01	-0.100000000D+01	0.0000 %
4	-.3000000000D+01	-0.300000000D+01	0.0000 %
5	0.3000000000D+01	0.300000000D+01	0.0000 %
6	0.1000000000D+01	0.100000000D+01	0.0000 %
7	0.7559057547D-14	0.000000000D+00	0.0649 ***
8	-.2000000000D+01	-0.200000000D+01	0.0000 %
9	0.6000000000D+01	0.600000000D+01	0.0000 %
10	0.4000000000D+01	0.400000000D+01	0.0000 %
11	0.3000000000D+01	0.300000000D+01	0.0000 %
12	0.1000000000D+01	0.100000000D+01	0.0000 %
13	0.1100000000D+02	0.110000000D+02	0.0000 %
14	0.9000000000D+01	0.900000000D+01	0.0000 %
15	0.8000000000D+01	0.800000000D+01	0.0000 %
16	0.6000000000D+01	0.600000000D+01	0.0000 %

ERROR ANALYSES

FINITE ELEMENT MESHES FOR ERROR ANALYSES I.



Mesh I. 1 12-noded element (max error = -0.1554)



Mesh II. 8 12-noded elements (max error = -0.0475)

Mesh III. 64 12-noded elements (max error = -0.0119)

NODE	NUMERICAL	ANALYTICAL	ERROR	% ERROR
1	-0.10000D+01	-0.10000D+01	0.0000	0.0000 %
2	0.25907D-33	0.00000D+00	0.0000	***
3	-0.10000D+01	-0.10000D+01	0.0000	0.0000 %
4	0.25907D-33	0.00000D+00	0.0000	***
5	-0.15544D+00	0.00000D+00	-0.1554	***
6	0.73316D+00	0.75000D+00	-0.0168	-2.2453 %
7	0.10000D+01	0.10000D+01	0.0000	0.0000 %
8	-0.15544D+00	0.00000D+00	-0.1554	***
9	0.10000D+01	0.10000D+01	0.0000	0.0000 %
10	-0.15544D+00	0.00000D+00	-0.1554	***
11	0.73316D+00	0.75000D+00	-0.0168	-2.2453 %
12	0.10000D+01	0.10000D+01	0.0000	0.0000 %

MESH I (1 12 noded element)

Max error = -0.1554

NODE	NUMERICAL	ANALYTICAL	ERROR	% ERROR
1	-0.10000D+01	-0.10000D+01	0.0000	0.0000 %
2	-0.25000D+00	-0.25000D+00	0.0000	0.0000 %
3	0.30528D-34	0.00000D+00	0.0000	***
4	-0.10000D+01	-0.10000D+01	0.0000	0.0000 %
5	-0.25000D+00	-0.25000D+00	0.0000	0.0000 %
6	0.61057D-34	0.00000D+00	0.0000	***
7	-0.10000D+01	-0.10000D+01	0.0000	0.0000 %
8	-0.25000D+00	-0.25000D+00	0.0000	0.0000 %
9	0.30528D-34	0.00000D+00	0.0000	***
10	-0.79747D+00	-0.75000D+00	-0.0475	-6.3294 %
11	-0.33249D+00	-0.31250D+00	-0.0200	-6.3981 %
12	-0.43250D-01	0.00000D+00	-0.0432	***
13	0.18326D+00	0.18750D+00	-0.0042	-2.2618 %
14	0.25000D+00	0.25000D+00	0.0000	0.0000 %
15	-0.79747D+00	-0.75000D+00	-0.0475	-6.3294 %
16	-0.43250D-01	0.00000D+00	-0.0432	***
17	0.25000D+00	0.25000D+00	0.0000	0.0000 %
18	-0.79747D+00	-0.75000D+00	-0.0475	-6.3294 %
19	-0.33249D+00	-0.31250D+00	-0.0200	-6.3981 %
20	-0.43250D-01	0.00000D+00	-0.0432	***
21	0.18326D+00	0.18750D+00	-0.0042	-2.2618 %
22	0.25000D+00	0.25000D+00	0.0000	0.0000 %
23	-0.79747D+00	-0.75000D+00	-0.0475	-6.3294 %
24	-0.43250D-01	0.00000D+00	-0.0432	***
25	0.25000D+00	0.25000D+00	0.0000	0.0000 %
26	-0.79747D+00	-0.75000D+00	-0.0475	-6.3294 %
27	-0.33249D+00	-0.31250D+00	-0.0200	-6.3981 %
28	-0.43250D-01	0.00000D+00	-0.0432	***
29	0.18326D+00	0.18750D+00	-0.0042	-2.2618 %
30	0.25000D+00	0.25000D+00	0.0000	0.0000 %
31	0.15747D-01	0.00000D+00	0.0157	***
32	0.77123D+00	0.75000D+00	0.0212	2.8301 %
33	0.10000D+01	0.10000D+01	0.0000	0.0000 %
34	0.15747D-01	0.00000D+00	0.0157	***
35	0.77123D+00	0.75000D+00	0.0212	2.8301 %
36	0.10000D+01	0.10000D+01	0.0000	0.0000 %
37	0.15747D-01	0.00000D+00	0.0157	***
38	0.77123D+00	0.75000D+00	0.0212	2.8301 %
39	0.10000D+01	0.10000D+01	0.0000	0.0000 %

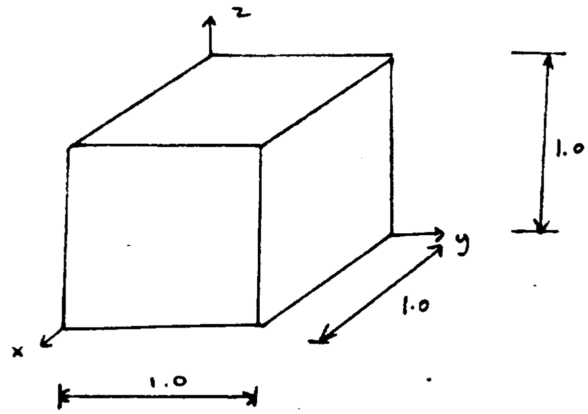
MESH II 8 12-noded elements (max error = -0.0475)

ORIGINAL PAGE IS
OF POOR QUALITY

NODE	NUMERICAL	ANALYTICAL	ERROR	% ERROR
1	-0.25000D+00	-0.25000D+00	0.0000	0.0000 %
2	-0.62500D-01	-0.62500D-01	0.0000	0.0000 %
3	0.38161D-35	0.00000D+00	0.0000	***
4	-0.25000D+00	-0.25000D+00	0.0000	0.0000 %
5	-0.62500D-01	-0.62500D-01	0.0000	0.0000 %
6	0.76321D-35	0.00000D+00	0.0000	***
7	-0.25000D+00	-0.25000D+00	0.0000	0.0000 %
8	-0.62500D-01	-0.62500D-01	0.0000	0.0000 %
9	0.38161D-35	0.00000D+00	0.0000	***
10	-0.19937D+00	-0.18750D+00	-0.0119	-6.3294 %
11	-0.83123D-01	-0.78125D-01	-0.0050	-6.3981 %
12	-0.10812D-01	0.00000D+00	-0.0108	***
13	0.45815D-01	0.46875D-01	-0.0011	-2.2618 %
14	0.62500D-01	0.62500D-01	0.0000	0.0000 %
15	-0.19937D+00	-0.18750D+00	-0.0119	-6.3294 %
16	-0.10812D-01	0.00000D+00	-0.0108	***
17	0.62500D-01	0.62500D-01	0.0000	0.0000 %
18	-0.19937D+00	-0.18750D+00	-0.0119	-6.3294 %
19	-0.83123D-01	-0.78125D-01	-0.0050	-6.3981 %
20	-0.10812D-01	0.00000D+00	-0.0108	***
21	0.45815D-01	0.46875D-01	-0.0011	-2.2618 %
22	0.62500D-01	0.62500D-01	0.0000	0.0000 %
23	-0.19937D+00	-0.18750D+00	-0.0119	-6.3294 %
24	-0.10812D-01	0.00000D+00	-0.0108	***
25	0.62500D-01	0.62500D-01	0.0000	0.0000 %
26	-0.19937D+00	-0.18750D+00	-0.0119	-6.3294 %
27	-0.83123D-01	-0.78125D-01	-0.0050	-6.3981 %
28	-0.10812D-01	0.00000D+00	-0.0108	***
29	0.45815D-01	0.46875D-01	-0.0011	-2.2618 %
30	0.62500D-01	0.62500D-01	0.0000	0.0000 %
31	0.39368D-02	0.00000D+00	0.0039	***
32	0.19281D+00	0.18750D+00	0.0053	2.8301 %
33	0.25000D+00	0.25000D+00	0.0000	0.0000 %
34	0.39368D-02	0.00000D+00	0.0039	***
35	0.19281D+00	0.18750D+00	0.0053	2.8301 %
36	0.25000D+00	0.25000D+00	0.0000	0.0000 %
37	0.39368D-02	0.00000D+00	0.0039	***
38	0.19281D+00	0.18750D+00	0.0053	2.8301 %
39	0.25000D+00	0.25000D+00	0.0000	0.0000 %

MESH III 64 12-noded elements (Max error = -0.0119)

ERROR ANALYSES II. (1 element, 8 element solutions)



