# STRESS ANALYSIS OF A U-FRAME STRUCTURE BY THE COUPLING OF BOUNDARY AND FINITE ELEMENT METHODS

By

CHI-AN HONG

Bachelor of Science
National Taiwan University
Taiwan, Republic of China
1978

Master of Science
Oklahoma State University
Stillwater, Oklahoma
1983

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
May, 1986

STRESS ANALYSIS OF A U FRAME-STRUCTURE
BY THE COUPLING OF BOUNDARY AND
FINITE ELEMENT METHODS

Thesis Approved:

Thesis Adviser

Donald R Snethen

A E Kelly

R L Lowery

Norman N Durham
Dean of Graduate College

# ACKNOWLEDGEMENTS

For anything of value that may be gained from this study, I owe much appreciation and gratitude to the following individuals who have offered particular assistance.

Dr. William P. Dawkins, major adviser and chairman of my committee, for his excellent guidance and valuable suggestions, especially during the final phase of the thesis writing.

Dr. Allen E. Kelly, Dr. John P. Lloyd, Dr. Richard L. Lowery, and Dr. Donald R. Snethen for their sound instructions throughout my graduate study, and their assistance and advisement while serving on my committee.

My friends and colleagues, for their friendship and the good times which made my educational experience at Oklahoma State University memorable and pleasant.

This thesis is dedicated to my parents, Mr. and Mrs. Tsun-yuan Hong, for their sacrifice, support, encouragement, and love.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# LIST OF SYMBOLS

| | |
|---|---|
| $[A]$ | resulting matrix of boundary element problem produced after considering displacement and traction boundaries |
| $b_i(S)$ | body force at point S in the $X_i$ direction |
| $\{b\}$ | body force function |
| $\{B\}$ | body force terms of boundary element problem after performing integrations for cells |
| $[B]$ | matrix indicating strain-displacement relationship of quadrilateral elements |
| $[B_J]$ | matirx indicating strain-displacement relationship of joint elements |
| $c_{ij}(Q)$ | boundary integral constant at singular point Q |
| $[c]$ | matrix containing all boundary integral constants at singular points |
| $[c^Q]$ | two by two matrix containing boundary integral constants at singular point Q |
| $[C]$ | elastic matrix |
| $\{d\}$ | nodal displacement vector at an element |
| $\{d\}_{S,N}$ | nodal displacement vector at a joint element in the local coordinate system |
| $\{D\}$ | global nodal displacement vector |
| $F_N$ | normal force acting at joint elements |
| $F_S$ | tangential force acting at joint elements |
| $\{F\}$ | resulting vector of boundary element problem produced after considering displacement and traction boundaries |
| $[F]^T$ | two by four interpolation matrix |
| $G$ | shear modulus |

| | |
|---|---|
| $[G]$ | influnce matrix related to nodal traction vector in the boundary element region |
| $[H]$ | influnce matrix related to nodal displacement vector in the boundary element region |
| $\|J\|$ | Jacobian |
| $k_N$ | joint element stiffness per unit length in the normal direction |
| $k_S$ | joint element stiffness per unit length in the tangential direction |
| $[k]$ | element stiffness matrix |
| $[K]$ | global stiffness matrix |
| $[K_J]_{S,N}$ | joint element stiffness matrix in the local coordinate system |
| $[K_J]_{X,Y}$ | joint element stiffness matrix in the global coordinate system |
| $[K']$ | global stiffness matrix generated from the equivalent finite element approach in solving combined boundary and finite element regions |
| $[M]$ | distribution matrix evaluated over the side of each finite element on the boundary |
| $n$ | direction cosine |
| $[N]^T$ | two by eight interpolation matrix |
| $\{p\}$ | nodal traction vector at an element |
| $\{P\}$ | global nodal traction vector |
| $r$ | distance between load point and field point |
| $\{r\}$ | nodal force vector at an element |
| $\{R\}$ | global nodal force vector |
| $\{R'\}$ | global nodal force vector generated from the equivalent finite element approach |
| $\{sb\}$ | equivalent nodal force vector due to surface and body forces in an element |
| $\{SB\}$ | equivalent nodal force vector due to surface and body forces in the structure |
| $t_j(S)$ | traction at point S in the $X_j$ direction |

| | |
|---|---|
| $T_{ij}(Q,S)$ | fundamental solution of traction at point S in the $X_j$ direction due to a unit force acting at point Q in the $X_i$ direction |
| $\{t\}$ | traction function |
| $[T]$ | two by two matrix containing fundamental solution of tractions |
| $[T']$ | two by two matrix containing fundamental solution of tractions at image point S' due to a unit load acting at actual node Q |
| $[T'']$ | two by two matrix containing fundamental solution of tractions at actual point S due to a unit load acting at image node Q' |
| $u_j(S)$ | displacement at point S in the $X_j$ direction |
| $U_{ij}(Q,S)$ | fundamental solution of displacement at point S in the $X_j$ direction due to a unit force acting at point Q in the $X_i$ direction |
| $\{u\}$ | displacement function |
| $[U]$ | two by two matrix containing fundamental solution of displacements |
| $[U']$ | two by two matrix containing fundamental solution of displacements at image point S' due to a unit load acting at actual node Q |
| $[U'']$ | two by two matrix containing fundamental solution of displacements at actual point S due to a unit load acting at image node Q' |
| $\|V\|$ | scale factor for linear boundary element when numerical integration is employed |
| $\{V\}$ | global body force vector in the finite element region |
| $\{V'\}$ | global body force vector generated from the equivalent finite element approach in solving combined boundary and finite element regions |
| $w$ | weighting coefficient |
| $[W]$ | eight by eight transformation matrix |
| $Y_{kij}$ | components of third order tensor in evaluating internal stresses |

| | |
|---|---|
| $[Y]$ | three by two matrix containing coefficients of third order tensor $Y_{kij}$ |
| $Z_{kij}$ | components of third order tensor in evaluating internal stresses |
| $[Z]$ | three by two matrix containing coefficients of third order tensor $Z_{kij}$ |
| $\delta$ | Kronecker or Dirac delta |
| $\nu$ | Poisson's ratio |
| $\pi_p$ | potential energy of the whole system |
| $\sigma_{ij}(Q)$ | stress at internal point Q |
| $\sigma_{ij}(Q,S)$ | stress at point S due to a unit load acting at point Q |
| $\{\varepsilon\}$ | strain vector |
| $\{\varepsilon_J\}_{S,N}$ | strain vector of a joint element in the local coordinate system |
| $\{\sigma\}$ | stress vector |
| $\{\sigma_J\}_{S,N}$ | stress vector of a joint element in the local coordinate system |
| $[\partial]$ | matrix containing differential operators |

# CHAPTER I

# INTRODUCTION

## 1.1 Purpose

The general purpose of this study is to analyze the elastic behavior of a U-Frame structure by the coupling of boundary and finite element methods and to compare the results with those obtained from the finite element method alone. The stress analysis of a U-Frame structure is a Soil Structure Interaction (SSI) problem where the behavior of the U-Frame structure and the surrounding soil are interdependent. The following are the specific objectives of this study:

1. To develop an SSI computer program for the coupling of boundary and finite element methods which incorporates both symmetrical and unsymmetrical solutions of plane strain problems;

2. To develop another SSI computer program, as general purpose programs STRUDL (1) and NASTRAN (2) cannot simulate the behavior at soil/structure interface, to solve plane strain problems by the finite element method alone for comparison;

3. To examine the accuracy and validity of the coupling of boundary and finite element methods in the stress

analysis of a U-Frame structure;

4. To compare the efficiencies between the coupling of boundary and finite element methods and the finite element methods alone in solving SSI problems.

## 1.2 Description of a U-Frame Structure

A typical U-Frame structure, shown in Figure 1, is a water-filled chamber used to raise or lower ships, barges, or boats from one elevation to another along parts of a canal. It consists of artificial sidewalls, movable water-tight gates at both ends which can be opened or closed as needed, and a water conduit with inlet and outlet valves for letting water in and out of the U-Frame structure by gravity flow (3).

A U-Frame structure is one in which the structure walls are designed to act monolithically with the floor slab. The structure monoliths are normally lightly reinforced. The walls and base are considered to form a continuous frame in the generalized shape of a "U".

## 1.3 Background for the Coupling of Boundary and Finite Element Methods

The boundary element method has been well developed recently and it is now accepted as a general numerical method applicable to a wide range of engineering problems. This method is frequently applied to problems with infinite domains such as wave diffraction, harbor resonance, fluid flow, etc., because it satisfies the conditions at infinity

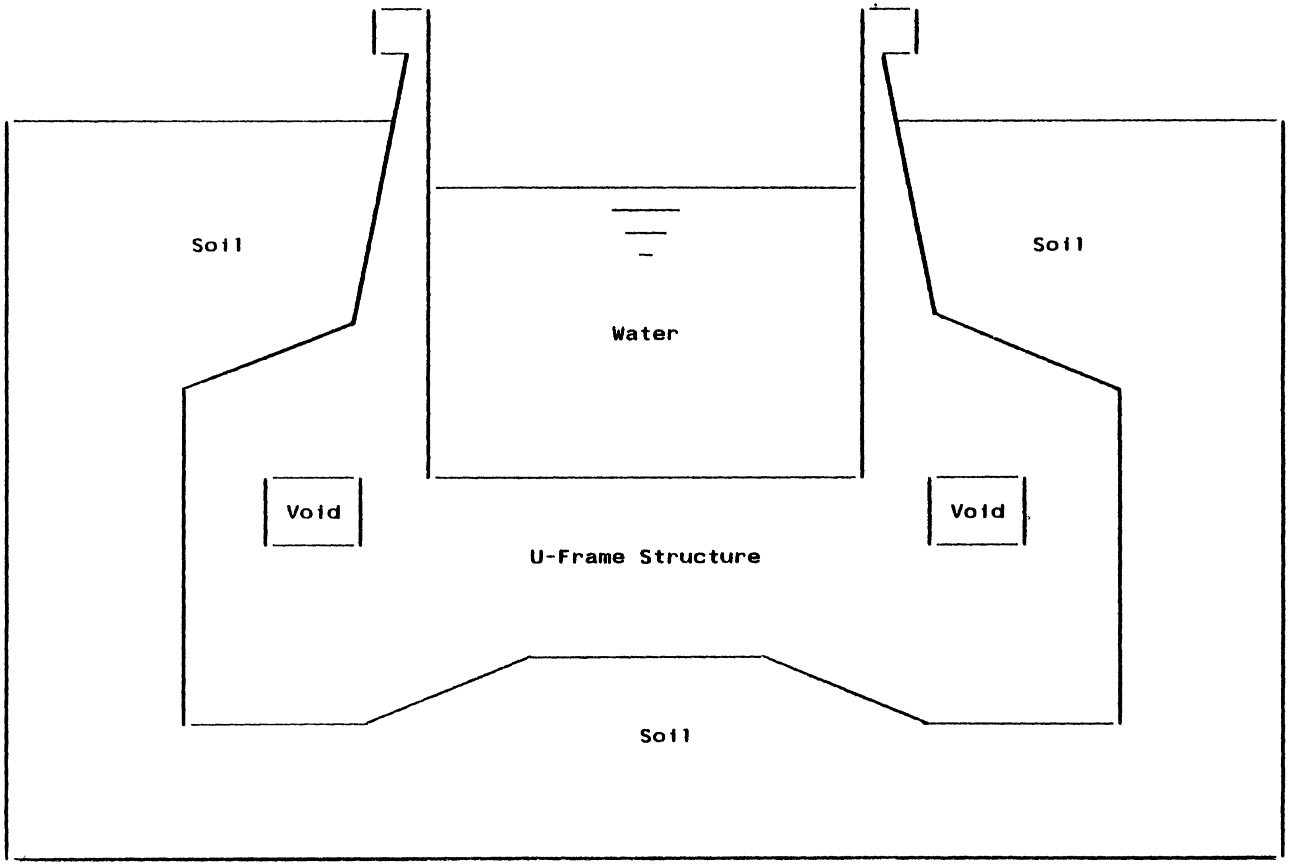


Figure 1. A Typical U-Frame Structure

which are difficult to represent by the finite element method. One of the shortcomings of the finite element method is its inability to model domains extending to infinity. The boundary element method, on the other hand, uses fundamental solutions which naturally satisfy the conditions at infinity.

The idea of the coupling of boundary and finite element methods is of great interest in solving problems with unbounded domains or regions of high stress concentration, both of which can be better represented by using boundary integral solutions. Finite elements, however, may be easier to apply to those parts of the domain which present anisotropic or nonlinear behavior (4).

Osias, Wilson, and Seitelman (5) first combined the boundary and the finite element methods by using boundary integral solutions to represent unbounded wave propagation problems in 1977. Zienkiewicz, Kelly, and Bettes (6) applied the same technique to elastostatics for which the boundary element region is treated as a finite element problem (see section 4.3). The following are the advantages of the coupling of boundary and finite element methods:

1. It allows the use of appropriate conditions to represent infinite domains.

2. It simplifies the required data input to run such computer programs due to a reduction in the dimensionality of the problem.

## 1.4 Methods of Approach

Two methods are used in the stress analysis of a U-Frame structure: the coupling of boundary and finite element methods; and, the finite element method alone. The essential steps of the two methods are as follows.

### 1.4.1 The Coupling of Boundary and Finite Element Methods

1. Use joint elements to simulate the behavior of the interface between the structure and the soil.

2. Discretize the structure into quadrilateral finite elements and assemble all element stiffness matrices (including joint element stiffness matrices) and nodal force vectors to generate a set of simultaneous equations.

3. Discretize the boundaries of the soil mass into linear boundary elements and discretize the soil stratum into cells of integration used in the calculation of body forces; assemble all boundary integral equations to generate another set of simultaneous equations.

4. Combine the equations obtained in steps 2 and 3 to solve for the unknown displacements and tractions. Note that equilibrium and compatability conditions must be satisfied along the interface.

5. Eliminate any tensile stresses present at the interface. This step can be done by iteration.

6. Calculate the stresses and displacements at any point of interest.

### 1.4.2 The Finite Element Method Alone

The required procedures in this method are the same as those in the coupling of boundary and finite element methods except that quadrilateral finite elements are used to model the soil mass in the domain instead of using linear boundary elements to model the boundary.

## 1.5 Limitations and Assumptions

The U-Frame structure and surrounding soil comprise a complex three dimensional system which exhibits nonlinear response to applied loads. For short term loads, however, the response of the system may be assumed to be linear. Except in unusual structures, the U-Frame structure is essentially prismatic. These observations allow the following limitations and assumptions to be imposed to permit the investigation of the structure-soil system by the coupling of boundary and finite element methods:

1. A representative two-dimensional slice of the soil/structure system in a state of plane strain is analyzed.

2. The soil is treated as a linearly elastic, isotropic, homogeneous half space.

3. The U-Frame structure is assumed to be linearly elastic, isotropic, and homogeneous.

4. There is no transfer of tensile stresses across the soil/structure interface. Shear stresses on the interface are assumed to be proportional to the compressive stresses on the interface.

5. Horizontal displacements of the soil surface are assumed to be negligible at a sufficient distance from the structure center line.

6. Vertical displacements are negligible at a sufficient depth below ground surface.

# CHAPTER II

# THE BOUNDARY ELEMENT METHOD

## 2.1 Introduction

The boundary element method is now well developed as a general numerical technique available for the solution of field problems. In contrast with the finite element method, degrees of freedoms only need to be defined on the boundary of the domain of the problem. Once these degrees of freedoms are determined, solutions within the domain are obtained by using appropriate surface/line integrals of the boundary solution.

The main idea of the boundary element method is to generate a system of boundary integral equations which precisely states the problem to be solved in terms of unknown field parameters. Boundary integral equations are generally established by using fundamental solutions of the given problem with the singular point located on the boundary. Therefore, an infinite number of boundary integral equations can be generated. After discretization and numerical integration are performed, the entire boundary of the given problem is first discretized into a finite number of boundary elements in the same manner as it is done in the finite element method; the resulting finite system of boundary integral equations becomes a finite system of algebraic equations suitable for solution with a computer.

The boundary element method is classified into two categories: the direct boundary element method and the indirect boundary element method. In the direct boundary element method, the boundary node unknowns are directly obtained by solving a boundary integral equation. Then, domain unknowns can be computed everywhere with the aid of boundary node values. In the indirect boundary element method, boundary node parameters are used. A boundary integral system has to be solved to compute these parameters which allow the calculation of boundary and domain unknowns.

The direct boundary element method (7) and the indirect boundary element method (8) have been simultaneously developed for elastostatics. These two approaches have provided an effective treatment of practical engineering problems in the last few years. Only the direct boundary element method is derived for plane elasticity in this study.

## 2.2 Boundary Integral Formulation

### 2.2.1 Fundamental Solution

The problem of a concentrated force at a point in an infinite elastic solid is known as Kelvin's problem. Navier's equation is the governing equation of Kelvin's problem, which expresses the equilibrium condition in the infinite domain when a unit load is applied at point Q in the $X_i$ direction. Navier's equation can be written as

$$\sigma_{ij}(Q,S)_{,j} + \delta_i(Q) = 0 \qquad i,\ j = 1,\ 2 \qquad (2.1)$$

where "," denotes partial differentiation when used before a subscript. A repeated subscript implies summation. $\delta$ is Dirac delta. $\delta_i(Q)=1$ if a load acts at point Q in the $X_i$ direction; otherwise, $\delta_i(Q)=0$. S is a field point.

The analytical or fundamental solution of equation 2.1 can be found in reference (9). This solution indicates the displacement $U_{ij}(Q,S)$ and traction $T_{ij}(Q,S)$ at any point S in the infinite domain in the $X_j$ direction due to a unit applied load at point Q in the $X_i$ direction. For two-dimensional plane strain problems,

$$\begin{aligned} U_{ij}(Q,S) &= -\{(3-4\nu)\ln(r)\delta_{ij} - r_{,i}\, r_{,j}\}/\{8\pi(1-\nu)G\} \\ T_{ij}(Q,S) &= -\{[(1-2\nu)\delta_{ij} + 2r_{,i}\, r_{,j}]\frac{\partial r}{\partial n} \qquad (2.2) \\ &\qquad -(1-2\nu)(r_{,i}\, n_j - r_{,j}\, n_i)\}/\{4\pi(1-\nu)r\} \end{aligned}$$

Where G is shear modulus, $\nu$ is Poisson's ratio, $r=r(Q,S)$ is the distance between load point and the field point, $\delta_{ij}=1$ if $i=j$; otherwise, $\delta_{ij}=0$, and $n_i$ is a direction cosine. The fundamental solution and geometric parameters used in equation 2.2 are illustrated in Figure 2.

### 2.2.2 The Reciprocal Theorem

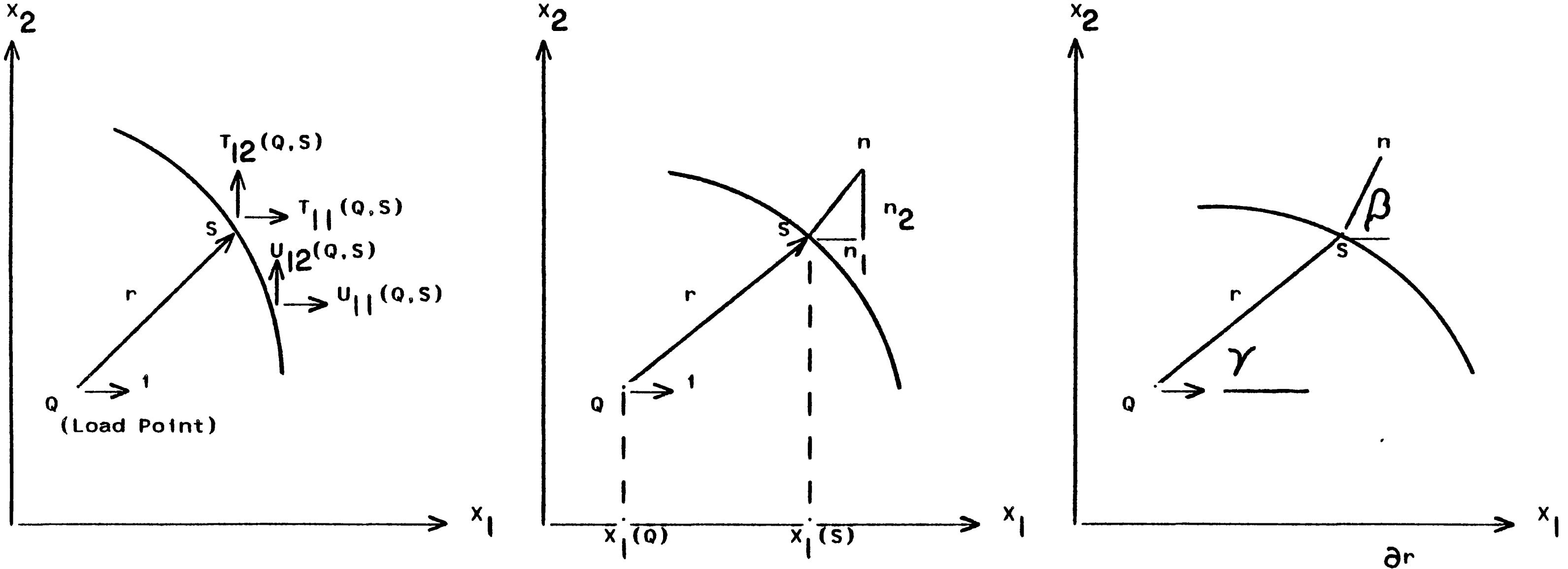
The reciprocal theorem is the key to the direct boundary element method. This theorem links the solutions to two different linear elasticity problems for the same region (10). Suppose that the first problem is characterized by a

$$U_{ij}(Q,S)=-\{(3-4\nu)\delta_{ij}\ln(r)-r,_i r,_j\}/\{8\pi G(1-\nu)\}$$

$$T_{ij}(Q,S)=-\{[(1-2\nu)\delta_{ij}+2r,_i r,_j]\frac{\partial r}{\partial n}-(1-2\nu)(r,_i n_j-r,_j n_i)\}/\{4\pi(1-\nu)r\}$$

$$\frac{\partial r}{\partial x_i}=\frac{x_i(S)-x_i(Q)}{r}=r,_i$$

$$\frac{\partial r}{\partial n}=\cos(\beta-\gamma)$$



(a) Fundamental Solution for 2-D Plane Strain Problems

(b) Partial Derivative $r,_i$ and Direction Cosines $n_i$

(c) Partial Derivative $\frac{\partial r}{\partial n}$

Figure 2. Fundamental Solution and Geometric Parameters

set of displacements {u} and force systems {f} including surface tractions, concentrated forces, and body forces over the region R. Suppose further that the second problem is characterized by another set of displacements {U} and force systems {F} including surface tractions, concentrated forces, and body forces over the same region. According to the reciprocal theorem, the work done by the first set of force systems {f} in moving through the second set of displacements {U} is equal to the work done by the second set of force systems {F} in moving through the first set of displacements {u}.

The statement of the reciprocal theorem provides the basis for the formulation of Somigliana identity and boundary integral equation (11).

### 2.2.3 Somigliana Identity

The Somigliana identity can be used to determine the displacements in the $X_i$ direction at an interior point Q, $u_i(Q)$, of region R once all of the displacements $u_j(S)$ and tractions $t_j(S)$ on the boundary C of region R are known. Imagine that the actual region R with boundary C is mapped into an infinite plane and a unit force is applied at the image of point Q in this plane in the same direction as the displacement to be determined, then all of the displacements $U_{ij}(Q,S)$ and tractions $T_{ij}(Q,S)$ on the auxiliary boundary in this plane can be determined from the fundamental solution (equation 2.2), and the required displacement $u_i(Q)$ can be computed from the reciprocal theorem directly.

The Somigliana identity can be fully explained with reference to Figure 3. Figure 3(a) represents the actual region R with boundary C. All of the displacements $u_j(S)$ and tractions $t_j(S)$ on boundary C are known and body forces $b_j(S)$ are assumed to be prescribed everywhere in region R. Figure 3(b) represents an infinite plane and an auxiliary boundary which is the tracing of the actual boundary onto this plane. $U_{ij}(Q,S)$ and $T_{ij}(Q,S)$ are displacements and tractions on the auxiliary boundary due to a unit force applied at point Q in the $X_i$ direction. Clearly, the only unknown in Figure 3(a) and Figure 3(b) is the displacement $u_i(Q)$. The mathematical statement of the reciprocal theorem in solving for this internal unknown displacement (the Somigliana identity) can be written as

$$u_i(Q)+\int T_{ij}(Q,S)u_j(S)ds=\int U_{ij}(Q,S)t_j(S)ds+\int U_{ij}(Q,S)b_j(S)dA$$

or (2.3)

$$u_i(Q)=\int U_{ij}(Q,S)t_j(S)ds+\int U_{ij}(Q,S)b_j(S)dA-\int T_{ij}(Q,S)u_j(S)ds$$

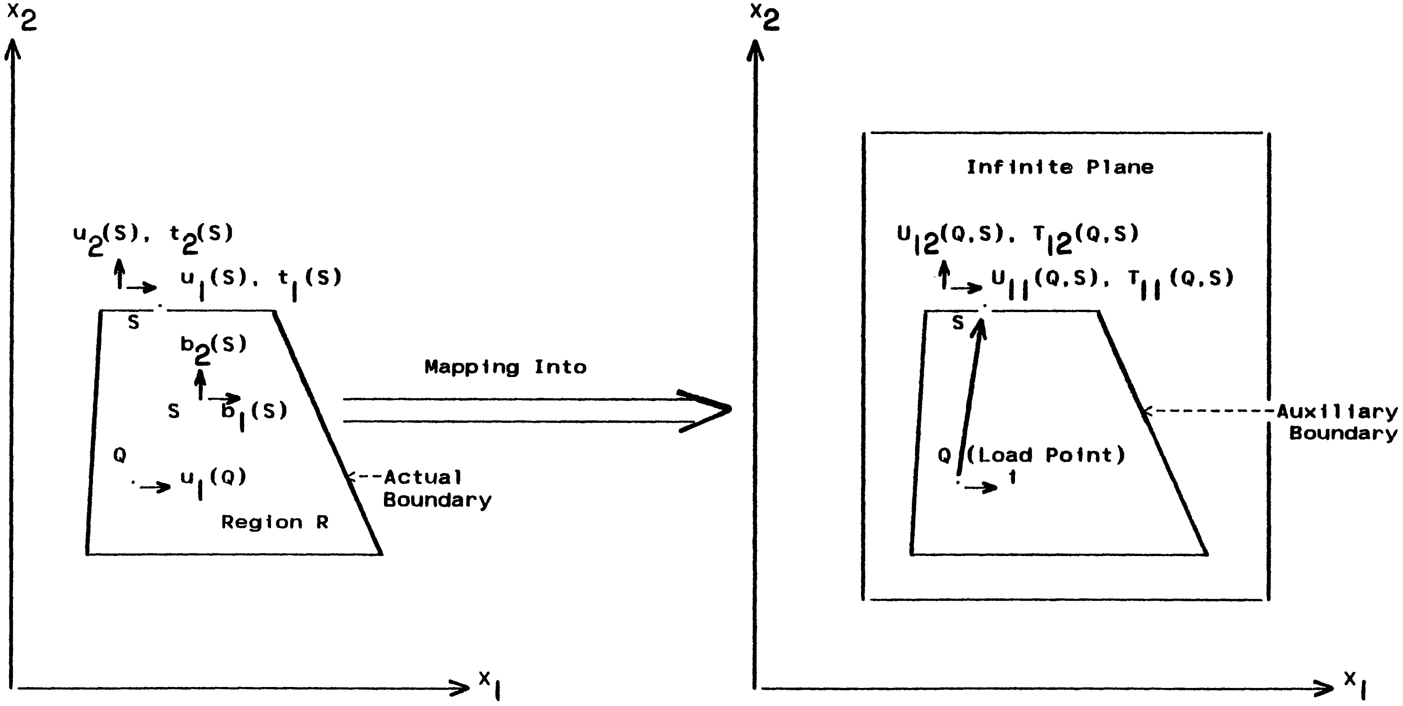
After computing the displacements at any point within the domain under consideration by equation 2.3, the stresses at this specific point can be computed by ordinary strain-displacement and stress-strain relationships. The method to calculate the stresses at an internal point is presented in section 2.3.

### 2.2.4 Boundary Integral equation

The boundary integral equation which is the starting equation of the boundary element method relates the or

$$u_i(Q) + \int T_{ij}(Q,S)u_j(S)ds = \int U_{ij}(Q,S)t_j(S)ds + \int U_{ij}(Q,S)b_j(S)dA$$

$$u_i(Q) = \int U_{ij}(Q,S)t_j(S)ds + \int U_{ij}(Q,S)b_j(S)dA - \int T_{ij}(Q,S)u_j(S)ds$$



(a) Actual Boundary with Known Tractions and Displacements

(b) Auxiliary Boundary and Its Fundamental Solutions

Figure 3. Illustration of Somigliana Identity

unspecified boundary displacements and tractions of the given problem to the specified boundary displacements and tractions plus the solution to another problem for the same region. Considering the Kelvin's problem, the Somigliana identity is not adequate to obtain boundary solutions unless the displacements and tractions on the boundary are known.