$\nabla^2 T - 18 = 0$

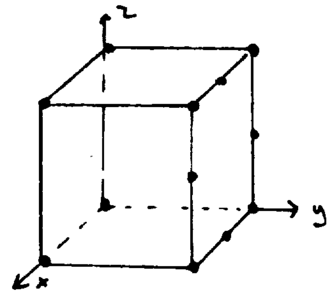
Boundary Conditions

$T|_{x=0} = 3y^2 + 4z^2$ $\qquad \dfrac{\partial T}{\partial x}\Big|_{x=1} = 4$

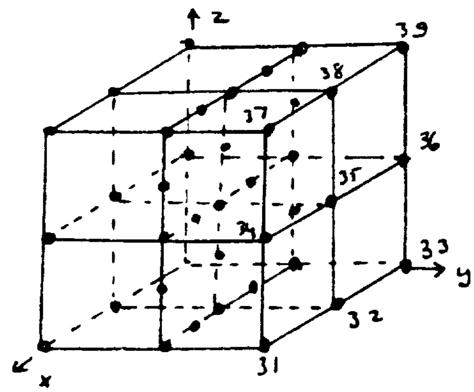
$T|_{y=0} = 2x^2 + 4z^2$ $\qquad \dfrac{\partial T}{\partial y}\Big|_{y=1} = 6$

$T|_{z=0} = 2x^2 + 3y^2$ $\qquad \dfrac{\partial T}{\partial z}\Big|_{z=1} = 8$

FINITE ELEMENT MESHES FOR ERROR ANALYSES II.



Mesh I. 1 12-noded element (max error = 0.983)



Mesh II. 8 12-noded elements (max error = 0.2733)

NODE	NUMERICAL	ANALYTICAL	ERROR	% ERROR
1	0.20000D+01	0.20000D+01	0.0000	0.0000 %
2	-0.20692D-32	0.00000D+00	0.0000	***
3	0.60000D+01	0.60000D+01	0.0000	0.0000 %
4	0.40000D+01	0.40000D+01	0.0000	0.0000 %
5	0.50000D+01	0.50000D+01	0.0000	0.0000 %
6	0.35000D+01	0.35000D+01	0.0000	0.0000 %
7	0.30000D+01	0.30000D+01	0.0000	0.0000 %
8	0.61973D+01	0.60000D+01	0.1973	3.2880 %
9	0.40000D+01	0.40000D+01	0.0000	0.0000 %
10	0.99830D+01	0.90000D+01	0.9830	10.9223 %
11	0.79362D+01	0.75000D+01	0.4362	5.8162 %
12	0.70000D+01	0.70000D+01	0.0000	0.0000 %

ERROR ANALYSES II

MESH I (max error = 0.983)

NODE	NUMERICAL	ANALYTICAL	ERROR	% ERROR
1	0.20000D+01	0.20000D+01	0.0000	0.0000 %
2	0.50000D+00	0.50000D+00	0.0000	0.0000 %
3	-0.24968D-33	0.00000D+00	0.0000	***
4	0.30000D+01	0.30000D+01	0.0000	0.0000 %
5	0.15000D+01	0.15000D+01	0.0000	0.0000 %
6	0.10000D+01	0.10000D+01	0.0000	0.0000 %
7	0.60000D+01	0.60000D+01	0.0000	0.0000 %
8	0.45000D+01	0.45000D+01	0.0000	0.0000 %
9	0.40000D+01	0.40000D+01	0.0000	0.0000 %
10	0.27500D+01	0.27500D+01	0.0000	0.0000 %
11	0.18750D+01	0.18750D+01	0.0000	0.0000 %
12	0.12500D+01	0.12500D+01	0.0000	0.0000 %
13	0.87500D+00	0.87500D+00	0.0000	0.0000 %
14	0.75000D+00	0.75000D+00	0.0000	0.0000 %
15	0.30766D+01	0.30000D+01	0.0766	2.5547 %
16	0.15632D+01	0.15000D+01	0.0632	4.2136 %
17	0.10000D+01	0.10000D+01	0.0000	0.0000 %
18	0.40072D+01	0.37500D+01	0.2572	6.8592 %
19	0.30749D+01	0.28750D+01	0.1999	6.9525 %
20	0.25183D+01	0.22500D+01	0.2683	11.9239 %
21	0.19937D+01	0.18750D+01	0.1187	6.3326 %
22	0.17500D+01	0.17500D+01	0.0000	0.0000 %
23	0.51688D+01	0.50000D+01	0.1688	3.3758 %
24	0.36437D+01	0.35000D+01	0.1437	4.1055 %
25	0.30000D+01	0.30000D+01	0.0000	0.0000 %
26	0.70112D+01	0.67500D+01	0.2612	3.8692 %
27	0.60986D+01	0.58750D+01	0.2236	3.8064 %
28	0.55233D+01	0.52500D+01	0.2733	5.2049 %
29	0.50042D+01	0.48750D+01	0.1292	2.6502 %
30	0.47500D+01	0.47500D+01	0.0000	0.0000 %
31	0.50000D+01	0.50000D+01	0.0000	0.0000 %
32	0.35000D+01	0.35000D+01	0.0000	0.0000 %
33	0.30000D+01	0.30000D+01	0.0000	0.0000 %
34	0.58681D+01	0.60000D+01	-0.1319	-2.1991 %
35	0.43490D+01	0.45000D+01	-0.1510	-3.3546 %
36	0.40000D+01	0.40000D+01	0.0000	0.0000 %
37	0.88888D+01	0.90000D+01	-0.1112	-1.2355 %
38	0.73743D+01	0.75000D+01	-0.1257	-1.6756 %
39	0.70000D+01	0.70000D+01	0.0000	0.0000 %

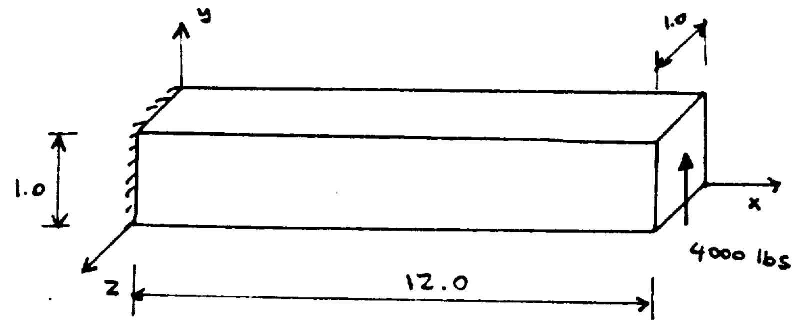
ERROR ANALYSES II

MESH II (max error = 0.2733)

8 12 noded elements

V.b. Structural Analysis Applications

DISPLACEMENT ANALYSIS (END LOADING)



$E = 2.9 \times 10^7$ $\nu = 0.3$

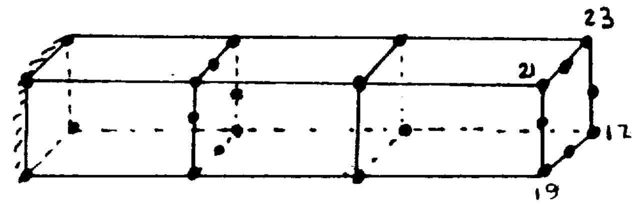
Boundary Conditions

at x=0 built-in end

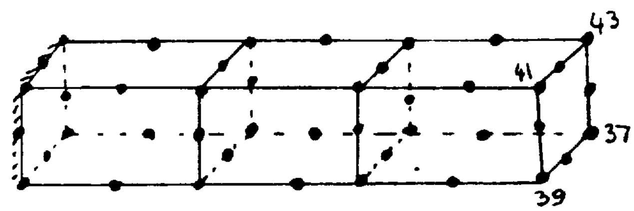
at all points w is prescribed

The problem is solved by using different finite element mesh types. Results obtained demonstrate the power of the mesh interface elements.