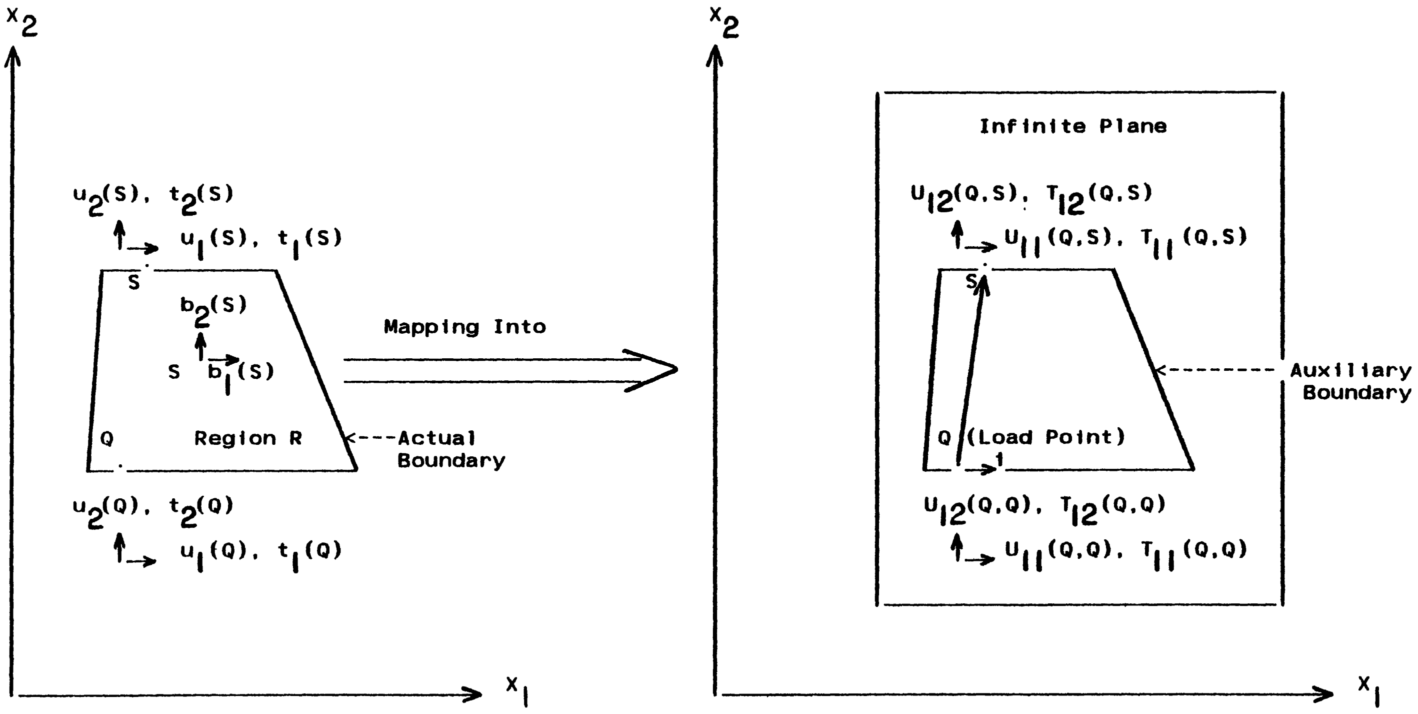
The boundary integral equation can be treated as a limiting case of the Somigliana identity (equation 2.3) as load point Q moves to the boundary. The formulation of the boundary integral equation at singular point Q is shown in Figure 4. Imagine that the actual region R with boundary C is mapped into an infinite plane and a unit force is applied at the image of point Q on the auxiliary boundary in this plane in the $X_i$ direction; then all of the displacements $U_{ij}(Q,S)$ and tractions $T_{ij}(Q,S)$ on the auxiliary boundary can be determined from the fundamental solution (equation 2.2) and the boundary integral equation at point Q can be generated from the reciprocal theorem.

$$\begin{aligned} c_{ij}(Q)u_i(Q)+\int T_{ij}(Q,S)u_j(S)ds &= \int U_{ij}(Q,S)t_j(S)ds \\ &+ \int U_{ij}(Q,S)b_j(S)dA \qquad (2.4) \end{aligned}$$

Where $c_{ij}$ is a constant, $c_{ij}$ is equal to $0.5\delta_{ij}$ for a smooth boundary but generally is different from this value (12). Fortunately explicit calculation of this value is unnecessary as it can be computed by using the rigid body motions explained later.

Boundary integral equations are used to compute all unknown displacements and tractions on the boundary of a

$$c_{ij}(Q)u_i(Q) + \int T_{ij}(Q,S)u_j(S)ds = \int U_{ij}(Q,S)t_j(S)ds + \int U_{ij}(Q,S)b_j(S)dA$$



(a) Actual Boundary

(b) Auxiliary Boundary and Its Fundamental Solutions

Figure 4. Illustration of Boundary Integral Equation

given problem. In fact, because either the displacement or traction is unprescribed at each boundary point in $X_i$ direction and because only one boundary integral equation can be generated at this specific boundary point in this direction, a set of simultaneous equations can be generated and the unknown displacements/tractions on the boundary can be solved since the number of boundary integral equations is equal to the number of unprescribed boundary displacements/ tractions.

## 2.3 Stresses at Internal Points

The Somigliana identity (equation 2.3) is a continuous representation of the displacement at a point Q within region R. Therefore, the stress state at this point can be evaluated by combining the derivatives of equation 2.3 with respect to the coordinates of Q to produce the strain tensor and then substituting the result into generalized Hooke's Law (13). The final expression of the stresses at an internal point Q of a two dimensional isotropic continuum is (14)

$$\sigma_{ij}(Q) = \int Y_{kij} t_k(S)\,ds + \int Y_{kij} b_k(S)\,dA - \int Z_{kij} u_k(S)\,ds \qquad (2.5)$$

where the third order tensor components $Y_{kij}$ and $Z_{kij}$ are

$$Y_{kij} = \{(1-2\nu)[\delta_{ki} r,_j + \delta_{kj} r,_i - \delta_{ij} r,_k] + 2r,_i r,_j r,_k\} \\ /[4\pi(1-\nu) r] \qquad (2.6)$$

$$Z_{kij} = 2\nu\{2\frac{\partial r}{\partial n}[(1-2\nu)\delta_{ij} r,_k + \nu(\delta_{ik} r,_j + \delta_{jk} r,_i) - 4r,_i r,_j r,_k] \\ +2\nu(n_i r,_j r,_k + n_j r,_i r,_k) + (1-2\nu)(2n_k r,_i r,_j + n_i \delta_{jk} \qquad (2.7) \\ +n_j \delta_{ik}) - (1-4\nu) n_k \delta_{ij}\}/[4\pi(1-\nu) r^2]$$

## 2.4 Numerical Implementation

Equations 2.3, 2.4, and 2.5 cannot be solved explicitly as the functions inside the surface/line integrals are very complex. By performing discretization, numerical integration, and special treatment of body forces, the surface/line integrals can be transformed into a finite system of algebraic equations which are the approximate solution of a field problem.

The required steps of numerical implementation for the boundary element method can be summarized as follows:

1. The boundary C is discretized into a series of elements over which displacements and tractions are chosen to be piecewise interpolated between the boundary nodes. The domain R is discretized into a number of cells which are used to calculate the integrals involving body forces.

2. The boundary integral equation (equation 2.4) is applied at each boundary node Q on the boundary C and the integrals are computed numerically over each boundary element. A system of linear algebraic equations are thus generated for a given problem.

3. Boundary conditions are imposed on the boundary C such that the resulting unknown displacements/tractions can be solved from the linear algebraic equations established in step 3.

4. Internal displacements (equation 2.3) and stresses (equation 2.5) for any point of interest can be obtained by numerical integration.

### 2.4.1 Matrix Formulation

For convenience equations 2.3, 2.4, and 2.5 are expressed in matrix form as follows. The boundary integral equation

$$c_{ij}(Q)u_i(Q)+\int T_{ij}(Q,S)u_j(S)ds = \int U_{ij}(Q,S)t_j(S)ds + \int U_{ij}(Q,S)b_j(S)dA \qquad (2.4)$$

can now be expressed in matrix form instead of using indicial notation. The displacements and tractions on the boundary C are expressed as {u} and {t}, and the body forces over the domain R are defined as {b} such that

$$\{u\} = \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix},\ \{t\} = \begin{Bmatrix} t_1 \\ t_2 \end{Bmatrix},\ \{b\} = \begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} \qquad (2.8)$$

The displacements at load point Q are denoted $\{d^Q\}$. In addition, the following three matrices are defined:

$$[U] = \begin{vmatrix} U_{11}(Q,S) & U_{12}(Q,S) \\ U_{21}(Q,S) & U_{22}(Q,S) \end{vmatrix},\ [T] = \begin{vmatrix} T_{11}(Q,S) & T_{12}(Q,S) \\ T_{21}(Q,S) & T_{22}(Q,S) \end{vmatrix}$$
$$[c^Q] = \begin{vmatrix} c_{11}(Q) & 0 \\ 0 & c_{22}(Q) \end{vmatrix} \qquad (2.9)$$

where the coefficients $U_{ij}(Q,S)$ and $T_{ij}(Q,S)$ in matrices [U] and [T] are the fundamental solutions of displacement and traction in the $X_j$ direction due to a unit force acting at node Q in $X_i$ direction. The matrix form of equation 2.4 can be written as

$$[c^Q]\{d^Q\}+\int[T]\{u\}ds = \int[U]\{t\}ds+\int[U]\{b\}dA \qquad (2.10)$$

This formulation is valid for a load point Q on the boundary

C. Note that [T], [U], and {b} are known and the diagonal terms in matrix $[c^Q]$ can be found from the rigid body conditions (see subsection 2.4.3). The unknowns are the unprescribed displacements and tractions over the boundary.

Similarly, internal displacements (equation 2.3) and stresses (equation 2.5) at an interior point Q are written in matrix form as follows. The matrix expression of equation 2.3 is

$$\{d^Q\} = \int [U]\{t\}ds + \int [U]\{b\}dA - \int [T]\{u\}ds \tag{2.11}$$

Where matrices [U] and [T] are already defined in equation 2.9, $\{d^Q\}$ indicates the displacements at internal point Q. The matrix expression of equation 2.5 is

$$\{\sigma^Q\} = \int [Y]\{t\}ds + \int [Y]\{b\}dA - \int [Z]\{u\}ds \tag{2.12}$$

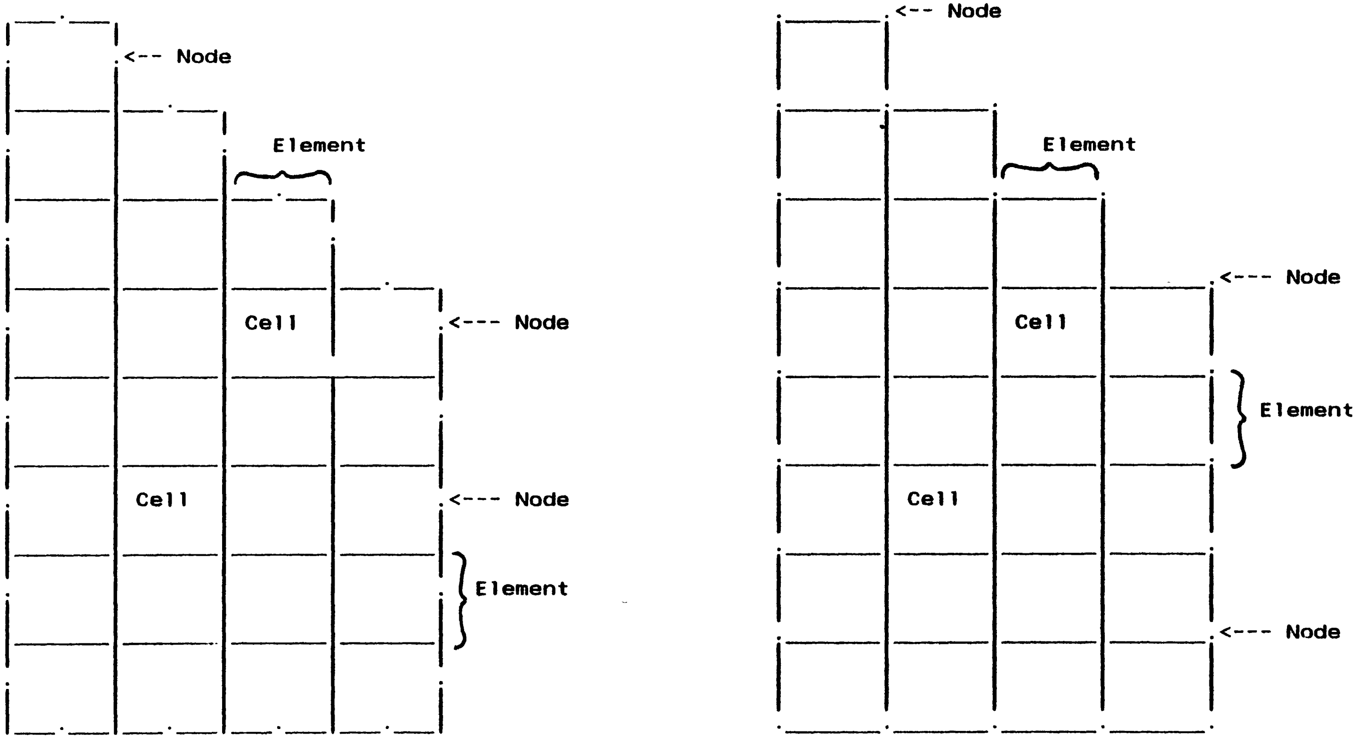
The vector $\{\sigma^Q\}$ and matrices [Y], [Z] are defined as

$$\{\sigma^Q\} = \begin{Bmatrix} \sigma_{11} \\ \sigma_{12} \\ \sigma_{22} \end{Bmatrix}, \quad [Y] = \begin{vmatrix} Y_{111} & Y_{211} \\ Y_{112} & Y_{212} \\ Y_{122} & Y_{222} \end{vmatrix}, \quad [Z] = \begin{vmatrix} Z_{111} & Z_{211} \\ Z_{112} & Z_{212} \\ Z_{122} & Z_{222} \end{vmatrix} \tag{2.13}$$

The terms $Y_{kij}$ and $Z_{kij}$ in matrices [Y] and [Z] can be calculated from equation 2.6 and equation 2.7.

### 2.4.2 Boundary Elements

Discretization is the process of dividing the given boundary into an equivalent system of boundary elements. The boundary elements may be constant, linear, quadratic, or higher order in a two dimensional continuum. Figure 5 is an



(a) Constant Boundary Element (b) Linear Boundary Element

Figure 5: Different Types of Boundary Elements and Internal Cells for Integration

illustration of constant and linear boundary elements. The domain under consideration is divided into a number of cells which are used only for the numerical integration of the body force terms and should not be confused with finite elements. Figure 5(a) shows a constant element where the values of displacement {u} and traction {t} are assumed to be constant over the element, and the boundary node is assumed to be located at the center of the element. Figure 5(b) shows a linear element j where the values of $\{u^j\}$ and $\{t^j\}$ at any point are defined in terms of nodal displacement $\{d^j\}$ and nodal traction $\{p^j\}$ of this element by interpolation functions $F_1$ and $F_2$ such that

$$\{u^j\} = \begin{vmatrix} F_1 & 0 & F_2 & 0 \\ 0 & F_1 & 0 & F_2 \end{vmatrix} \{d^j\} = [F]^T\{d^j\}$$

$$\{t^j\} = \begin{vmatrix} F_1 & 0 & F_2 & 0 \\ 0 & F_1 & 0 & F_2 \end{vmatrix} \{p^j\} = [F]^T\{p^j\} \qquad (2.14)$$

The functions $F_1$ and $F_2$ are given by

$$F_1 = 0.5(1-2\xi/L), \quad F_2 = 0.5(1+2\xi/L) \qquad (2.15)$$

where L is the element length. $\xi$ is the distance between the element centroid and any point of interest.

Linear boundary elements are used to demonstrate the boundary element method. The boundary of a given problem is divided into linear elements and two boundary nodes are placed at the ends of each element. Equation 2.14 is substituted into equation 2.10 to obtain an approximate boundary integral equation for load point Q:

$$[c^Q]\{d^Q\}+\sum_{j=1}^{M}[(\int_{C_j}[T][F]^T ds)\{d^j\}] = \sum_{j=1}^{M}[(\int_{C_j}[U][F]^T ds)\{p^j\}]+\sum_{s=1}^{K}(\int_{A_s}[U]\{b^s\}dA) \tag{2.16}$$

where the summation from j = 1 to M indicates summation over M elements on the boundary and $C_j$ is the boundary of element j. The summation from s = 1 to K is carried out over the internal cells and $A_s$ is the area of cell s.

Applying numerical integration, equation 2.16 becomes (15)

$$[c^Q]\{d^Q\}+\sum_{j=1}^{M}[\{|V|\sum_{r=1}^{L}w_r([T][F]^T)_r\}\{d^j\}] = \sum_{j=1}^{M}[\{|V|\sum_{r=1}^{L}w_r([U][F]^T)_r\}\{p^j\}]+\sum_{s=1}^{K}[|J|\sum_{r=1}^{I}w_r([U]\{b^s\})_r] \tag{2.17}$$

Where L and I are the number of integration points; $w_r$ are weighting coefficients; $([T][F]^T)_r$, $([U][F]^T)_r$, and $([U]\{b^s\})_r$ are the values of the function at the integration points; $|V|$ is a scale factor equal to the half length of the linear boundary element; and $|J|$ is the Jacobian for the internal cell under consideration.

### 2.4.3 System of equations

Equation 2.17 gives two influence equations corresponding to a particular node Q. The evaluation of the body force term at cell s produces a vector $\{B^s\}$. The terms

$$\{|V|\sum_{r=1}^{L}w_r([T][F]^T)_r\} \text{ and } \{|V|\sum_{r=1}^{L}w_r([U][F]^T)_r\}$$

relate the "Q" node with the nodes of the "j" element over which the summation is carried out. These two 2X4 matrices are denoted $[h_{qj}]$ and $[g_{qj}]$, and equation 2.17 can be expressed as two algebraic equations:

$$[c^Q]\{d^Q\}+\sum_{j=1}^{M}([h_{qj}]\{d^j\}) = \sum_{j=1}^{M}([g_{qj}]\{p^j\})+\sum_{s=1}^{K}\{B^s\} \qquad (2.18)$$

Equation 2.18 relates the value of displacement at node Q with the value of displacements and tractions at all the nodes on the boundary, including "Q".

After assemblage of equation 2.18 for each boundary node, a set of simultaneous algebraic equations can be expressed in matrix form as

$$[c]\{D\} + [\bar{H}]\{D\} = [G]\{P\} + \{B\}$$

or (2.19)

$$[H]\{D\} = [G]\{P\} + \{B\}$$

where $[H] = [c] + [\bar{H}]$, and $[c]$ is a diagonal matrix. $\{D\}$ is the nodal displacement vector and $\{P\}$ is a vector of nodal tractions on the boundary.

The diagonal coefficients in matrix [H] can be obtained by applying rigid body conditions (16). For a unit rigid body displacement in any one direction, equation 2.19 becomes

$$[H][I_i] = [0] \qquad (2.20)$$

where $[I_i]$ is a vector defining a unit rigid body displacement in the $X_i$ direction. Hence the diagonal terms of [H] are simply

$$h_{ii} = -\sum_{\substack{i=1\\ i\neq k}}^{2N} h_{ik} \qquad (2.21)$$

where N is the number of boundary nodes. This equation means that the diagonal terms in matrix [c] do not need to be determined explicitly.

As N1 values of displacements and N2 values of tractions are prescribed (N1+N2=2N), after reordering the equations such that the 2N unknowns {X} appear on the left hand side, equation 2.19 can be written as

$$[A]\{X\} = \{F\} + \{B\} \qquad (2.22)$$

### 2.4.4 Internal Displacements and Tractions

The displacements and tractions at an interior point Q can be computed once the nodal displacements {D} and tractions {P} on the boundary are found from equation 2.22.

Assuming the boundary under consideration is discretized by linear elements, the displacements and stresses at node Q can be obtained by substituting equation 2.14 into equations 2.11 and 2.12. Hence the internal displacements at point Q can be expressed as

$$\begin{aligned}\{d^{Q}\} = {} & \sum_{j=1}^{M}[(\int_{C_j}[U][F]^{T}ds)\{p^{j}\}] + \sum_{s=1}^{K}(\int_{A_s}[U]\{b^{s}\}dA) \\ & - \sum_{j=1}^{M}[(\int_{C_j}[T][F]^{T}ds)\{d^{j}\}]\end{aligned} \qquad (2.23)$$

and the internal stresses at point Q can be expressed as

$$\{\sigma^Q\} = \sum_{j=1}^{M}[(\int_{C_j}[Y][F]^T ds)\{p^j\}] + \sum_{s=1}^{K}(\int_{A_s}[Y]\{b^s\}dA) - \sum_{j=1}^{M}[(\int_{C_j}[Z][F]^T ds)\{d^j\}] \tag{2.24}$$

where $\{d^j\}$ and $\{p^j\}$ are the nodal displacements and tractions at element j. The summation from j = 1 to M indicates summation over M elements on the boundary and $C_j$ is the boundary of element j. The summation from s = 1 to K is carried out over the internal cells and $A_s$ is the area of cell s.

Applying numerical integration (usually the Gaussian quadrature scheme), the values of $\{d^Q\}$ in equation 2.23 and the values of $\{\sigma^Q\}$ in equation 2.24 can be determined.

## 2.5 Traction Discontinuity

Special techniques are required to deal with traction discontinuity problems. These problems will arise when the boundary is discretized by linear elements because linear interpolations cannot be applied to the tractions over the element for which a specific boundary node has two different values of tractions.

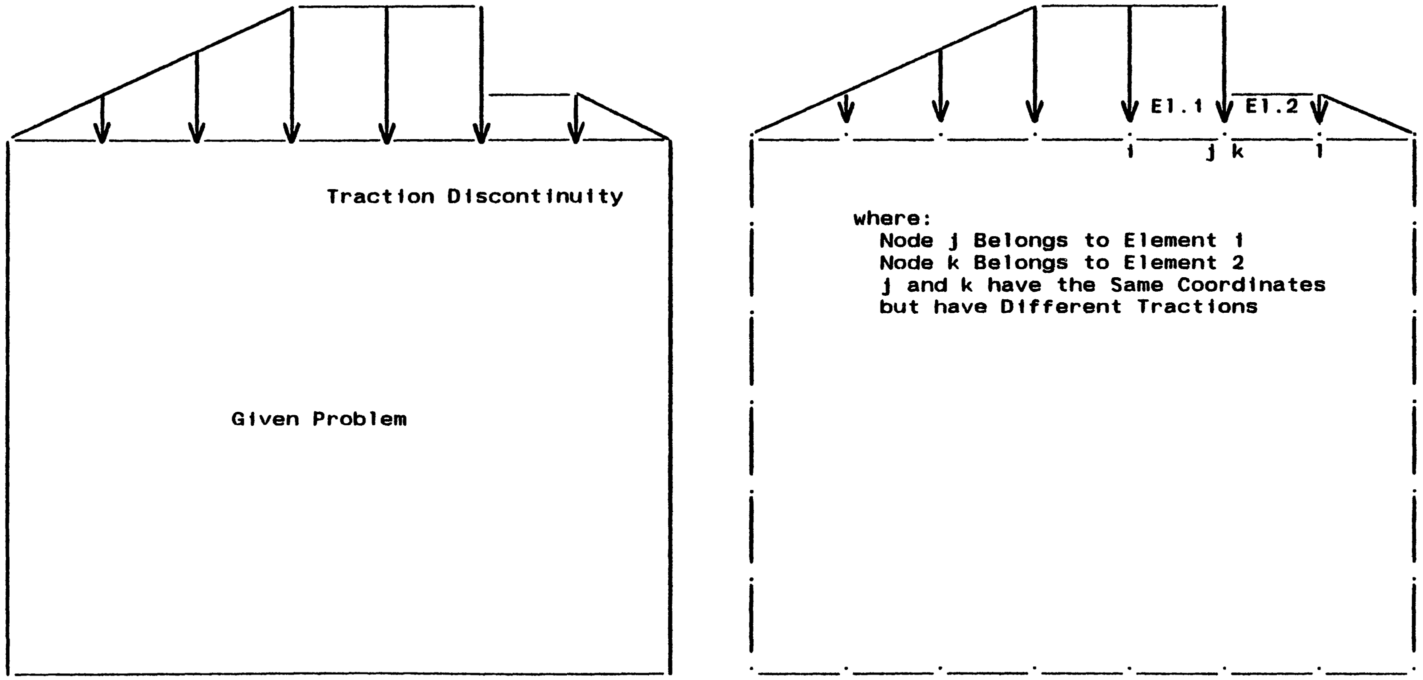
Two concepts have been applied to simulate traction discontinuities over the boundary: the concept of double nodes (17) and the concept of an artificial small element (18).

In the concept of double nodes, two boundary nodes are placed with exactly the same coordinates without any boundary element in between. This approach is illustrated in Figure 6. Figure 6(a) represents traction discontinuities over the boundary in a real problem. Figure 6(b) illustrates the concept of double nodes, which defines the connectivity of the elements where traction discontinuities can be simulated by assigning different values for the tractions at node j, t(j), and node k, t(k). However, this approach has a limitation when both nodes have a prescribed displacement component in the same direction. This condition generates a singular matrix [A] (such possibility violates the displacement continuity condition) (19).

In the concept of an artificial small element, an artificial element is placed to simulate traction discontinuities as shown in figure 7. Figure 7(a) represents the actual distribution of traction discontinuities over the boundary. Figure 7(b) represents the modified distribution of tractions by using an artificial small element where traction discontinuities no longer exist.

## 2.6 Symmetry Conditions

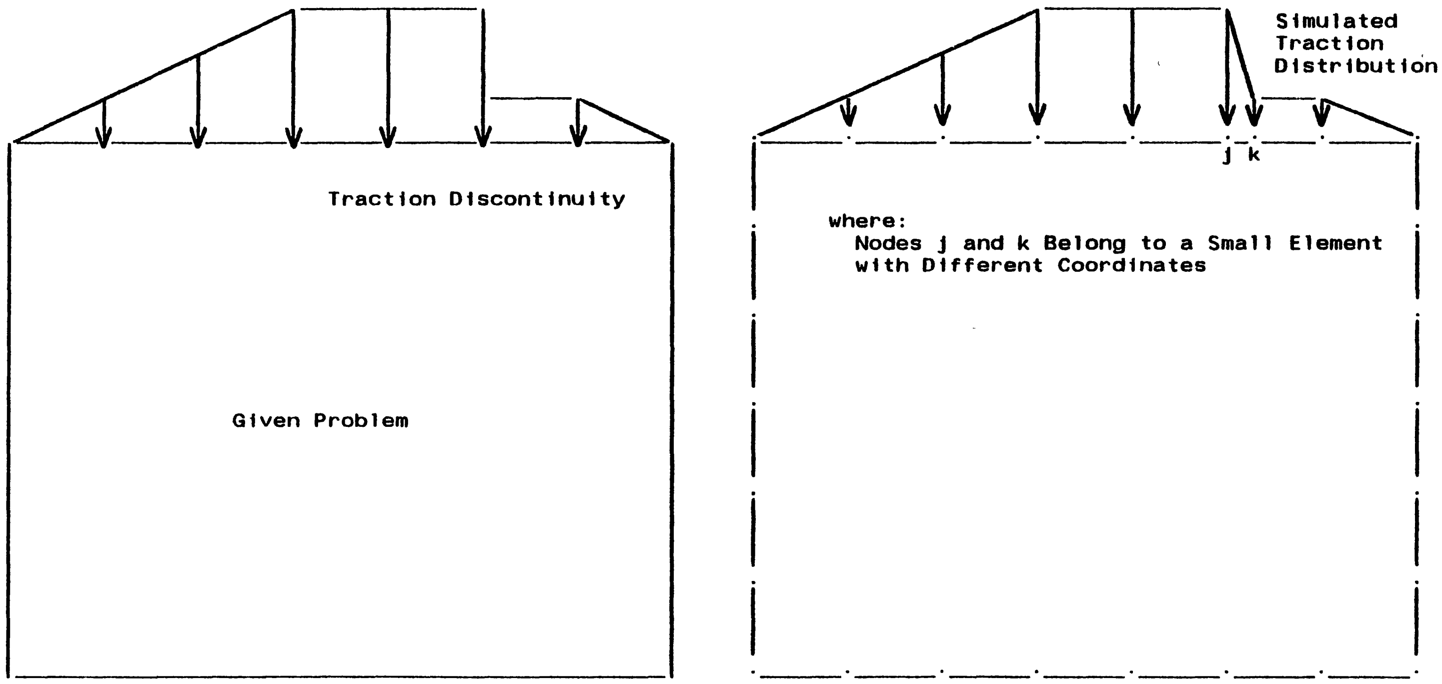
Symmetry about a vertical axis may exist for certain soil-structure interaction problem when the elastic properties of the material, the geometric configuration of the boundaries, and the loading conditions are all symmetrical with respect to the vertical axis. The effects of symmetry cause no horizontal displacements and no shear stresses along the vertical axis.



(a) Traction Discontinuity

(b) Simulation by Double Nodes

Figure 6. The Concept of Double Nodes in Traction Discontinuities



where:
Nodes j and k Belong to a Small Element with Different Coordinates

(a) Traction Discontinuity

(b) Simulation by Small Elements

Figure 7. The Concept of Small Elements in Traction Discontinuities

Two physical consequences should be noted in the incorporation of symmetry conditions in the boundary element method:

1. Only one half of the given problem is analyzed by reflecting the image of nodes, elements, and internal cells at the appropriate location with respect to the line of symmetry.

2. When symmetry conditions are taken into consideration, the contribution of an actual boundary node to the image elements and cells can be replaced by the contribution of an image node to the actual elements and cells. The results of reflection for symmetry about a vertical axis are that the horizontal displacements and shear stresses at the actual and image boundary node/element are always equal in magnitude but opposite in sign and that the vertical displacements and normal stresses are always equal.

In order to explain the incorporation of symmetry about a vertical axis in the boundary element method, equation 2.16 is repeated for completeness:

$$[c^Q]\{d^Q\}+\sum_{j=1}^{M}[(\int_{C_j}[T][F]^T ds)\{d^j\}] = \sum_{j=1}^{M}[(\int_{C_j}[U][F]^T ds)\{p^j\}]+\sum_{s=1}^{K}(\int_{A_s}[U]\{b^s\}dA) \tag{2.16}$$

This equation is the discretized form of the boundary integral equation which relates the value of displacement at node Q with the value of displacements and tractions at all the nodes on the boundary, including "Q".