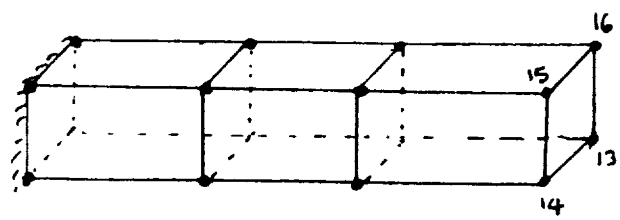
FINITE ELEMENT MESHES FOR END LOADING PROBLEM



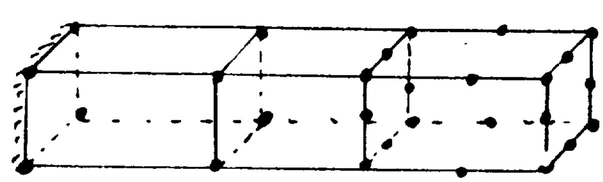
Mesh I. 3 12-noded elements



Mesh II. 3 20-noded elements



Mesh III. 3 8-noded elements



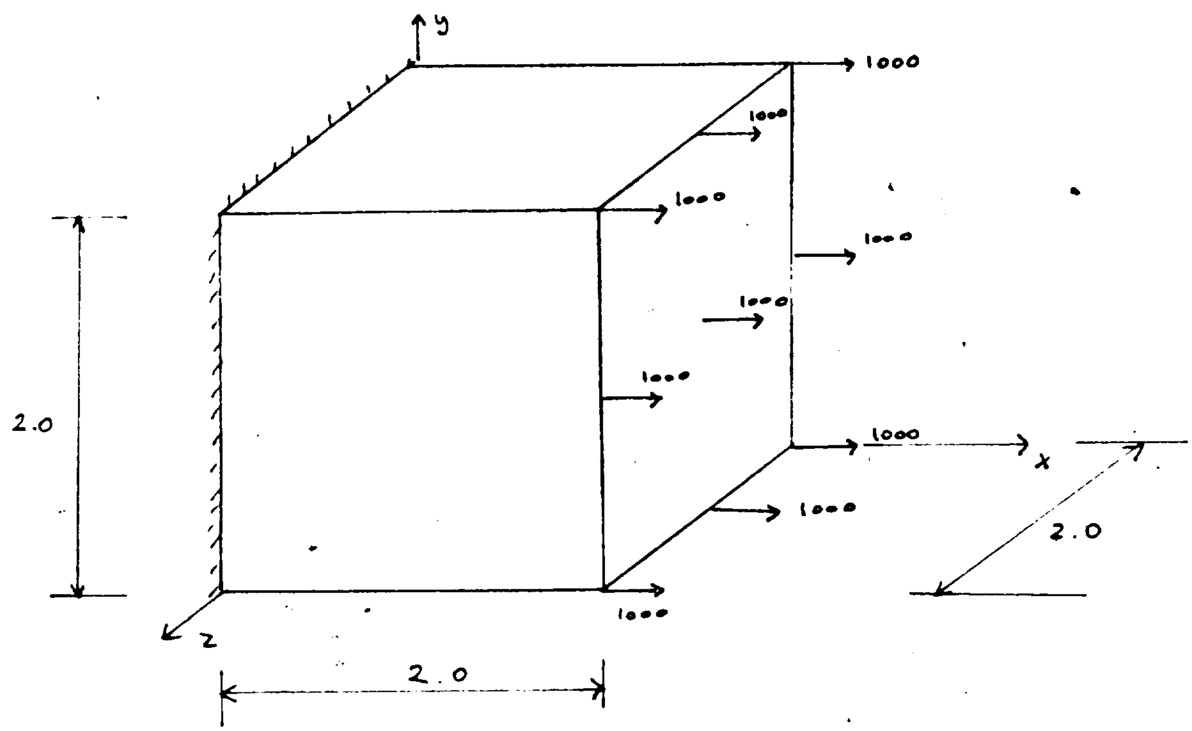
Mesh IV. 8-noded/12-noded/20-noded elements

```
TEST CASE:BEAM 3  8/12/20 -ELEM  (MESH IV)
1  0
28  28  3  3  3  28  1  4  8  6  1  3  0  0  0
8
1   1  14    1   5   8   4   2   6   7   3
12
2   1  15    5   9  15   8   6  11  13   7
            16  10  14  12
20
3   1  16    9  21  27  15  11  23  25  13
            17  28  20  16  10  22  26  14
            18  24  19  12
1      0.    0.0       0.0
2      0.    0.0       1.0
3      0.    1.0       1.0
4      0.    1.0       0.0
5      1.    0.0       0.0
6      1.    0.0       1.0
7      1.    1.0       1.0
8      1.    1.0       0.0
9      2.    0.0       0.0
10     2.    0.0       0.5
11     2.    0.0       1.0
12     2.    0.5       1.0
13     2.    1.0       1.0
14     2.    1.0       0.5
15     2.    1.0       0.0
16     2.    0.5       0.0
17     3.    0.0       0.0
18     3.    0.0       1.0
19     3.    1.0       1.0
20     3.    1.0       0.0
21     3.    0.0       0.0
22     3.    0.0       0.5
23     3.    0.0       1.0
24     3.    0.5       1.0
25     3.    1.0       1.0
26     3.    1.0       0.5
27     3.    1.0       0.0
28     3.    0.5       0.0
1  2.95E+07  0.3    1.    0.0
1  1  0.  1  0.  1  0.
2  1  0.  1  0.  1  0.
3  1  0.  1  0.  1  0.
4  1  0.  1  0.  1  0.
5  0  0.  0  0.  1  0.
6  0  0.  0  0.  1  0.
7  0  0.  0  0.  1  0.
8  0  0.  0  0.  1  0.
9  0  0.  0  0.  1  0.
10 0  0.  0  0.  1  0.
11 0  0.  0  0.  1  0.
12 0  0.  0  0.  1  0.
13 0  0.  0  0.  1  0.
14 0  0.  0  0.  1  0.
15 0  0.  0  0.  1  0.
16 0  0.  0  0.  1  0.
17 0  0.  0  0.  1  0.
18 0  0.  0  0.  1  0.
19 0  0.  0  0.  1  0.
20 0  0.  0  0.  1  0.
21 0  0.  0  0.  1  0.
22 0  0.  0  0.  1  0.
23 0  0.  0  0.  1  0.
24 0  0.  0  0.  1  0.
25 0  0.  0  0.  1  0.
26 0  0.  0  0.  1  0.
27 0  0.  0  0.  1  0.
28 0  0.  0  0.  1  0.
21     0.   500.  0.
22     0.   500.  0.
23     0.   500.  0.
24     0.   500.  0.
25     0.   500.  0.
26     0.   500.  0.
27     0.   500.  0.
28     0.   500.  0.
```

DEFLECTIONS AT THE TIP (END LOADING)
(x=12, y=1, z=0)

Mesh Type	u	v	w
Mesh I	-0.002388	0.010315	Presc.
Mesh II	-0.00323	0.01373	Presc.
Mesh III	-0.00211	0.00928	Presc.
Mesh IV	-0.00236	0.00983	Presc.

DISPLACEMENT ANALYSIS (AXIAL LOAD)



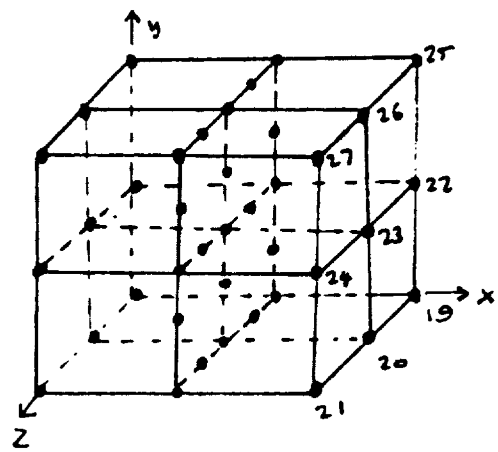
$E = 2.95 \times 10^7$

$v = 0.3$

Boundary Conditions

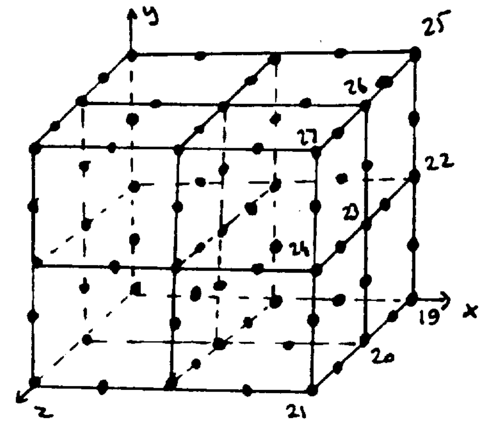
Built-in at x=0

DISPLACEMENT ANALYSIS (AXIAL LOAD)
Mesh I (8 12-noded elements)



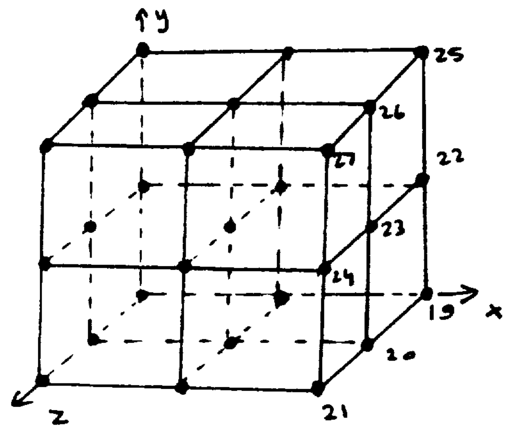
Node Number	Tip Deflections u	Tip Deflections v	Tip Deflections w
19	0.2347×10^{-3}	0.4313×10^{-4}	0.4313×10^{-4}
20	0.1242×10^{-3}	0.2760×10^{-4}	0.0
21	0.2347×10^{-3}	0.4313×10^{-4}	-0.4313×10^{-4}
22	0.1242×10^{-3}	0.0	0.2760×10^{-4}
23	0.9639×10^{-4}	0.0	0.0
24	0.12417×10^{-3}	0.0	-0.2760×10^{-4}
25	0.2347×10^{-3}	-0.4313×10^{-4}	0.4313×10^{-4}
26	0.1242×10^{-3}	-0.2760×10^{-4}	0.0
27	0.2347×10^{-3}	-0.4313×10^{-4}	-0.4313×10^{-4}

DISPLACEMENT ANALYSIS (AXIAL LOAD)
Mesh II (8 20-noded elements)