The consideration of symmetry about the vertical $(X_2)$

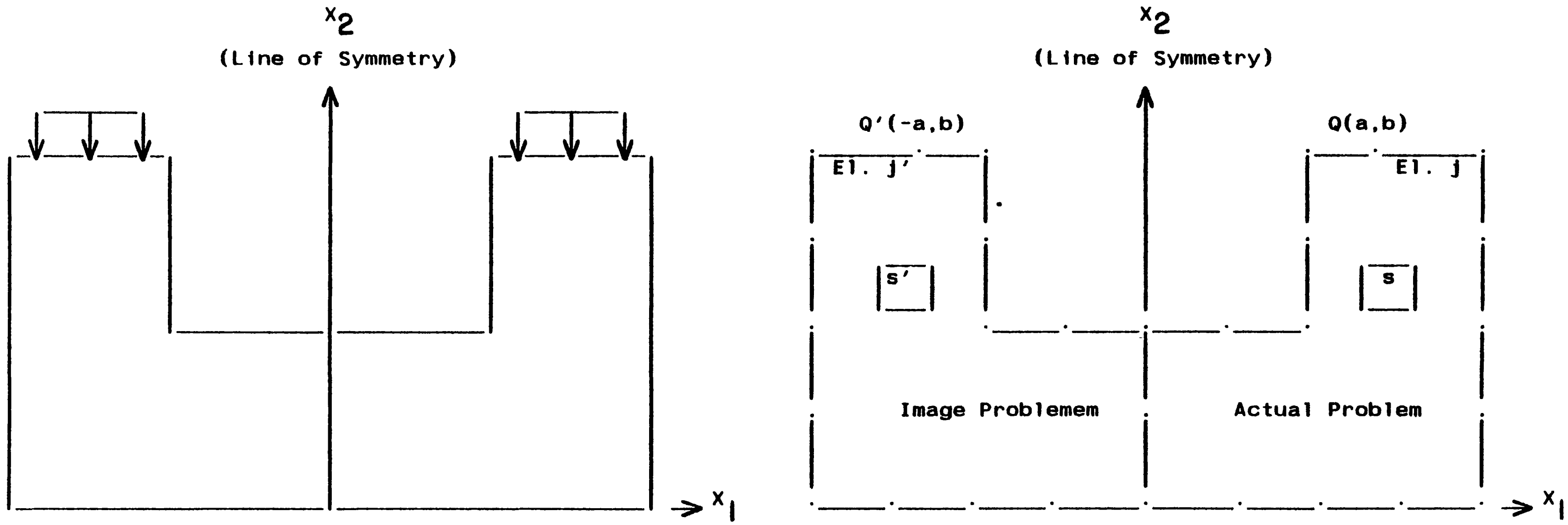
axis is shown in Figure 8. Figure 8(a) shows a given problem which is symmetrical about a vertical axis. This problem can be solved by generating the boundary integral equation in a form such as equation 2.16 without taking the symmetry into account. Figure 8(b) shows the same problem as in Figure 8(a) for which the entire domain is divided into two halves. The one on the right hand side is the actual problem and the other is the image of the actual problem. For each boundary node Q, element j, and internal cell s in the actual problem, an image boundary node Q', element j', and internal cell s' are reflected on the opposite side with respect to the line of symmetry. Due to the influence of this reflection, the coordinates of any image point S can be written in terms of the coordinates of the corresponding actual point S as

$$X_1(S') = -X_1(S), \quad X_2(S') = X_2(S) \tag{2.25}$$

Note that the actual and image nodes (or elements) coincide along the line of symmetry.

Assuming that I and L represent the number of elements and internal cells in the actual and image problem, respectively, several terms in equation 2.16 can be seperated into actual and image parts as follows:

$$\begin{aligned}[c^Q]\{d^Q\} &+ \sum_{j=1}^{I}[(\int_{C_j}[T][F]^T ds)\{d^j\}] + \sum_{j'=1}^{I}[(\int_{C'_j}[T'][F]^T ds)\{d^{j'}\}] \\ &= \sum_{j=1}^{I}[(\int_{C_j}[U][F]^T ds)\{p^j\}] + \sum_{j'=1}^{I}[(\int_{C'_j}[U'][F]^T ds)\{p^{j'}\}] \\ &+ \sum_{s=1}^{L}(\int_{A_s}[U]\{b^s\}dA) + \sum_{s'=1}^{L}(\int_{A'_s}[U']\{b^{s'}\}dA)\end{aligned} \tag{2.26}$$



(a) Given Problem with Symmetry about Vertical Axis

(b) Actual Analyzed Problem and Its Image

Figure 8. Analysis of a Given Problem with Symmetry about Vertical Axis

where the summation from j or j' = 1 to I indicates summation over I actual or image elements. The summation from s or s' = 1 to L is carried out over the actual or image internal cells. The terms

$$\sum_{j'=1}^{I}[(\int_{C_j'}[T'][F]^T ds\{d^{j'}\}],\quad \sum_{j'=1}^{I}[(\int_{C_j'}[U'][F]^T ds\{p^{j'}\}],$$
$$\text{and}\ \sum_{s'=1}^{L}(\int_{A_s'}[U']\{b^{s'}\}dA) \tag{2.27}$$

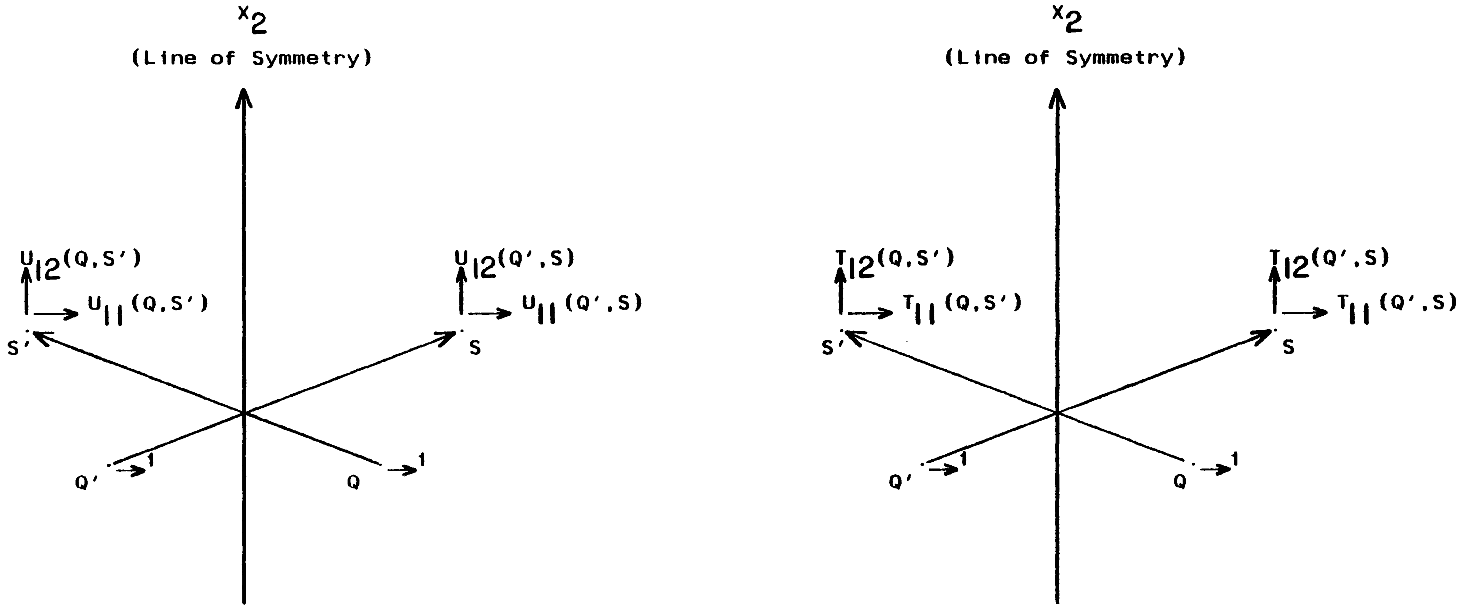
represent the contribution of actual node Q to each image element j' and cell s'.

The remaining task is to seek the image of actual node, Q', such that the terms in equation 2.27 can be replaced by the contribution of Q' to each actual element j and cell s. The mathematical statements are as follows:

$$\sum_{j=1}^{I}[(\int_{C_j}[\bar{T}][F]^T ds)\{d^{j}\}] = \sum_{j'=1}^{I}[(\int_{C_j'}[T'][F]^T ds)\{d^{j'}\}],$$
$$\sum_{j=1}^{I}[(\int_{C_j}[\bar{U}][F]^T ds)\{p^{j}\}] = \sum_{j'=1}^{I}[(\int_{C_j'}[U'][F]^T ds)\{p^{j'}\}], \tag{2.28}$$
$$\text{and}\ \sum_{s=1}^{L}(\int_{A_s}[\bar{U}]\{b^{s}\}dA) = \sum_{s'=1}^{L}(\int_{A_s'}[U']\{b^{s'}\}dA)$$

where $[F]^T$ is the interpolation matrix for both actual and image elements. $[\bar{U}]$ and $[\bar{T}]$ are matrices to be evaluated in terms of $U_{ij}(Q',S)$ and $T_{ij}(Q',S)$.

$U_{ij}(Q',S)$ and $T_{ij}(Q',S)$ are fundamental solutions at actual elements when a unit force acts at image node Q'. The definitions of fundamental solutions $U_{ij}(Q,S')$, $U_{ij}(Q',S)$, $T_{ij}(Q,S')$, and $T_{ij}(Q',S)$ are given in Figure 9.



(a) Fundamental Soluitons of $U_{ij}(Q,S')$ and $U_{ij}(Q',S)$

(b) Fundamental Solutions of $T_{ij}(Q,S')$ and $T_{ij}(Q',S)$

Figure 9. Fundamental Solutions with Respect to Actual and Its Image Nodes

For symmetry about the vertical axis, matrices [U'] and [T'] can be expressed in terms of $U_{ij}(Q',S)$ and $T_{ij}(Q',S)$ by substituting the coordinates of points Q, Q', S, and S' into fundamental solution (equation 2.2) as

$$\begin{array}{ccc} & \text{(acts on image elements)} & \text{(acts on actual elements)} \\ [U'] = & \begin{vmatrix} U_{11}(Q,S'), & U_{12}(Q,S') \\ U_{21}(Q,S'), & U_{22}(Q,S') \end{vmatrix} = & \begin{vmatrix} U_{11}(Q',S), & -U_{12}(Q',S) \\ -U_{21}(Q',S), & U_{22}(Q',S) \end{vmatrix} = [U''] \end{array} \tag{2.29}$$

$$\begin{array}{ccc} & \text{(acts on image elements)} & \text{(acts on actual elements)} \\ [T'] = & \begin{vmatrix} T_{11}(Q,S'), & T_{12}(Q,S') \\ T_{21}(Q,S'), & T_{22}(Q,S') \end{vmatrix} = & \begin{vmatrix} T_{11}(Q',S), & -T_{12}(Q',S) \\ -T_{21}(Q',S), & T_{22}(Q',S) \end{vmatrix} = [T''] \end{array}$$

where point S is located at an actual element and point S' is the image of point S. Equation 2.29 can be substituted for the matrix terms on the right hand side of equation 2.28 to evaluate the summation over actual elements instead of over image elements. Therefore equations 2.28 become

$$\begin{aligned} \sum_{j=1}^{I}[(\int_{C_j}[\bar{T}][F]^T ds)\{d^j\}] &= \sum_{j=1}^{I}[(\int_{C_j}[T''][F]^T ds)\{d^{j'}\}], \\ \sum_{j=1}^{I}[(\int_{C_j}[\bar{U}][F]^T ds)\{p^j\}] &= \sum_{j=1}^{I}[(\int_{C_j}[U''][F]^T ds)\{p^{j'}\}], \\ \text{and} \quad \sum_{s=1}^{L}(\int_{A_s}[\bar{U}]\{b^s\}dA) &= \sum_{s=1}^{L}(\int_{A_s}[U'']\{b^{s'}\}dA) \end{aligned} \tag{2.30}$$

Matrices $[\bar{U}]$ and $[\bar{T}]$ are now ready to be computed once the symmetry condition of nodal displacements and tractions between actual and image elements, and body forces between actual and image cells are considered. For instance, the following relations should be satisfied when the vertical axis is the line of symmetry.

$$\{d^{j'}\} = \begin{Bmatrix} {}^{1}d'_{1} \\ {}^{1}d'_{2} \\ {}^{2}d'_{1} \\ {}^{2}d'_{2} \end{Bmatrix} = \begin{Bmatrix} -{}^{1}d_{1} \\ {}^{1}d_{2} \\ -{}^{2}d_{1} \\ {}^{2}d_{2} \end{Bmatrix}, \quad \{p^{j'}\} = \begin{Bmatrix} {}^{1}p'_{1} \\ {}^{1}p'_{2} \\ {}^{2}p'_{1} \\ {}^{2}p'_{2} \end{Bmatrix} = \begin{Bmatrix} -{}^{1}p_{1} \\ {}^{1}p_{2} \\ -{}^{2}p_{1} \\ {}^{2}p_{2} \end{Bmatrix}, \tag{2.31}$$

$$\text{and} \quad \{b^{s'}\} = \begin{Bmatrix} b'_{1} \\ b'_{2} \end{Bmatrix} = \begin{Bmatrix} -\ b_{1} \\ b_{2} \end{Bmatrix}$$

where superscripts 1 and 2 are the node numbers of an element, and subscripts 1 and 2 indicate directions. Matrices $[\bar{U}]$ and $[\bar{T}]$ can be determined in terms of $U_{ij}(Q',S)$ and $T_{ij}(Q',S)$ by substituting equation 2.31 for the vector terms on the right hand side of equation 2.30. The final results are

$$[\bar{U}] = \begin{vmatrix} -U_{11}(Q',S), -U_{12}(Q',S) \\ U_{21}(Q',S),\ U_{22}(Q',S) \end{vmatrix}, \quad [\bar{T}] = \begin{vmatrix} -T_{11}(Q',S), -T_{12}(Q',S) \\ T_{21}(Q',S),\ T_{22}(Q',S) \end{vmatrix} \tag{2.32}$$

Equations 2.32 indicate that the coefficients in matrices $[\bar{U}]$ and $[\bar{T}]$ can be found with appropriate signs from the fundamental solutions at actual elements when an image node is taken as the load point. Hence the discretized form of the boundary integral equation need be analyzed only on actual elements and cells to obtain effects for actual and image nodes. Then equation 2.26 is reduced to

$$\begin{aligned} &[c^{Q}]\{d^{Q}\} + \sum_{j=1}^{I}[(\int_{C_j}\{[T][F]^{T} + [\bar{T}][F]^{T}\}ds)\{d^{j}\}] = \\ &\sum_{j=1}^{I}[(\int_{C_j}\{[U][F]^{T} + [\bar{U}][F]^{T}\}ds)\{p^{j}\}] + \sum_{s=1}^{L}[\int_{A_s}([U]+[\bar{U}])\{b^{s}\}dA] \end{aligned} \tag{2.33}$$

This equation is the compact form of the boundary integral equation which incorporates the symmetry condition about a vertical axis.

Similarly, under the same symmetric condition, the discretized forms for internal displacements and stresses are expressed as

$$\{d^{Q}\} = \sum_{j=1}^{I}[(\int_{C_j}\{[U][F]^{T}+[\bar{U}][F]^{T}\}ds)\{p^{j}\}] - \sum_{j=1}^{I}[(\int_{C_j}\{[T][F]^{T}+[\bar{T}][F]^{T}\}ds)\{d^{j}\}]+\sum_{s=1}^{L}[\int_{A_s}([U]+[\bar{U}])\{b^{s}\}dA] \qquad (2.34)$$

and

$$\{\sigma^{Q}\} = \sum_{j=1}^{I}[(\int_{C_j}\{[Y][F]^{T}+[\bar{Y}][F]^{T}\}ds)\{p^{j}\}] - \sum_{j=1}^{I}[(\int_{C_j}\{[Z][F]^{T}+[\bar{Z}][F]^{T}\}ds)\{d^{j}\}]+\sum_{s=1}^{L}[\int_{A_s}([Y]+[\bar{Y}])\{b^{s}\}dA] \qquad (2.35)$$

where $[\bar{Y}]$ and $[\bar{Z}]$ are influences performed on actual elements and cells due to a unit force acting on image nodes. The matrices can be represented in terms of the coefficients in $[Y]$ and $[Z]$ with appropriate signs:

$$[\bar{Y}] = \begin{vmatrix} Y_{111} & Y_{211} \\ -Y_{112} & -Y_{212} \\ Y_{122} & Y_{222} \end{vmatrix}, \quad [\bar{Z}] = \begin{vmatrix} Z_{111} & Z_{211} \\ -Z_{112} & -Z_{212} \\ Z_{122} & Z_{222} \end{vmatrix} \qquad (2.36)$$

Performing numerical integration on equations 2.33, 2.34, and 2.35, the problem may be solved by considering only half of the boundary and domain of interest.

# CHAPTER III

# THE FINITE ELEMENT METHOD

## 3.1 Introduction

The finite element method is a digital method for stress analysis and other field problems of large size. It is an especially powerful method for Soil Structure Interaction problems for which the complex behavior of soil, structure, and the interface between soil and structure can be simulated by different types of finite elements.

The finite element method is classified into three approaches depending on the selection of assumed displacement or stress function over the continuum: the displacement method, the equilibrium method, and the mixed method. Displacements are assumed as primary unknown quantities in the displacement method; stresses are assumed as primary unknown quantities in the equilibrium method; and some displacements and some stresses are assumed as unknown quantities in the mixed method. The displacement method is the only one to be further presented in this report.

In the displacement method of finite element analysis of a continuum, the continuous body is represented by an assemblage of discrete elements connected at various nodal points to build a discretized model of the body. Assumed displacement functions are chosen to approximate the behavior of the actual displacement field over each element. The principle of minimum potential energy is usually applied to obtain a set of equilibrium equations for each element. The overall performance of the continuum can be established by superimposing the equilibrium equations of each element. After incorporating boundary conditions (prescribed displacements along the boundary), the whole set of simultaneous equations are ready to be solved.

In order to achieve a realistic modeling in the study of U-Frame-soil system by the finite element method, the behavior of the U-Frame structure and the response of the interface between the U-Frame structure and surrounding soil must be examined. The idealization of the U-Frame structure has been presented in section 1.5 and it can be discretized by any type of 2-D finite elements. Isoparametric quadrilateral elements are used herein.

The soil/structure interface may produce discontinuities in displacements and stresses. The physical behavior of such discontinuities involves debonding and slip. The term "debonding" describes the separation of the two blocks of the continuum adjacent to the interface surface, which are initially in contact. Subsequent contact can also develop by the movement of the two blocks towards each other. The term "slip" defines the relative motion along the interface surface when the shearing force exceeds the shear strength of the interface. The debonding and slip make the discontinuities physically nonlinear; therefore, special solution techniques must be employed (20).

Previous attempts have been made to develop discrete elements to represent the interface behavior. Goodman, Taylor, and Brekke (21) developed a simple rectangular, two-dimensional element with eight degrees-of-freedom. With this element, adjacent blocks of continuous elements can penetrate into each other. Zienkiewicz, et al. (22) advocate the use of continuous isoparametric elements with a simple nonlinear material property for shear and normal stresses, assuming uniform strain in the thickness direction.

Goodman, Taylor, and Brekke's joint elements are applied to model the interface between the U-Frame structure and surrounding soil. As numerical difficulties may arise in their suggested iterative procedure in simulating no-tension behavior along the interface, the iterative procedure proposed by Zienkiewicz, Valliappan, and King (23)-which has been proved always convergent-is employed instead.

## 3.2 Basic Steps of Displacement Method

The displacement method of finite element analysis can be considered to involve six steps (24)(25).

<u>Step 1. Discretization of a continuum:</u> Discretization is the process of dividing the given body into an equivalent system of finite elements. In particular, for the infinite continuum such as encountered in SSI problems, only a significant portion of such a continuum needs to be considered and discretized.

Step 2. Selection of element displacement function: In this step, a pattern of solution for the unknown displacements is assumed over each element. A number of conditions must be satisfied for the chosen pattern to yield a satisfactory, consistent, and convergent solution. Details of mathematics of these requirements, such as conforming and nonconforming conditions can be found elsewhere (26). In general, the assumed displacement function $\{u\}$ is in a polynomial form expressed in terms of a series of interpolation functions $[N]^T$ and a set of nodal displacements $\{d\}$ such that

$$\{u\} = [N]^T\{d\} \tag{3.1}$$

Step 3. Derivation of element stiffness and element equations: Several procedures are available for the derivation of equations defining properties of a finite element. The strain vector $\{\varepsilon\}$ and stress vector $\{\sigma\}$ are first calculated in terms of the matrix of differential operators $[\partial]$ and the elastic matrix $[C]$ by strain-displacement and stress-strain relationships.

$$\{\varepsilon\} = [\partial]\{u\} = [\partial][N]^T\{d\} = [B]\{d\} \tag{3.2}$$

$$\{\sigma\} = [C]\{\varepsilon\} = [C][B]\{d\} \tag{3.3}$$

The total strain energy in an element is then calculated and the element stiffness $[k]$ can be derived from the principal of minimum potential energy by taking the first variation of strain energy with respect to nodal displacements. The strain energy and the element stiffness can be expressed as

$$\text{S.E.} = \int 0.5\ \{\varepsilon\}^T\{\sigma\}dV = 0.5\int\{d\}^T[B]^T[C][B]\{d\}dV \tag{3.4}$$

$$[k] = \int [B]^T [C][B]dV \tag{3.5}$$

where the integral denotes a volume integral performed over a 3-D element or an area integral performed over a 2-D element. An element in the stiffness matrix [k], $k_{ij}$, is the influence coefficient which indicates the force induced in the ith degree-of-freedom due to a unit displacment allowed in the jth degree-of-freedom.

Since the surface/body forces acting on the element can be converted into an equivalent nodal force vector {sb}, a set of simultaneous algebraic equations is generated when the equilibrium relation among the stiffness matrix [k], nodal force vector {r}, equivalent nodal force vector {sb}, and nodal displacement vector {d} is applied:

$$[k]\{d\} = \{r\} + \{sb\} \tag{3.6}$$

Step 4. Assembly of element stiffness and nodal forces: Equation 3.6 is evaluated for each element in the structure and combined to obtain a stiffness relation for the entire system. This is done by the "direct stiffness method" by adding the matrix equations for each element one by one. Again, the overall equilibrium equations can be expressed as a set of simultaneous algebraic equations in terms of global stiffness [K], global nodal force vector {R}, displacement vector {D}, and equivalent nodal force vector {SB} due to surface/body forces for the entire body. Thus,

$$[K]\{D\} = \{R\} + \{SB\} \tag{3.7}$$

Before the set of simultaneous equations, 3.7, can be solved, prescribed displacement boundary conditions must be taken into account by appropriate modifications.

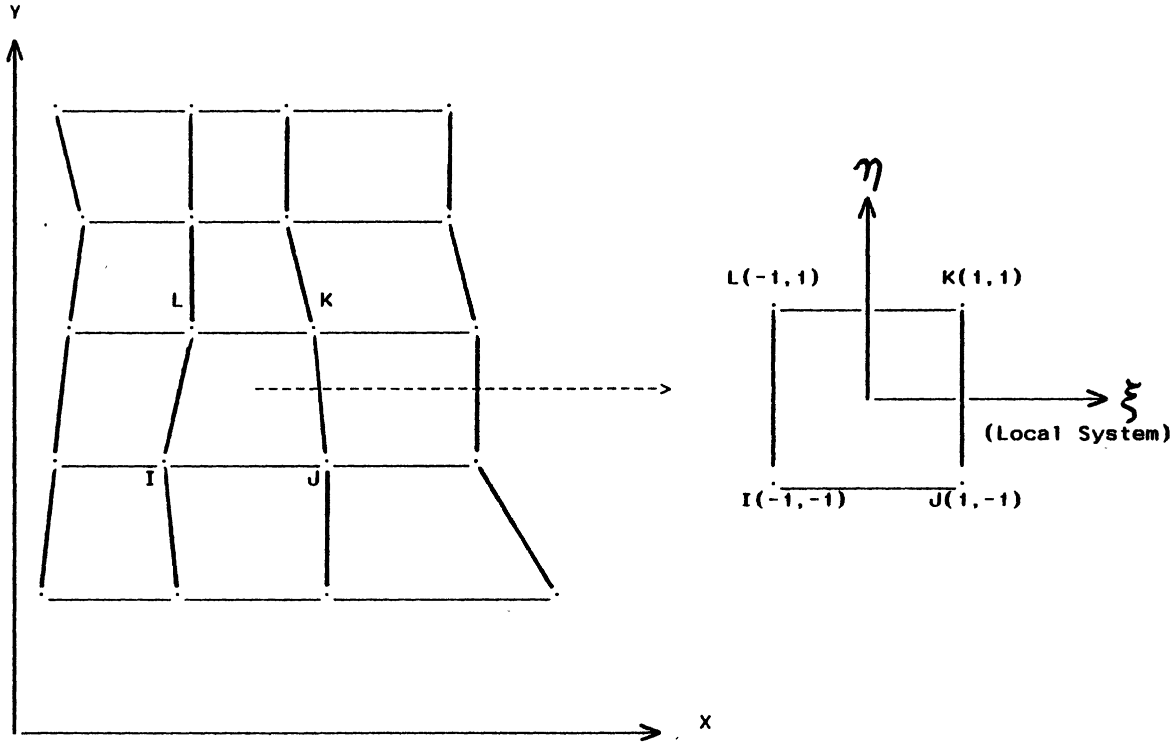
Step 5. Solutions for the unknown displacements: The algebraic equations assembled in step 4 are solved for the unknown nodal displacements {D}. In linear equilibrium problems, this is a relatively straightforward application of matrix algebra techniques. However, for nonlinear problems the desired solutions are obtained by a sequence of steps, each step involving modifications of the stiffness matrix and/or load vector.

Step 6. Computation of element stresses and strains: In the displacement method, nodal displacements are computed as primary unknown quantities by solving equation 3.7. Stresses and strains are the secondary quantities that can be computed based on the nodal displacements from equation 3.2 and equation 3.3.

## 3.3 The Isoparametric Formulation of Quadrilateral Elements

The concept of isoparametric elements has been used commonly for the finite element formulations. It offers a number of advantages such as efficient integration and differentiation, and easy handling of curved and arbitary geometric shapes. The basic idea of isoparametric elements is to express both the displacement and the geometry of the element by using the same interpolation functions $N_i$.

For a four-node quadrilateral isoparametric element shown in Figure 10, the displacements at any point within this element are given by



Assemblage (Global System)

Figure 10. The Isoparametric Quadrilateral Elements

$$\{u\} = \begin{vmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{vmatrix} \{d\} = [N]^T \{d\} \qquad (3.8)$$

Where {d} is the vector of nodal displacements, $[N]^T$ is the interpolation matrix.

In the isoparametric concept, the coordinates of any point within the element, {x}, can be expressed in terms of the same functions $N_i$. Hence,

$$\{x\} = \begin{vmatrix} N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\ 0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 \end{vmatrix} \{x_n\} = [N]^T \{x_n\} \qquad (3.9)$$

where $\{x_n\}$ contains the coordinates of the nodal points.

The matrix $[N]^T$ in equations 3.8 and 3.9 is composed of the following interpolation functions:

$$\begin{aligned} N_1 &= (1-\xi)(1-\eta)/4, \quad N_2 = (1+\xi)(1-\eta)/4 \\ N_3 &= (1+\xi)(1+\eta)/4, \quad N_4 = (1-\xi)(1+\eta)/4 \end{aligned} \qquad (3.10)$$

If plane strain conditions are assumed, the strain-displacement relation for small strains is

$$\{\varepsilon\} = \begin{Bmatrix} \varepsilon_X \\ \varepsilon_Y \\ \gamma_{XY} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{Bmatrix} = [B]\{d\} \qquad (3.11)$$

where [B] is obtained by taking appropriate derivatives of $N_i$ in matrix $[N]^T$ of equation 3.8.

$$[B] = \begin{vmatrix} \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \frac{\partial N_3}{\partial x} & 0 & \frac{\partial N_4}{\partial x} & 0 \\ 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \frac{\partial N_3}{\partial y} & 0 & \frac{\partial N_4}{\partial y} \\ \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial y} & \frac{\partial N_3}{\partial x} & \frac{\partial N_4}{\partial y} & \frac{\partial N_4}{\partial x} \end{vmatrix} \qquad (3.12)$$

The global (x,y) and local ($\xi,\eta$) derivatives are related through the Jacobian as

$$\begin{Bmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{Bmatrix} = [J]^{-1} \begin{Bmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{Bmatrix}, \text{ where } [J]^{-1} = \frac{1}{|J|} \begin{vmatrix} \frac{\partial y}{\partial \eta} & \frac{-\partial y}{\partial \xi} \\ \frac{-\partial x}{\partial \eta} & \frac{\partial x}{\partial \xi} \end{vmatrix} \qquad (3.13)$$

and $|J|$ is the determinant of $[J]$:

$$|J| = \det \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{vmatrix} = \sum_{i=1}^{n} \sum_{j=1}^{n} \left( \frac{\partial N_i}{\partial \xi} \frac{\partial N_j}{\partial \eta} - \frac{\partial N_i}{\partial \xi} \frac{\partial N_j}{\partial \eta} \right) x_i y_j \qquad (3.14)$$

where n is the number of nodes in the element.

The variational functional for the displacement method is given by the potential energy $\pi_p$ of the system, which can be written as

$$\pi_p = 0.5 \int \{\epsilon\}^T \{\sigma\} dV - \int \{u\}^T \{b\} dV - \int \{u\}^T \{t\} dS \qquad (3.15)$$

Where $0.5\{\epsilon\}^T \{\sigma\} dV$ is the strain energy per unit volume, {b} and {t} are the prescribed body force and surface traction vector respectively.

By assuming that the material behavior is linearly elastic, the stress-strain relation can be expressed for the plane strain case as

$$\{\sigma\} = \begin{Bmatrix} \sigma_X \\ \sigma_Y \\ \tau_{XY} \end{Bmatrix} = \frac{E}{(1+\nu)(1-2\nu)} \begin{vmatrix} 1-\nu & & 0 \\ & 1-\nu & 0 \\ 0 & 0 & 1-2\nu \end{vmatrix} \begin{Bmatrix} \varepsilon_X \\ \varepsilon_Y \\ \gamma_{XY} \end{Bmatrix} = [C]\{\epsilon\} \quad (3.16)$$

where [C] is the elastic matrix, E is Young's modulus, and $\nu$ is Poisson's ratio. Substitution of equations 3.8, 3.11, and 3.16 into equation 3.15 leads to

$$\pi_p = 0.5\int(\{d\}^T[B]^T[C][B]\{d\}-2\{d\}^T[N]\{b\})dV - \int\{d\}^T[N]\{t\}dS \quad (3.17)$$

By taking the first variation of $\pi_p$ with respect to nodal displacements and considering the principal of minimum potential energy,

$$\delta\pi_p = 0 \quad (3.18)$$

the following is obtained

$$[k]\{d\} = \int[N]\{b\}dV + \int[N]\{t\}dS = \{r\} \quad (3.19)$$

where for the element in Figure 10

$$[k] = h\int_{-1}^{1}\int_{-1}^{1} [B]^T[C][B]|J|d\xi d\eta \quad (3.20)$$

and

$$\{r\} = h\int_{-1}^{1}\int_{-1}^{1} [N]\{b\}|J|d\xi d\eta + h\int[N]\{t\}dS \quad (3.21)$$

in which h is the thickness of the element; for plane strain conditions h is taken as unity.