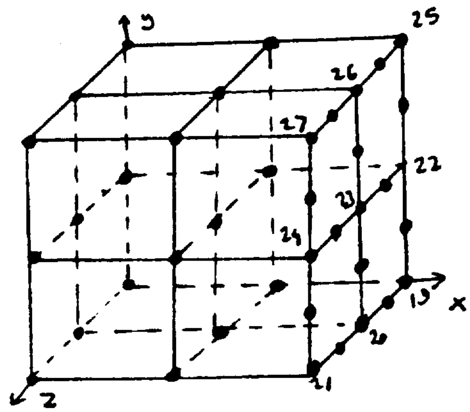
Node Number	Tip Deflections u	v	w
19	0.5233×10^{-3}	0.1393×10^{-3}	0.1393×10^{-3}
20	0.2191×10^{-3}	0.505×10^{-4}	0.0
21	0.5233×10^{-3}	0.1393×10^{-3}	-0.1393×10^{-3}
22	0.2191×10^{-3}	0.0	0.505×10^{-4}
23	0.9122×10^{-4}	0.0	0.0
24	0.2191×10^{-3}	0.0	-0.505×10^{-4}
25	0.5233×10^{-3}	-0.1393×10^{-3}	0.1393×10^{-3}
26	0.2191×10^{-3}	-0.505×10^{-4}	0.0
27	0.5233×10^{-3}	-0.1393×10^{-3}	-0.1393×10^{-3}

DISPLACEMENT ANALYSIS (AXIAL LOAD)
Mesh III (8 8-noded elements)



Node Number	Tip Deflections u	v	w
19	0.2265×10^{-3}	0.4194×10^{-4}	0.4194×10^{-4}
20	0.1252×10^{-3}	0.2727×10^{-4}	0.0
21	0.2265×10^{-3}	0.4194×10^{-4}	-0.4194×10^{-4}
22	0.1252×10^{-3}	0.0	0.2727×10^{-4}
23	0.9994×10^{-4}	0.0	0.0
24	0.1252×10^{-3}	0.0	-0.2727×10^{-4}
25	0.2265×10^{-3}	-0.4194×10^{-4}	0.4194×10^{-4}
26	0.1252×10^{-3}	-0.2727×10^{-4}	0.0
27	0.2265×10^{-3}	-0.4194×10^{-4}	-0.4194×10^{-4}

DISPLACEMENT ANALYSIS (AXIAL LOAD)
Mesh IV (4 8-noded/4 12-noded elements)



8 12

Node Number	Tip Deflections u	v	w
19	0.4129×10^{-3}	0.722×10^{-4}	0.722×10^{-4}
20	0.2142×10^{-3}	0.3731×10^{-4}	0.0
21	0.4129×10^{-3}	0.722×10^{-4}	-0.722×10^{-4}
22	0.2142×10^{-3}	0.0	0.3731×10^{-4}
23	0.1089×10^{-3}	0.0	0.0
24	0.2142×10^{-3}	0.0	-0.3731×10^{-4}
25	0.4129×10^{-3}	-0.722×10^{-4}	0.722×10^{-4}
26	0.2141×10^{-3}	-0.3731×10^{-4}	0.0
27	0.4129×10^{-3}	-0.7221×10^{-4}	-0.722×10^{-4}

```
C
C
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 LU
C
      DIMENSION COOR(200,3),NCONEC(20,21),NBOUN(100,8),NBKIND(100,3)
      DIMENSION CBOUN(100,3),GSTIF(200,200),GLOAD(200),GTEMP(200)
      DIMENSION STIF(20,20),ELOAD(8),ESTIF(8,8),TEMP(200)
      DIMENSION INDEXR(200),LU(200,200),W(6),Z(6),SHAP(200)
      DIMENSION TEMPAN(200),C(200),FORCE(20),BLOAD(20)
C
C     N1= Max number of nodes
C     N2= Max number of boundary surfaces
C     N3= Max number of nodes in any element 'in this case N3=20'
C     N4= Max number of elements
C
      DATA N1,N2,N3,N4/200,100,20,20/
C     DATA (W(M),M=1,3)/.888888888888889,.555555555555556,.555555555555556/
C     DATA (Z(M),M=1,3)/.0,.774596669241483,-.774596669241483/
C
      DATA (W(M),M=1,6)/.467913934572691,.467913934572691,
     >                  .360761573048139,.360761573048139,
     >                  .171324492379170,.171324492379170/
C
      DATA (Z(M),M=1,6)/.238619186083197,-.238619186083197,
     >                  .661209386466265,-.661209386466265,
     >                  .932469514203152,-.932469514203152/
C
      DATA PENALT/100000000000000000000000000000000./
C
C
      OPEN (UNIT=1,FILE='ONUR.DAT',STATUS='OLD',READONLY)
      OPEN (UNIT=2,FILE='ONUR.OUT',STATUS='NEW')
C
C
C
C     READ NUMBER OF NODES,ELEMENTS,BOUNDARIES
C
      READ(1,*) NNODES
      READ(1,*) NELEM
      READ(1,*) NBOUND
C
C     READ COORDINATES
C
C
      DO 10 I=1,NNODES
         READ(1,*) COOR(I,1),COOR(I,2),COOR(I,3)
   10 CONTINUE
C
C
C     READ CONNECTIVITIES
C
      DO 20 I=1,NELEM
C
C     * NCONEC(I,1) defines the type of the element '8,12 or 20'
C
         READ(1,21) NCONEC(I,1)
         READ(1,21) (NCONEC(I,J),J=2,NCONEC(I,1)+1)
         READ(1,*) FORCE(I)
```

```
   20 CONTINUE
   21 FORMAT (20I3)
C
C
C
C
C     READ BOUNDARY PARAMETERS
C
C     Surface number,node number,element number
      DO 25 I=1,NBOUND
         READ(1,*) NBKIND(I,1),NBKIND(I,2),NBKIND(I,3)
   25 CONTINUE
C
C
C     READ BOUNDARY CONNECTIVITIES
C
C
      DO 30 I=1,NBOUND
         READ(1,31) (NBOUN(I,J),J=1,NBKIND(I,2))
   30 CONTINUE
   31 FORMAT(8I3)
C
C
C     READ BOUNDARY CONSTANTS
C
C     Conduction coef.,convection coef.
      DO 40 I=1,NBOUND
         READ (1,*) CBOUN(I,1),CBOUN(I,2),CBOUN(I,3)
   40 CONTINUE
C
C
C     READ THE CONDUCTION COEFFICIENT OF THE DOMAIN
C
C
      READ(1,*) COEF
C
C     PRESPECIFIED NODAL TEMPERATURES
C
      DO 43 I=1,N1
      TEMP(I)=0.D0
   43 CONTINUE
C
C
C     * NTEMP= Number of nodes where temperature is already specified
C     * ITEMP= Node number
C
C
      DO 44 I=1,N1
         GLOAD(I)=0.D0
         GTEMP(I)=0.D0
      DO 44 I1=1,N1
         GSTIF(I,I1)=0.D0
   44 CONTINUE
C
C
      READ(1,*) NTEMP
      DO 45 I=1,NTEMP
      READ(1,*) ITEMP,TEMP(ITEMP)
      GSTIF(ITEMP,ITEMP)=PENALT
   45 CONTINUE
```

```
C
C
C
      READ(1,*) C1,C2,C3,C4,C5,C6,C7
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C
      DO 46 I=1,NNODES
         TEMPAN(I)=C1+ COOR(I,1)*(C2+C5*COOR(I,1))
     >                + COOR(I,2)*(C3+C6*COOR(I,2))
     >                + COOR(I,3)*(C4+C7*COOR(I,3))
   46 CONTINUE
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C
      DO 50 I=1,N1
         GLOAD(I)=PENALT*TEMP(I)
         GTEMP(I)=0.D0
   50 CONTINUE
C
C
C
C
C     CALCULATE THE GLOBAL STIFFNESS MATRIX
C
C
C
      DO 100 I=1,NELEM
C
      CALL ELEM(COOR,NCONEC,STIF,COEF,I,N1,N2,N3,N4,W,Z,BLOAD,
     >          FORCE,SHAP)
C
C
C
C
C
      DO 60 J=1,NCONEC(I,1)
      IY=NCONEC(I,J+1)
      GLOAD(IY)=GLOAD(IY)+BLOAD(J)
C
      DO 59 K=1,NCONEC(I,1)
C
      IX=NCONEC(I,K+1)
C
      GSTIF ( IY,IX ) = GSTIF ( IY,IX ) + STIF ( J,K ,
C
C
  59  CONTINUE
  60  CONTINUE
 100  CONTINUE
C
C     CALCULATE THE GLOBAL LOAD VECTOR
C
      DO 110 I=1,NBOUND
             N=NBKIND(I,1)
         INODE=NBKIND(I,2)
         IELEM=NBKIND(I,3)
C
C
C
      IF(N.EQ.1) CALL BOUNDR(0.D0,COOR,NBOUN,NCONEC,ELOAD,IELEM,INODE,I,
     >                        W,Z,N1,N2,N3,N4,SHAP,ESTIF)
```

```
      IF(N.EQ.2) CALL BOUNDR(1.DO,COOR,NBOUN,NCONEC,ELOAD,IELEM,INODE,I,
     >                      W,Z,N1,N2,N3,N4,SHAP,ESTIF)
      IF(N.EQ.3) CALL BOUNDS(0.DO,COOR,NBOUN,NCONEC,ELOAD,IELEM,INODE,I,
     >                      W,Z,N1,N2,N3,N4,SHAP,ESTIF)
      IF(N.EQ.4) CALL BOUNDS(1.DO,COOR,NBOUN,NCONEC,ELOAD,IELEM,INODE,I,
     >                      W,Z,N1,N2,N3,N4,SHAP,ESTIF)
      IF(N.EQ.5) CALL BOUNDT(0.DO,COOR,NBOUN,NCONEC,ELOAD,IELEM,INODE,I,
     >                      W,Z,N1,N2,N3,N4,SHAP,ESTIF)
      IF(N.EQ.6) CALL BOUNDT(1.DO,COOR,NBOUN,NCONEC,ELOAD,IELEM,INODE,I,
     >                      W,Z,N1,N2,N3,N4,SHAP,ESTIF)
C
C
C
      DO 105 J=1,INODE
         LL=NBOUN(I,J)
       GLOAD(LL)=GLOAD(LL)+(CBOUN(I,1)+CBOUN(I,2)*CBOUN(I,3))*ELOAD(J)
      DO 105 K=1,INODE
         LK=NBOUN(I,K)
       GSTIF(LL,LK)=GSTIF(LL,LK)+CBOUN(I,2)*ESTIF(J,K)
  105 CONTINUE
  110 CONTINUE
C
C
C
C
      CALL DECOMP(NNODES,GSTIF,LU,INDEXR,N1)
      CALL SOLVE(NNODES,LU,GLOAD,GTEMP,INDEXR,N1)
C
C
C
C
C
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C
      WRITE(2,999)
      DO 998 K=1,NNODES
      C(K)=(GTEMP(K)-TEMPAN(K))
      IF(DABS(TEMPAN(K)).LT. 0.00000000000001) THEN
        WRITE(2,1000) K,GTEMP(K),TEMPAN(K),C(K)
      ELSE
        CK=C(K)*100/DABS(TEMPAN(K))
        WRITE(2,1001) K,GTEMP(K),TEMPAN(K),C(K),CK
      ENDIF
  998 CONTINUE
C
  999 FORMAT('1',//,2X,' NODE ',5X,' NUMERICAL ',5X,
     >' ANALYTICAL ',5X,'  ERROR ',5X,' % ERROR ',/,2X,
     >                 '------',5X,'-----------',5X,
     >'------------',5X,' -------',5X,'---------',//)
 1000 FORMAT (3X,I3,6X,D12.5,5X,D12.5,4X,F9.4,8X,' ***')
 1001 FORMAT (3X,I3,6X,D12.5,5X,D12.5,4X,F9.4,5X,F9.4,' %')
C
C!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
C
C
C
      STOP
      END
C
C
```

```
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE ELEM(COOR,NCONEC,STIF,COEF,IJK,N1,N2,N3,N4,W,Z,
     >                BLOAD,FORCE,SHAP)
C
C     THIS PROGRAM CALCULATES THE STIFFNES MATRIX FOR 8/12/20
C     NODED FINITE ELEMENTS
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COOR(N1,3),W(6),Z(6),STIF(N3,N3),SHAPE(3,20)
      DIMENSION SJ(3,3),DJ(3,3),ST(3,20),NCONEC(N4,N3+1)
      DIMENSION BLOAD(20),FORCE(N4),SHAP(N1)
C############################
      DIMENSION TRAN(3,3)
C############################
C
      R=0.D0
      S=0.D0
      T=0.D0
C
      DO 20 I=1,NCONEC(IJK,1)
            BLOAD(I)=0.D0
      DO 20 J=1,NCONEC(IJK,1)
            STIF(I,J)=0.D0
   20 CONTINUE
C
C############################################
      CALL TRANS(IJK,N1,N3,N4,COOR,NCONEC,TRAN)
C############################################
C
C
      DO 1000 I=1,6
              R=0.5D0*(1.D0+Z(I))
      DO 900 J=1,6
              S=0.5D0*(1.D0+Z(J))
      DO 800 K=1,6