## 3.4 Analysis of Joint Elements

A discontinuity at the interface between the U-Frame structure and surrounding soil can be simulated as a special kind of joint between faces of blocks. The characteristic of joint elements is that they will separate in response to tension, slide in response to shear, and transmit any force in response to compression.

Figure 11 shows a four-node joint element. This element has length L and very small width h. The origin is at the center and $\gamma$ is the angle between local (s,n) and global (x,y) coordinate systems.

The derivation of the joint element stiffness matrix is obtained from the work of Goodman, Taylor, and Brekke (21). The iterative solution to simulate real properties of joint elements is based on the so called "load transfer method" proposed by Zienkiewicz, Valliappan, and King (23).

### 3.4.1 "Strain"-Displacement Relationship for the Joint Elements

The strain-displacement relationship $[B_J]$ describes the relative displacement between joint walls JK and HI (see Figure 11) as a function of nodal displacements $\{d\}_{S,N}$.

$$\{d\}_{S,N} = \{u_H \quad v_H \quad u_I \quad v_I \quad u_J \quad v_J \quad u_K \quad v_K\}^T \qquad (3.22)$$

where $u_i$ and $v_i$ are the displacements of node i in the tangential and normal directions.

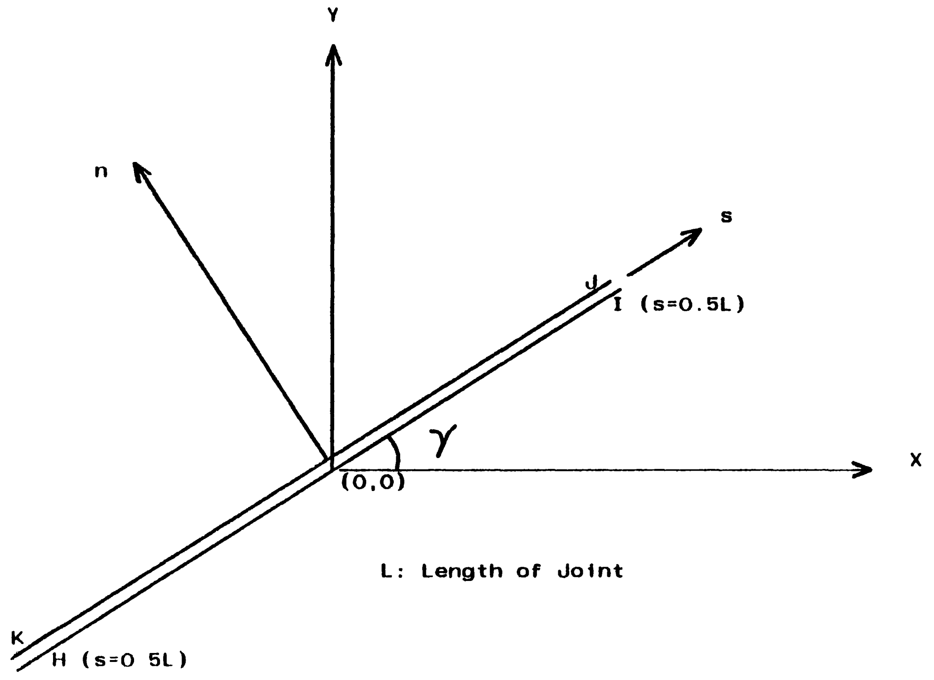


Figure 11. A Typical Joint Element

The joint "strain" is defined as the relative displacement between wall JK and HI in the tangential and normal direction. Thus, the shear "strain" $\lambda u$ and normal "strain" $\lambda v$ are given by

$$\{\varepsilon_J\}_{S,N} = \begin{Bmatrix} \lambda u \\ \\ \lambda v \end{Bmatrix} = \begin{Bmatrix} u^{top} - u^{bottom} \\ \\ v^{top} - v^{bottom} \end{Bmatrix} \tag{3.23}$$

where $\{\varepsilon_J\}_{S,N}$ is the strain vector in local coordinate system.

The displacements in the joint element can be expressed in terms of nodal displacements $\{d\}_{S,N}$ through a linear interpolation formula. Thus, the displacements along the bottom wall HI are

$$\begin{Bmatrix} u^{bottom} \\ \\ v^{bottom} \end{Bmatrix} = 0.5 \begin{vmatrix} 1-2s/L & 0 & 1+2s/L & 0 \\ \\ 0 & 1-2s/L & 0 & 1+2s/L \end{vmatrix} \begin{Bmatrix} u_H \\ v_H \\ u_I \\ v_I \end{Bmatrix} \tag{3.24}$$

If a=1-2s/L and b=1+2s/L, with a similar expression, the displacements along the top wall JK are

$$\begin{Bmatrix} u^{top} \\ \\ v^{top} \end{Bmatrix} = 0.5 \begin{vmatrix} b & 0 & a & 0 \\ \\ 0 & b & 0 & a \end{vmatrix} \begin{Bmatrix} u_J \\ v_J \\ u_K \\ v_K \end{Bmatrix} \tag{3.25}$$

Subsititution of equations 3.24 and 3.25 into equation 3.23 leads to

$$\{\epsilon_J\}_{S,N} = \begin{Bmatrix} \lambda u \\ \lambda v \end{Bmatrix} = 0.5 \begin{vmatrix} -a & 0 & -b & 0 & b & 0 & a & 0 \\ 0 & -a & 0 & -b & 0 & b & 0 & a \end{vmatrix} \{d\}_{S,N}$$

$$= [B_J]\{d\}_{S,N} \tag{3.26}$$

Equation 3.26 relates "strains" to nodal displacements for the joint elements.

### 3.4.2 "Stress-Strain" Relationship for the Joint Elements

Since the actual load transfer across a rough interface may occur at point contacts, Goodman (27) defined the joint element "stresses" as follows: the normal and shear stresses on the interface wall are equal to the total normal and shear forces per unit area (the thickness of the element is taken as unity).

$$\{\sigma_J\}_{S,N} = \begin{Bmatrix} \sigma_S \\ \sigma_N \end{Bmatrix} = \frac{1}{L} \begin{Bmatrix} F_S \\ F_N \end{Bmatrix} \tag{3.27}$$

where $\{\sigma_J\}_{S,N}$ = the "stress" vector of joint elements,

$F_S$ = tangential force in the joint element, and

$F_N$ = normal force in the joint element.

The "stress-strain" relationship for joint elements can be expressed as

$$\{\sigma_J\}_{S,N} = \begin{vmatrix} k_S & 0 \\ o & k_N \end{vmatrix} \{\epsilon_J\}_{S,N} = [C_J]\{\epsilon_J\}_{S,N} \tag{3.28}$$

where $k_S$ = stiffness per unit length in tangential direction

and $k_N$ = stiffness per unit length in normal direction.

The unit normal stiffness $k_N$ and shear stiffness $k_S$ can be obtained from a direct shear test. For a joint element with length L and unit thickness, at first, a normal force is applied and the specimen shortens. The joint normal deformation $\lambda v$ is measured and it may be plotted against the applied force per unit length, $F_N/L$, as illustrated in curve 1 of Figure 12 (21). Similarly, the tangential deformation $\lambda u$ may be plotted against the shearing force per unit length, $F_S/L$, as shown in curve 2 of Figure 12, when a tangential force is applied.

### 3.4.3 Derivation of Joint Stiffness Matrix

When the "strain"-displacement and "stress-strain" relationship are obtained for a joint element, the joint stiffness matrix can be evaluated from equation 3.5:

$$[k] = \int [B]^T [C][B] \, dV \tag{3.5}$$

Thus, the joint stiffness $[k_J]_{S,N}$ with unit thickness can be expressed in terms of its length L as

$$[k_J]_{S,N} = \int_{L/2}^{L/2} [B_J]^T [C_J][B_J] \, ds \tag{3.29}$$

where $[B_J]$ and $[C_J]$ are given by equations 3.26 and 3.28.

The only terms in equation 3.29 varying along the length are the products of a $(1-2s/L)$ and b $(1+2s/L)$ in matrix $[B_J]$. After performing the integration with respect to length for these a and b terms, equation 3.29 is reduced to

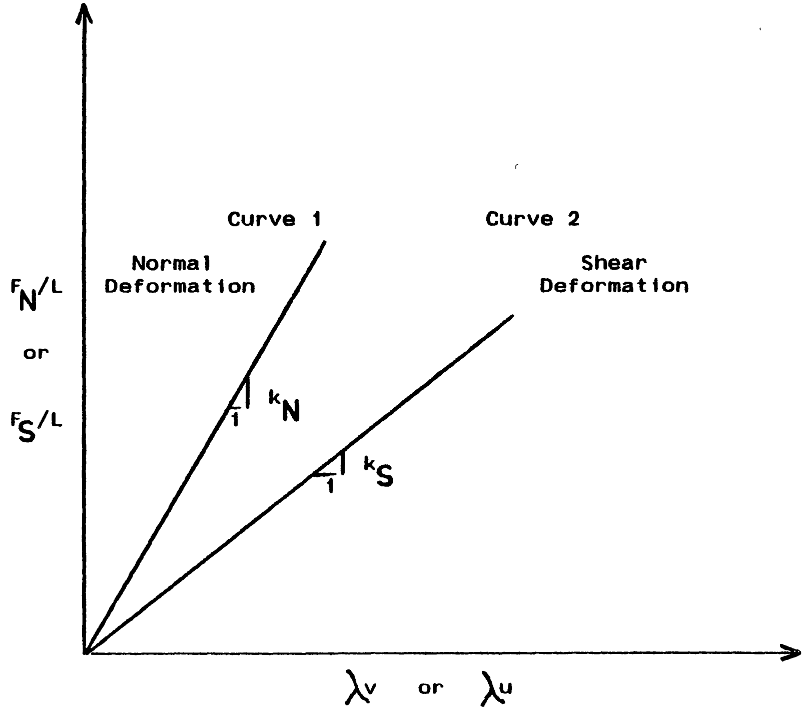


Figure 12. The Unit Tangential and Normal Stiffness of Rock Joints from the Result of Direct Shear Test

$$[k_J]_{S,N} = \frac{L}{6}\begin{vmatrix} 2k_S & 0 & 1k_S & 0 & -1k_S & 0 & -2k_S & 0 \\ 0 & 2k_N & 0 & 1k_N & 0 & -1k_N & 0 & -2k_N \\ 1k_S & 0 & 2k_S & 0 & -2k_S & 0 & -1k_S & 0 \\ 0 & 1k_N & 0 & 2k_N & 0 & -2k_N & 0 & -1k_N \\ -1k_S & 0 & -2k_S & 0 & 2k_S & 0 & 1k_S & 0 \\ 0 & -1k_N & 0 & -2k_N & 0 & 2k_N & 0 & 1k_N \\ -2k_S & 0 & -1k_S & 0 & 1k_S & 0 & 2k_S & 0 \\ 0 & -2k_N & 0 & -1k_N & 0 & 1k_N & 0 & 2k_N \end{vmatrix} \tag{3.30}$$

The task remaining now is to rotate the local stiffness matrix $[k_J]_{S,N}$ to global stiffness matrix $[k_J]_{X,Y}$ with reference to Figure 11. The final results can be expressed as

$$[k_J]_{X,Y} = [W]^T [k_J]_{S,N}[W] \tag{3.31}$$

in which [W] is the transformation matrix and

$$[W]=\begin{vmatrix} \cos\gamma & \sin\gamma & 0 & 0 & 0 & 0 & 0 & 0 \\ -\sin\gamma & \cos\gamma & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \cos\gamma & \sin\gamma & 0 & 0 & 0 & 0 \\ 0 & 0 & -\sin\gamma & \cos\gamma & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \cos\gamma & \sin\gamma & 0 & 0 \\ 0 & 0 & 0 & 0 & -\sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos\gamma & \sin\gamma \\ 0 & 0 & 0 & 0 & 0 & 0 & -\sin\gamma & \cos\gamma \end{vmatrix} \tag{3.32}$$

where $\gamma$ is the angle between local and global coordinate systems.

### 3.4.4 Iterative Solution to Simulate Real Properties of Joint Elements

Joint elements are used to simulate the interface between the U-Frame structure and surrounding soil. They are generally incapable of sustaining tensile stresses. Therefore, the assumptions for linear-elastic behavior can only exist when all joint elements are subjected to compressive stresses within the elastic limit.

The load transfer method is an iterative process devised for the stress analysis of joint elements resulting in a no tension state. The essential steps of the load transfer method are summarized as follows (23):

1. Analyze the problem as an elastic one and compute the principal stresses in each joint element.

2. At the end of stage 1, certain tensile stresses may develop. As joint elements are assumed incapable of sustaining tensions, they should be eliminated without permitting any point in the structure to displace. Hence, "restraining" forces have to be applied temporarily to maintain the structure in "equilibrium" at this stage. Such restraining forces along the wall of joint elements can be evaluated by using the existing tensile stresses and shear stresses. These restraining forces are transformed into equivalent nodal forces in terms of the reverse procedure used in calculating shear and normal stresses.

3. As the restraining forces do not in fact exist, their effects have to be removed from the structure by

superposition of equal but opposite nodal forces. The structure in now reanalyzed with a new nodal force vector which is updated at the nodes of joint elements by including the "de-restraining" forces. The structure is again assumed elastic and it will be found that tensions may still develop. However, these tensions will be much reduced compared with those of the previous stage.

4. If at the end of stage 3, principal tensions are still in existence, steps 2 and 3 are repeated until all tensile stresses are reduced to a negligible value.

The load transfer method which has been proved always convergent (23) provides an effective treatment for the iterative solution to simulate real properties of joint elements. As the global stiffness matrix remains the same throughout, the inversion of this matrix must be computed only in the first solution.

## 3.5 Incorporation of Symmetry Conditions

The incorporation of symmetry conditions in the finite element analysis is simple. Only one half of the domain of interest needs to be discretized and analyzed, since stiffness matrices are symmetrical. Care must be taken that some artificial displacement boundary conditions must be imposed along the line of symmetry.

In the case of symmetry about a vertical axis, for instance, the effects of such symmetry cause no horizontal displacements along the vertical axis. Therefore, only one half of the original problem need be analyzed by the procedures presented in section 3.2 with imposed artificial roller supports to prevent horizontal movements along the vertical axis.

# CHAPTER IV

# THE COUPLING OF BOUNDARY AND FINITE ELEMENT METHODS

## 4.1 Introduction

The boundary and the finite element methods are applicable to solve most engineering problems. Neither of the two methods is a unique technique applied to elasticity problems.

The finite element method discretizes the domain of interest into a number of finite elements. The equilibrium equations are then approximated by displacement functions which satisfy displacement boundary conditions. The boundary element method discretizes the boundary of interest into a number of boundary elements. Both displacement and traction boundary conditions are then approximated by "fundamental solutions" which satisfy the equilibrium equations at infinity. This implies that the idea of combining boundary and finite element methods is of great interest in analyzing the U-Frame structure as the surrounding soil extends to infinity which can be better represented by boundary elements. Finite elements, on the other hand, are easier to apply to simulate the U-Frame structure and the interface of soil-structure system.