              T=0.5D0*(1.D0+Z(K))
C
      IF(NCONEC(IJK,1).EQ.8)  CALL ELEM08(R,S,T,SHAPE)
      IF(NCONEC(IJK,1).EQ.12) CALL ELEM12(R,S,T,SHAPE)
      IF(NCONEC(IJK,1).EQ.20) CALL ELEM20(R,S,T,SHAPE)
C
C
      IF(FORCE(IJK).EQ.0.D0) GOTO 29
      NC=NCONEC(IJK,1)
      IF(NC.EQ.8)  CALL SHAP08(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(NC.EQ.12) CALL SHAP12(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(NC.EQ.20) CALL SHAP20(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
C
C
C
   29 DO 30 II=1,3
      DO 30 JJ=1,3
           SJ(II,JJ)=0.D0
      DO 30 KK=1,NCONEC(IJK,1)
           IA=NCONEC(IJK,KK+1)
C
           SJ(II,JJ)=SJ(II,JJ)+SHAPE(II,KK)*COOR(IA,JJ)
   30 CONTINUE
C
```

```
C###########################################
      CALL MULTIP(TRAN,SJ)
C###########################################
C
      DET=SJ(1,1)*(SJ(2,2)*SJ(3,3)-SJ(3,2)*SJ(2,3))
      DET=DET+SJ(1,3)*(SJ(2,1)*SJ(3,2)-SJ(3,1)*SJ(2,2))
      DET=DET-SJ(1,2)*(SJ(2,1)*SJ(3,3)-SJ(3,1)*SJ(2,3))
C
C
      DJ(1,1)=(SJ(2,2)*SJ(3,3)-SJ(3,2)*SJ(2,3))/DET
      DJ(2,2)=(SJ(1,1)*SJ(3,3)-SJ(3,1)*SJ(1,3))/DET
      DJ(3,3)=(SJ(1,1)*SJ(2,2)-SJ(1,2)*SJ(2,1))/DET
      DJ(1,2)=(SJ(3,2)*SJ(1,3)-SJ(1,2)*SJ(3,3))/DET
      DJ(1,3)=(SJ(1,2)*SJ(2,3)-SJ(2,2)*SJ(1,3))/DET
      DJ(2,1)=(SJ(3,1)*SJ(2,3)-SJ(2,1)*SJ(3,3))/DET
      DJ(2,3)=(SJ(2,1)*SJ(1,3)-SJ(1,1)*SJ(2,3))/DET
      DJ(3,1)=(SJ(2,1)*SJ(3,2)-SJ(3,1)*SJ(2,2))/DET
      DJ(3,2)=(SJ(3,1)*SJ(1,2)-SJ(1,1)*SJ(3,2))/DET
C
C
      DO 40 III=1,3
      DO 40 JJJ=1,NCONEC(IJK,1)
            ST(III,JJJ)=0.D0
      DO 40 KKK=1,3
            ST(III,JJJ)=ST(III,JJJ)+DJ(III,KKK)*SHAPE(KKK,JJJ)
   40 CONTINUE
C
C
      DO 50 I1=1,NCONEC(IJK,1)
      IRZ=NCONEC(IJK,I1+1)
      BLOAD(I1)=BLOAD(I1)+W(I)*W(J)*W(K)*FORCE(IJK)*SHAP(IRZ)*
     >          DET/8.D0
      DO 50 J1=1,NCONEC(IJK,1)
            ALPHA=0.D0
      DO 60 K1=1,3
            ALPHA=ALPHA+ST(K1,I1)*ST(K1,J1)
   60 CONTINUE
C
            ALPHA=ALPHA*COEF*DET/8.D0
C
            STIF(I1,J1)=STIF(I1,J1)+W(I)*W(J)*W(K)*ALPHA
C
   50 CONTINUE
  800 CONTINUE
  900 CONTINUE
 1000 CONTINUE
C
C
      RETURN
C
      END
C
C
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE BOUNDR(R,COOR,NBOUN,NCONEC,ELOAD,IJK,INODE,IB,
     >                  W,Z,N1,N2,N3,N4,SHAP,ESTIF)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COOR(N1,3),W(6),Z(6),SHAPE(3,20)
```

```
      DIMENSION SJ(3,3),SHAP(N1),NBOUN(N2,8),NCONEC(N4,N3+1)
      DIMENSION ELOAD(8),ESTIF(8,8)
C################################
      DIMENSION TRAN(3,3)
C################################
C
      M1=NCONEC(IJK,1)
      DO 10 I=1,8
      ELOAD(I)=0.D0
      DO 10 I1=1,8
      ESTIF(I,I1)=0.D0
   10 CONTINUE
C
C############################################
      CALL TRANS(IJK,N1,N3,N4,COOR,NCONEC,TRAN)
C############################################
C
C
      DO 50 J= 1,6
            S=0.5D0*(1.D0+Z(J))
      DO 40 K= 1,6
            T=0.5D0*(1.D0+Z(K))
C
C
      IF(M1.EQ.8)CALL SHAP08(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.8)CALL ELEM08(R,S,T,SHAPE)
C
      IF(M1.EQ.12)CALL SHAP12(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.12)CALL ELEM12(R,S,T,SHAPE)
C
      IF(M1.EQ.20)CALL SHAP20(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.20)CALL ELEM20(R,S,T,SHAPE)
C
      DO 20 II=1,3
      DO 20 JJ=1,3
          SJ(II,JJ)=0.D0
      DO 20 KK=1,M1
         IA=NCONEC(IJK,KK+1)
          SJ(II,JJ)=SJ(II,JJ)+SHAPE(II,KK)*COOR(IA,JJ)
   20 CONTINUE
C##############################
      CALL MULTIP(TRAN,SJ)
C##############################
C
C
      DET=(SJ(2,2)*SJ(3,3)-SJ(3,2)*SJ(2,3))
C
C
      DO 30 I=1,INODE
         NIJ=NBOUN(IB,I)
C
         ELOAD(I)=ELOAD(I)+SHAP(NIJ)*DET*W(J)*W(K)/4.D0
      DO 30 L=1,INODE
         NIK=NBOUN(IB,L)
C
         ESTIF(I,L)=ESTIF(I,L)+SHAP(NIJ)*SHAP(NIK)*DET*
     >              W(J)*W(K)/4.D0
   30 CONTINUE
C
   40 CONTINUE
```

```
   50 CONTINUE
C
C
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE BOUNDS(S,COOR,NBOUN,NCONEC,ELOAD,IJK,INODE,IB,
     >                  W,Z,N1,N2,N3,N4,SHAP,ESTIF)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COOR(N1,3),W(6),Z(6),SHAPE(3,20)
      DIMENSION SJ(3,3),SHAP(N1),NBOUN(N2,8),NCONEC(N4,N3+1)
      DIMENSION ELOAD(8),ESTIF(8,8)
C###################################
      DIMENSION TRAN(3,3)
C###################################
C
      M1=NCONEC(IJK,1)
      DO 10 I=1,8
      ELOAD(I)=0.D0
      DO 10 I1=1,8
      ESTIF(I,I1)=0.D0
   10 CONTINUE
C
C###############################################
      CALL TRANS(IJK,N1,N3,N4,COOR,NCONEC,TRAN)
C###############################################
C
C
C
C
      DO 50 J= 1,6
              R=0.5D0*(1.D0+Z(J))
      DO 40 K= 1,6
              T=0.5D0*(1.D0+Z(K))
C
C
      IF(M1.EQ.8)CALL SHAP08(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.8)CALL ELEM08(R,S,T,SHAPE)
C
      IF(M1.EQ.12)CALL SHAP12(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.12)CALL ELEM12(R,S,T,SHAPE)
C
      IF(M1.EQ.20)CALL SHAP20(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.20)CALL ELEM20(R,S,T,SHAPE)
C
      DO 20 II=1,3
      DO 20 JJ=1,3
           SJ(II,JJ)=0.D0
      DO 20 KK=1,M1
         IA=NCONEC(IJK,KK+1)
           SJ(II,JJ)=SJ(II,JJ)+SHAPE(II,KK)*COOR(IA,JJ)
   20 CONTINUE
C
C####################################
       CALL MULTIP(TRAN,SJ)
C####################################
C
C
```

```
C
       DET=-SJ(1,3)*SJ(3,1)+SJ(1,1)*SJ(3,3)
C
C
      DO 30 I=1,INODE
         NIJ=NBOUN(IB,I)
C
         ELOAD(I)=ELOAD(I)+SHAP(NIJ)*DET*W(J)*W(K)/4.DO
      DO 30 L=1,INODE
         NIK=NBOUN(IB,L)
C
         ESTIF(I,L)=ESTIF(I,L)+SHAP(NIJ)*SHAP(NIK)*DET*
     >               W(J)*W(K)/4.DO
   30 CONTINUE
C
   40 CONTINUE
   50 CONTINUE
C
C
C
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE BOUNDT(T,COOR,NBOUN,NCONEC,ELOAD,IJK,INODE,IB,
     >                  W,Z,N1,N2,N3,N4,SHAP,ESTIF)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION COOR(N1,3),W(6),Z(6),SHAPE(3,20)
      DIMENSION SJ(3,3),SHAP(N1),NBOUN(N2,8),NCONEC(N4,N3+1)
      DIMENSION ELOAD(8),ESTIF(8,8)
C################################
      DIMENSION TRAN(3,3)
C################################
C
      M1=NCONEC(IJK,1)
      DO 10 I=1,8
      ELOAD(I)=0.DO
      DO 10 I1=1,8
      ESTIF(I,I1)=0.DO
   10 CONTINUE
C
C##############################################
      CALL TRANS(IJK,N1,N3,N4,COOR,NCONEC,TRAN)
C##############################################
C
C
C
      DO 50 J= 1,6
              R=0.5DO*(1.DO+Z(J))
      DO 40 K= 1,6
              S=0.5DO*(1.DO+Z(K))
C
C
      IF(M1.EQ.8)CALL SHAP08(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.8)CALL ELEM08(R,S,T,SHAPE)
C
      IF(M1.EQ.12)CALL SHAP12(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.12)CALL ELEM12(R,S,T,SHAPE)
C
```

```
      IF(M1.EQ.20)CALL SHAP20(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
      IF(M1.EQ.20)CALL ELEM20(R,S,T,SHAPE)
C
      DO 20 II=1,3
      DO 20 JJ=1,3
            SJ(II,JJ)=0.D0
      DO 20 KK=1,M1
         IA=NCONEC(IJK,KK+1)
            SJ(II,JJ)=SJ(II,JJ)+SHAPE(II,KK)*COOR(IA,JJ)
   20 CONTINUE
C
C###############################################
       CALL MULTIP(TRAN,SJ)
C###############################################
C
       DET=SJ(1,1)*SJ(2,2)-SJ(1,2)*SJ(2,1)
C
C
      DO 30 I=1,INODE
         NIJ=NBOUN(IB,I)
C
         ELOAD(I)=ELOAD(I)+SHAP(NIJ)*DET*W(J)*W(K)/4.D0
      DO 30 L=1,INODE
         NIK=NBOUN(IB,L)
C
         ESTIF(I,L)=ESTIF(I,L)+SHAP(NIJ)*SHAP(NIK)*DET*
     >               W(J)*W(K)/4.D0
   30 CONTINUE
C
   40 CONTINUE
   50 CONTINUE
C
C
C
      RETURN
      END
C
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE DECOMP(N,A,LU,INDEXR,NN)
C
C     This routine implements the Gaussian forward elimination
C     algorithm to find the LU decomposition of the N by N matrix
C     A. Partial pivoting is used along with scaling for row
C     equilibration
C
C
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 LU
      DIMENSION A(NN,NN),LU(NN,NN),SCALEF(200),INDEXR(NN)
C
      PTOL=.0000000000000000000000000001
C
      DO 5 I=1,N
         INDEXR(I)=I
         SCALEF(I)=A(I,1)
         DO 4 J=1,N
            LU(I,J)=A(I,J)
            IF( DABS(LU(I,J)) .GT. SCALEF(I)) SCALEF(I)=DABS(LU(I,J))
    4 CONTINUE
```

```
C
C
      IF (SCALEF(I).NE.0.D0) GO TO 5
         WRITE (2,*) 'ALL ZERO ROW'
         RETURN
    5 CONTINUE
C
C
      NM1=N-1
      DO 50 K=1,NM1
         BIG=DABS(LU(K,K)/SCALEF(K))
         IBIG=K
         KP1=K+1
         DO 10 IR=KP1,N
            IF(DABS(LU(IR,K)/SCALEF(IR)).LE.BIG) GOTO 10
            BIG=DABS(LU(IR,K)/SCALEF(IR))
            IBIG=IR
   10    CONTINUE
C
C
         IF(BIG.GT.PTOL) GOTO 12
         WRITE(2,*) 'SMALL PIVOT'
         RETURN
   12    IF(IBIG.EQ.K) GOTO 16
C
C
         ISAVE=INDEXR(K)
         INDEXR(K)=INDEXR(IBIG)
         INDEXR(IBIG)=ISAVE
         SAVE=SCALEF(K)
         SCALEF(K)=SCALEF(IBIG)
         SCALEF(IBIG)=SAVE
C
C
         DO 15 J=1,N
            SAVE=LU(IBIG,J)
            LU(IBIG,J)=LU(K,J)
            LU(K,J)=SAVE
   15    CONTINUE
C
C
   16    DO 30 I=KP1,N
C
            IF(LU(I,K).EQ.0.D0) GO TO 30
            LU(I,K)=LU(I,K)/LU(K,K)
            DO 20 J=KP1,N
               LU(I,J)=LU(I,J)-LU(I,K)*LU(K,J)
   20       CONTINUE
   30    CONTINUE
   50 CONTINUE
C
C
      IF(DABS(LU(N,N)/SCALEF(N)).GT.PTOL) GO TO 60
      WRITE(2,*) 'SMALL PIVOT'
   60 RETURN
      END
C
C
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
```

```
C
      SUBROUTINE SOLVE(N,LU,B,X,INDEXR,NN)
C
C     This routine solves the linear equations with right-hand
C     side vector B using the LU decomposition matrices already
C     computed by routine decomp. The final solution is stored
C     in X array.
C
C

      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 LU
      DIMENSION LU(NN,NN),B(NN),X(NN),INDEXR(NN)
C
C
      DO 1 I=1,N
         X(I)=B(INDEXR(I))
    1 CONTINUE
C
C
      NM1=N-1
      DO 3 I=2,N
         IM1=I-1
         DO 2 J=1,IM1
            X(I)=X(I)-LU(I,J)*X(J)
    2    CONTINUE
    3 CONTINUE
C
C
      X(N)=X(N)/LU(N,N)
      DO 5 II=1,NM1
         I=N-II
         IP1=I+1
         DO 4 J=IP1,N
            X(I)=X(I)-LU(I,J)*X(J)
    4    CONTINUE
         X(I)=X(I)/LU(I,I)
    5 CONTINUE
      RETURN
      END
C
C
C     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE ELEM08(R,S,T,SHAPE)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION SHAPE(3,20)
C
      SHAPE(1,1)=(1.D0-S)*(1.D0-T)
      SHAPE(1,2)=S*(1.D0-T)
      SHAPE(1,3)=S*(T-1.D0)
      SHAPE(1,4)=(S-1.D0)*(1.D0-T)
      SHAPE(1,5)=(1.D0-S)*T
      SHAPE(1,6)=S*T
      SHAPE(1,7)=-1.D0*S*T
      SHAPE(1,8)=(S-1.D0)*T
C
C
      SHAPE(2,1)=R*(T-1.D0)
      SHAPE(2,2)=R*(1.D0-T)
```

```
      SHAPE(2,3)=(1.DO-R)*(1.DO-T)
      SHAPE(2,4)=(R-1.DO)*(1.DO-T)
      SHAPE(2,5)=-1.DO*R*T
      SHAPE(2,6)=R*T
      SHAPE(2,7)=(1.DO-R)*T
      SHAPE(2,8)=(R-1.DO)*T
C
C
      SHAPE(3,1)=R*(S-1.DO)
      SHAPE(3,2)=-1.DO*R*S
      SHAPE(3,3)=(R-1.DO)*S
      SHAPE(3,4)=(R-1.DO)*(1.DO-S)
      SHAPE(3,5)=R*(1.DO-S)
      SHAPE(3,6)=R*S
      SHAPE(3,7)=(1.DO-R)*S
      SHAPE(3,8)=(1.DO-R)*(1.DO-S)
C
      RETURN
      END
C
C
C     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE ELEM12(R,S,T,SHAPE)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION SHAPE(3,20)
C
      SHAPE(1,1)=(1.DO-S)*(1.DO-T)
      SHAPE(1,2)=S*(2.DO*T*T-4.DO*R*T+4.DO*R-T-1.DO)
      SHAPE(1,3)=S*(-2.DO*T*T-4.DO*R*T+4.DO*R+5.DO*T-3.DO)
      SHAPE(1,4)=(S-1.DO)*(1.DO-T)
      SHAPE(1,5)=(1.DO-S)*T
      SHAPE(1,6)=S*(2.DO*T*T+4.DO*R*T-3.DO*T)
      SHAPE(1,7)=S*(4.DO*R*T-2.DO*T*T-T)
      SHAPE(1,8)=(S-1.DO)*T
      SHAPE(1,9)=(4.DO-8.DO*R)*S*(1.DO-T)
      SHAPE(1,10)=S*4.DO*T*(1.DO-T)
      SHAPE(1,11)=S*4.DO*T*(T-1.DO)
      SHAPE(1,12)=(4.DO-8.DO*R)*S*T
C
C
      SHAPE(2,1)=R*(T-1.DO)
      SHAPE(2,2)=2.DO*R*T*T-2.DO*R*R*T+2.DO*R*R-R*T-R
      SHAPE(2,3)=2.DO*(R*R+T*T-R*T*T-R*R*T)+5.DO*R*T
     >                -3.DO*T-3.DO*R+1.DO
      SHAPE(2,4)=(R-1.DO)*(1.DO-T)
      SHAPE(2,5)=-1.DO*R*T
      SHAPE(2,6)=2.DO*R*T*T+2.DO*R*R*T-3.DO*R*T
      SHAPE(2,7)=2.DO*R*R*T-2.DO*R*T*T+2.DO*T*T-R*T-T
      SHAPE(2,8)=(R-1.DO)*T
      SHAPE(2,9)=4.DO*R*(1.DO-R)*(1.DO-T)
      SHAPE(2,10)=R*4.DO*T*(1.DO-T)
      SHAPE(2,11)=(1.DO-R)*4.DO*T*(1.DO-T)
      SHAPE(2,12)=4.DO*R*(1.DO-R)*T
C
C
      SHAPE(3,1)=R*(S-1.DO)
      SHAPE(3,2)=S*(4.DO*R*T-2.DO*R*R-R)
      SHAPE(3,3)=S*(-4.DO*R*T-2.DO*R*R+4.DO*T+5.DO*R-3.DO)
      SHAPE(3,4)=(R-1.DO)*(1.DO-S)
```

```
      SHAPE(3,5)=R*(1.DO-S)
      SHAPE(3,6)=S*(4.DO*R*T+2.DO*R*R-3.DO*R)
      SHAPE(3,7)=S*(2.DO*R*R-4.DO*R*T+4.DO*T-R-1.DO)
      SHAPE(3,8)=(1.DO-R)*(1.DO-S)
      SHAPE(3,9)=4.DO*R*(R-1.DO)*S
      SHAPE(3,10)=R*S*(4.DO-8.DO*T)
      SHAPE(3,11)=(1.DO-R)*S*(4.DO-8.DO*T)
      SHAPE(3,12)=4.DO*R*(1.DO-R)*S
C
      RETURN
      END
C     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE ELEM20(R,S,T,SHAPE)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION SHAPE(3,20)
C
      SHAPE(1,1)=(1-S)*(1-T)*(4*R-2*S-2*T-1)
      SHAPE(1,2)=S*(1-T)*(4*R+2*S-2*T-3)
      SHAPE(1,3)=S*(1-T)*(4*R-2*S+2*T-1)
      SHAPE(1,4)=(1-S)*(1-T)*(4*R+2*S+2*T-3)
      SHAPE(1,5)=(1-S)*T*(4*R-2*S+2*T-3)
      SHAPE(1,6)=S*T*(4*R+2*S+2*T-5)
      SHAPE(1,7)=S*T*(4*R-2*S-2*T+1)
      SHAPE(1,8)=(1-S)*T*(4*R+2*S-2*T-1)
      SHAPE(1,9)=4*S*(1-S)*(1-T)
      SHAPE(1,10)=(4-8*R)*S*(1-T)
      SHAPE(1,11)=4*S*(S-1)*(1-T)
      SHAPE(1,12)=(4-8*R)*(1-S)*(1-T)
      SHAPE(1,13)=4*S*(1-S)*T
      SHAPE(1,14)=(4-8*R)*S*T
      SHAPE(1,15)=4*S*(S-1)*T
      SHAPE(1,16)=(4-8*R)*(1-S)*T
      SHAPE(1,17)=(1-S)*4*T*(1-T)
      SHAPE(1,18)=S*4*T*(1-T)
      SHAPE(1,19)=S*4*T*(T-1)
      SHAPE(1,20)=(S-1)*4*T*(1-T)
C
C
      SHAPE(2,1)=R*(1-T)*(-2*R+4*S+2*T-1)
      SHAPE(2,2)=R*(1-T)*(2*R+4*S-2*T-3)
      SHAPE(2,3)=(1-R)*(1-T)*(-2*R+4*S-2*T-1)
      SHAPE(2,4)=(1-R)*(1-T)*(2*R+4*S+2*T-3)
      SHAPE(2,5)=R*T*(-2*R+4*S-2*T+1)
      SHAPE(2,6)=R*T*(2*R+4*S+2*T-5)
      SHAPE(2,7)=(1-R)*T*(-2*R+4*S+2*T-3)
      SHAPE(2,8)=(1-R)*T*(2*R+4*S-2*T-1)
      SHAPE(2,9)=R*(4-8*S)*(1-T)
      SHAPE(2,10)=4*R*(1-R)*(1-T)
      SHAPE(2,11)=(1-R)*(4-8*S)*(1-T)
      SHAPE(2,12)=4*R*(R-1)*(1-T)
      SHAPE(2,13)=R*(4-8*S)*T
      SHAPE(2,14)=4*R*(1-R)*T
      SHAPE(2,15)=(1-R)*(4-8*S)*T
      SHAPE(2,16)=4*R*(R-1)*T
      SHAPE(2,17)=R*4*T*(T-1)
      SHAPE(2,18)=R*4*T*(1-T)
      SHAPE(2,19)=(1-R)*4*T*(1-T)
      SHAPE(2,20)=(R-1)*4*T*(1-T)
C
```

```
C
      SHAPE(3,1)=R*(1-S)*(-2*R+2*S+4*T-1)
      SHAPE(3,2)=R*S*(-2*R-2*S+4*T+1)
      SHAPE(3,3)=(1-R)*S*(2*R-2*S+4*T-1)
      SHAPE(3,4)=(1-R)*(1-S)*(2*R+2*S+4*T-3)
      SHAPE(3,5)=R*(1-S)*(2*R-2*S+4*T-3)
      SHAPE(3,6)=R*S*(2*R+2*S+4*T-5)
      SHAPE(3,7)=(1-R)*S*(-2*R+2*S+4*T-3)
      SHAPE(3,8)=(1-R)*(1-S)*(-2*R-2*S+4*T-1)
      SHAPE(3,9)=R*4*S*(S-1)
      SHAPE(3,10)=4*R*(R-1)*S
      SHAPE(3,11)=(R-1)*4*S*(1-S)
      SHAPE(3,12)=4*R*(R-1)*(1-S)
      SHAPE(3,13)=R*4*S*(1-S)
      SHAPE(3,14)=4*R*(1-R)*S
      SHAPE(3,15)=(1-R)*4*S*(1-S)
      SHAPE(3,16)=4*R*(1-R)*(1-S)
      SHAPE(3,17)=R*(1-S)*(4-8*T)
      SHAPE(3,18)=R*S*(4-8*T)
      SHAPE(3,19)=(1-R)*S*(4-8*T)
      SHAPE(3,20)=(1-R)*(1-S)*(4-8*T)
C
      RETURN
      END
C
C
C     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE SHAP08(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION NCONEC(N4,N3+1),SHAP(N1)
C
C
      SHAP(NCONEC(IJK,2))=R*(1-S)*(1-T)
      SHAP(NCONEC(IJK,3))=R*S*(1-T)
      SHAP(NCONEC(IJK,4))=(1-R)*S*(1-T)
      SHAP(NCONEC(IJK,5))=(1-R)*(1-S)*(1-T)
      SHAP(NCONEC(IJK,6))=R*(1-S)*T
      SHAP(NCONEC(IJK,7))=R*S*T
      SHAP(NCONEC(IJK,8))=(1-R)*S*T
      SHAP(NCONEC(IJK,9))=(1-R)*(1-S)*T
C
      RETURN
      END
C
C     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE SHAP12(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION NCONEC(N4,N3+1),SHAP(N1)
C
      SHAP(NCONEC(IJK,2))=R*(1-S)*(1-T)
      SHAP(NCONEC(IJK,3))=S*(2*R*T*T-2*R*R*T+2*R*R-R*T-R)
      SHAP(NCONEC(IJK,4))=S*(-2*R*T*T-2*R*R*T+2*R*R+2*T*T+5*R*T
     >          -3*T-3*R+1)
      SHAP(NCONEC(IJK,5))=(1-R)*(1-S)*(1-T)
      SHAP(NCONEC(IJK,6))=R*(1-S)*T
      SHAP(NCONEC(IJK,7))=S*(2*R*T*T+2*R*R*T-3*R*T)
```

```
      SHAP(NCONEC(IJK,8))=S*(2*R*R*T-2*R*T*T+2*T*T-R*T-T)
      SHAP(NCONEC(IJK,9))=(1-R)*(1-S)*T
      SHAP(NCONEC(IJK,10))=4*R*(1-R)*S*(1-T)
      SHAP(NCONEC(IJK,11))=R*S*4*T*(1-T)
      SHAP(NCONEC(IJK,12))=(1-R)*S*4*T*(1-T)
      SHAP(NCONEC(IJK,13))=4*R*(1-R)*S*T
C
      RETURN
      END
C
C     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE SHAP20(R,S,T,NCONEC,SHAP,N1,N3,N4,IJK)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION NCONEC(N4,N3+1),SHAP(N1)
C
C
      SHAP(NCONEC(IJK,2))=R*(1-S)*(1-T)*(2*R-2*S-2*T-1)
      SHAP(NCONEC(IJK,3))=R*S*(1-T)*(2*R+2*S-2*T-3)
      SHAP(NCONEC(IJK,4))=(1-R)*S*(1-T)*(-2*R+2*S-2*T-1)
      SHAP(NCONEC(IJK,5))=(1-R)*(1-S)*(1-T)*(1-2*R-2*S-2*T)
      SHAP(NCONEC(IJK,6))=R*(1-S)*T*(2*R-2*S+2*T-3)
      SHAP(NCONEC(IJK,7))=R*S*T*(2*R+2*S+2*T-5)
      SHAP(NCONEC(IJK,8))=(1-R)*S*T*(-2*R+2*S+2*T-3)
      SHAP(NCONEC(IJK,9))=(1-R)*(1-S)*T*(-2*R-2*S+2*T-1)
      SHAP(NCONEC(IJK,10))=R*4*S*(1-S)*(1-T)
      SHAP(NCONEC(IJK,11))=4*R*(1-R)*S*(1-T)
      SHAP(NCONEC(IJK,12))=(1-R)*4*S*(1-S)*(1-T)
      SHAP(NCONEC(IJK,13))=4*R*(1-R)*(1-S)*(1-T)
      SHAP(NCONEC(IJK,14))=R*4*S*(1-S)*T
      SHAP(NCONEC(IJK,15))=4*R*(1-R)*S*T
      SHAP(NCONEC(IJK,16))=(1-R)*4*S*(1-S)*T
      SHAP(NCONEC(IJK,17))=4*R*(1-R)*(1-S)*T
      SHAP(NCONEC(IJK,18))=R*(1-S)*4*T*(1-T)
      SHAP(NCONEC(IJK,19))=R*S*4*T*(1-T)
      SHAP(NCONEC(IJK,20))=(1-R)*S*4*T*(1-T)
      SHAP(NCONEC(IJK,21))=(1-R)*(1-S)*4*T*(1-T)
C
      RETURN
      END
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE MULTIP(TRAN,SJ)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION TRAN(3,3),SJ(3,3),CH(3,3)
C
      DO 10 I=1,3
      DO 10 J=1,3
         CH(I,J)=0.D0
      DO 10 K=1,3
         CH(I,J)=CH(I,J)+TRAN(I,K)*SJ(K,J)
   10 CONTINUE
C
      DO 20 I=1,3
      DO 20 J=1,3
         SJ(I,J)=CH(I,J)
   20 CONTINUE
C
```

```
      RETURN
      END
C
C
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
C
      SUBROUTINE TRANS(I,N1,N3,N4,COOR,NCONEC,TRAN)
C
      IMPLICIT REAL*8(A-H,O-Z)
C
      DIMENSION COOR(N1,3),NCONEC(N4,N3+1),TRAN(3,3)
C
C
      X0=COOR(NCONEC(I,5),1)
      Y0=COOR(NCONEC(I,5),2)
      Z0=COOR(NCONEC(I,5),3)
C
      X1=COOR(NCONEC(I,2),1)-X0
      Y1=COOR(NCONEC(I,2),2)-Y0
      Z1=COOR(NCONEC(I,2),3)-Z0
C
      X2=COOR(NCONEC(I,4),1)-X0
      Y2=COOR(NCONEC(I,4),2)-Y0
      Z2=COOR(NCONEC(I,4),3)-Z0
C
      X3=COOR(NCONEC(I,9),1)-X0
      Y3=COOR(NCONEC(I,9),2)-Y0
      Z3=COOR(NCONEC(I,9),3)-Z0
C
      A1=DSQRT(X1*X1+Y1*Y1+Z1*Z1)
      A2=DSQRT(X2*X2+Y2*Y2+Z2*Z2)
      A3=DSQRT(X3*X3+Y3*Y3+Z3*Z3)
C
      TRAN(1,1)=X1/A1
      TRAN(1,2)=X2/A2
      TRAN(1,3)=X3/A3
C
      TRAN(2,1)=Y1/A1
      TRAN(2,2)=Y2/A2
      TRAN(2,3)=Y3/A3
C
      TRAN(3,1)=Z1/A1
      TRAN(3,2)=Z2/A2
      TRAN(3,3)=Z3/A3
C
      RETURN
      END
C
C
C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (*Leave blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 1991	Final Contractor Report - Jan. 89

4. TITLE AND SUBTITLE

A Novel Approach in Formulation of Special Transition Elements: Mesh Interface Elements

5. FUNDING NUMBERS

WU-505-63-5B
G-NAG3-790

6. AUTHOR(S)

Nesrin Sarigul

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Ohio State University
2036 Neil Avenue, 330 CAE Building
Columbus, Ohio 43210

8. PERFORMING ORGANIZATION REPORT NUMBER

None

9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135-3191

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA CR-189050

11. SUPPLEMENTARY NOTES

Project Manager, C.C. Chamis, Structures Division, NASA Lewis Research Center, (216) 433-3252.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified - Unlimited
Subject Category 39

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

The development of more accurate and efficient advanced methods for solution of singular problems encountered in various branches of mechanics are described. These methods are categorized into three levels. First two levels involve the formulation of a new class of elements called "Mesh Interface Elements" (MIE) to connect meshes of traditional elements either in three dimensions or in three and two dimensions. The finite element formulations are based on the boolean sum and blending operators. The results obtained for (1) the general formulation for heat transfer and structural analysis by including the newly introduce varying material properties at material nodal points, (2) the formulation of mesh interface elements, and (3) a hidden-symbolic computational concept. Validation examples are included from heat transfer and structural response problems. The listing of the computer programs is included in the appendix.

14. SUBJECT TERMS

Boolean sums; Two dimensional; Three-dimensional; Heat transfer; Structural analysis; Validation; Computer program

15. NUMBER OF PAGES

64

16. PRICE CODE

A04

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	

NSN 7540-01-280-5500

U.S. GOVERNMENT PRINTING OFFICE: 1992 - 648-034/40052

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

National Aeronautics and
Space Administration

Lewis Research Center
Cleveland, Ohio 44135

Official Business
Penalty for Private Use $300

FOURTH CLASS MAIL

ADDRESS CORRECTION REQUESTED

U.S.MAIL

Postage and Fees Paid
National Aeronautics and
Space Administration
NASA 451

NASA