In order to obtain the required matrices used in the coupling of boundary and finite element methods, the element equilibrium equation is rearranged and assembled such that the global equilibrium equations can be expressed in a fashion similar to that used for the governing boundary integral equation.

The starting expression of an elasticity problem for a finite element solution is presented in equation 3.19:

$$[k]\{d\} = \int [N]\{b\}dV + \int [N]\{t\}dS = \{r\} \qquad (3.19)$$

where $\{t\}$ is the applied traction function over the element side on the boundary and it can be linearly interpolated in terms of the nodal traction vector $\{p\}$ on the same side with length L:

$$\{t\} = \begin{vmatrix} F_1 & 0 & F_2 & 0 \\ 0 & F_1 & 0 & F_2 \end{vmatrix} \{p\} = [F]^T \{p\} \qquad (4.1)$$

where $F_1 = 1-2\xi/L$ and $F_2 = 1+2\xi/L$.

Equation 4.1 is substituted into equation 3.19 and equilibrium equations are assembled for each element, then the global equilibrium equation for two-dimensional problems can be written as

$$[K]\{D\} = \sum_{j=1}^{J} \int_{C_j} [N][F]^T \{P\}ds^j + \{V\} \qquad (4.2)$$

where $\{V\}$ is the global body force term, $[N]$ and $[F]^T$ are interpolation functions of nodal displacements and tractions, and the summation applies over the side of the jth element on the boundary. Thus, the integral terms in this equation can be evaluated as matrix [M] from:

$$\sum_{j=1}^{J}\int_{C_j}[N][F]^T\{P\}ds^j = [M]\{P\} \qquad (4.3)$$

where [M] is the distribution matrix. The coefficients of [M] depend on the type of interpolation functions used for the displacements and tractions. Hence the global equilibrium equation becomes

$$[K]\{D\} = [M]\{P\} + \{V\} \qquad (4.4)$$

which is similar to the governing boundary integral equation

$$[H]\{D\} = [G]\{P\} + \{B\} \qquad (2.19)$$

Suppose that a given problem consists of two regions, $R^1$ and $R^2$ as shown in Figure 13. Region 1 is studied using finite elements and region 2 is formulated by boundary elements. $\Gamma_I$ is the interface between the two regions. The compatibility and equilibrium requirements along the interface when the two regions are joined together are

$$U_I^1 = U_I^2 \qquad P_I^1 + P_I^2 = 0 \qquad (4.5)$$

in which $U_I^1$, $U_I^2$, $P_I^1$, and $P_I^2$ refer to the displacements and tractions on the interface for regions 1 and 2.

The coupling of boundary and finite element methods may be achieved in either of two ways: by considering the whole problem using an equivalent boundary element method or converting the boundary element subregions into an equivalent finite element method (28)(29). These two approaches will be described in detail in the following sections.

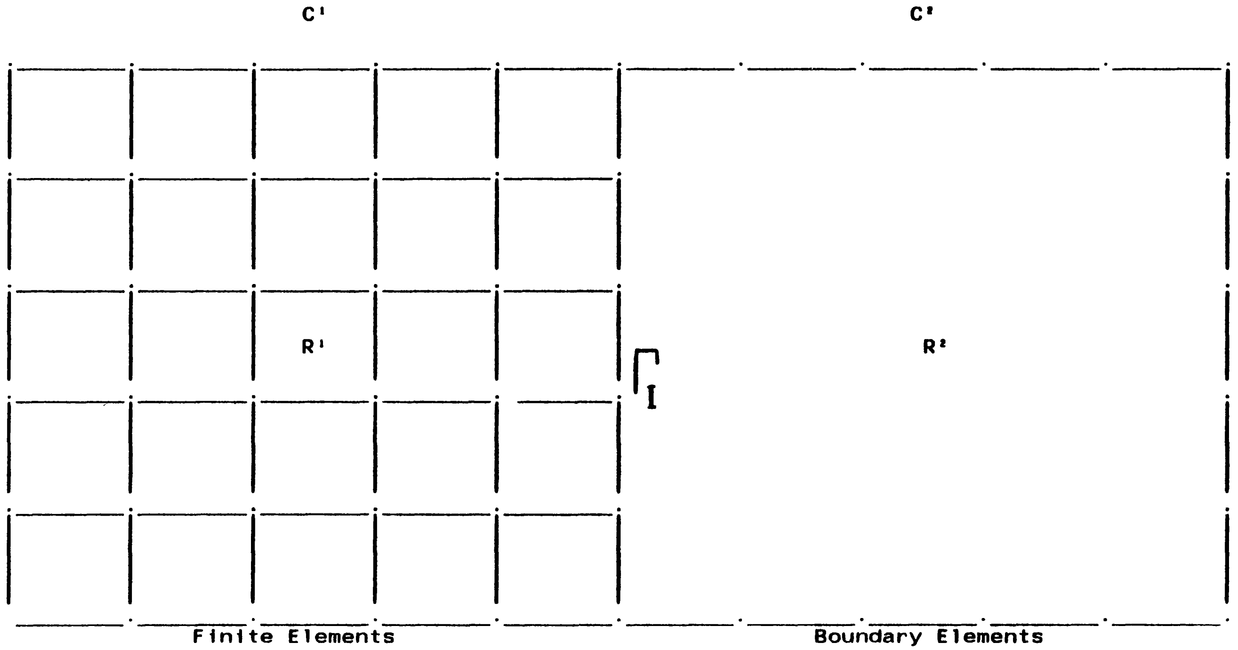


Figure 13. Domain Divided into Finite and Boundary Element Regions

## 4.2 The Equivalent Boundary Element Method

In the equivalent boundary element method, region 1 (see Figure 13) is treated as a boundary element type region. For this region, equation 4.4 can be written as

$$[K^1 \quad K_I^1]\begin{Bmatrix} D^1 \\ D_I^1 \end{Bmatrix} = [M^1 \quad M_I^1]\begin{Bmatrix} P^1 \\ P_I^1 \end{Bmatrix} + \{V\} \tag{4.6}$$

and for region 2, equation 2.19 becomes

$$[H_I^2 \quad H^2]\begin{Bmatrix} D_I^2 \\ D^2 \end{Bmatrix} = [G_I^2 \quad G^2]\begin{Bmatrix} P_I^2 \\ P^2 \end{Bmatrix} + \{B\} \tag{4.7}$$

by letting $P_I = P_I^2 = - P_I^1$ and $U_I = U_I^1 = U_I^2$ which satisfy the compatibility and equilibrium conditions (equation 4.5). Equations 4.6 and 4.7 can be reordered as follows:

$$[K^1 \quad K_I^1 \quad M_I^1]\begin{Bmatrix} D^1 \\ D_I \\ P_I \end{Bmatrix} = [M^1]\{P^1\} + \{V\} \tag{4.8}$$

and

$$[H_I^2 \quad -G_I^2 \quad H^2]\begin{Bmatrix} D_I \\ P_I \\ D^2 \end{Bmatrix} = [G^2]\{P^2\} + \{B\} \tag{4.9}$$

Writing equations 4.8 and 4.9 together as a single matrix equation, yields

$$\begin{vmatrix} K^1 & K_I^1 & M_I^1 & 0 \\ 0 & H_I^2 & -G_I^2 & H^2 \end{vmatrix}\begin{Bmatrix} D^1 \\ D_I \\ P_I \\ U^2 \end{Bmatrix} = \begin{vmatrix} M^1 & 0 \\ 0 & G^2 \end{vmatrix}\begin{Bmatrix} P^1 \\ P^2 \end{Bmatrix} + \begin{Bmatrix} V \\ B \end{Bmatrix} \tag{4.10}$$

Notice that on the boundary of finite element region $R^1$, only the displacements have to be prescribed; however, on the boundary of $R^2$, displacements or tractions need to be defined. The disadvantage of the equivalent boundary element method is that the equations of the boundary element region must be reordered.

### 4.3 The Equivalent Finite Element Method

The equivalent finite element method transforms the boundary element region (region 2 in Figure 13) into a finite element type region. The traction {P} in equation 4.4 can be computed by inverting matrix [G].

$$[G]^{-1}([H]\{D\} - \{B\}) = \{P\} \tag{4.11}$$

Premultiplying equation 4.11 by matrix [M] defined by equation 4.3 yields:

$$([M][G]^{-1}[H])\{D\} - ([M][G]^{-1})\{B\} = [M]\{P\} \tag{4.12}$$

Hence the following can be defined:

$$[K'] = [M][G]^{-1}[H],\ \{V'\} = [M][G]^{-1}\{B\},\ \{R'\} = [M]\{P\} \tag{4.13}$$

Thus equation 4.12 has the same form of a finite element problem:

$$[K']\{D\} = \{R'\} + \{V'\} \tag{4.14}$$

The main difficulty in the above formulation is that the matrix [K'] is generally unsymmetrical, although from the reciprocal theorem a stiffness matrix should be symmetric. The asymmetry is due to the approximation involved in the discretization process and the choice of the assumed solution (29). The matrix can be made symmetric by applying the least square method to the nonsymmetric off-diagonal terms. After minimizing the errors, the new symmetric coefficients are

$$k_{ij} = 0.5\ (k'_{ij} + k'_{ji}) \qquad (4.15)$$

The equivalent finite element type matrices of equation 4.14 may be assembled with matrices obtained from region 1 to form a global system of equations. The disadvantage of this method is that the inverse of matrix [G] must be computed and a number of matrix multiplications must be performed.

As the equivalent finite element method involves a number of matrix multiplications, it needs more computer execution time and requires more storage to save the intermediate data in the multiplication process. Therefore, the equivalent boundary element method is employed here.

# CHAPTER V

# DESCRIPTION OF COMPUTER PROGRAMS

## 5.1 Flowcharts of Computer Programs

The equivalent boundary element approach of the coupling of boundary and finite element methods has been coded in program BOUFIN to implement the numerical processes developed in chapters II, III, and IV. The displacement approach of the finite element method alone has been coded in program FINITE since general purpose programs STRUDL and NASTRAN can not simulate the behavior at soil/structure interface. Both programs are written in the FORTRAN programming language. In program BOUFIN, isoparametric quadrilateral elements are used to simulate structure behavior; linear boundary elements and internal cells are used to model the boundary of surrounding soil and to compute the body force terms of soil mass. In program FINITE, isoparametric quadrilateral elements are employed for both structure and surrounding soil.

The flowchart of program BOUFIN is shown in Figure 14, which presents the essential numerical procedures performed in the coupling of boundary and finite element methods. To check if the soil-structure system is symmetrical about a vertical axis is crucial as, unlike the finite element

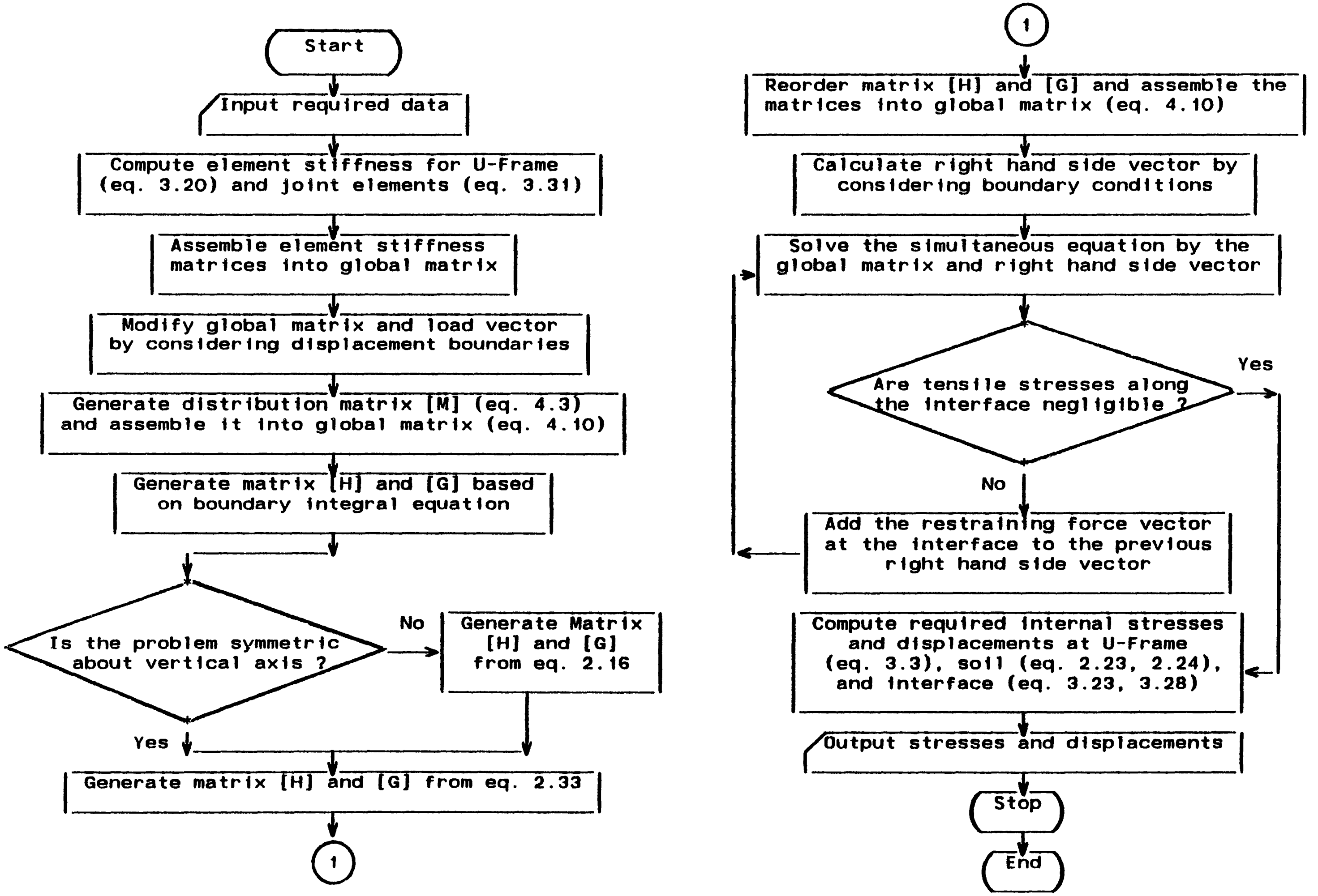


Figure 14. Flowcharts of Program BOUFIN

method which can satisfy required symmetry conditions by simply modifying displacement boundaries, special treatments must be taken into consideration. Note that in the formulation of matrices [H] and [G], the required integrals are evaluated analytically to obtain more accurate results for a linear boundary element with singularities at either of its extremities (30).

The flowchart of program FINITE is shown in Figure 15, which presents the essential numerical procedures adopted in the finite element method alone. It requires fewer steps than those in program BOUFIN.

## 5.2 Guide for Data Input

This section provides the details of required data input to run programs BOUFIN and FINITE. Due to the facilities in evaluating the complex behavior in the interface of soil-structure system, each joint element can be considered as part of the structure and part of the interface. For instance

1. The nodes on the side of a joint element attached to the structure are classified as sructure nodes; however, the nodes on the other side are classified as interface nodes.

2. All joint elements are classified as structure elements; however, the side of each joint element attached to the soil mass is classified as an interface element.

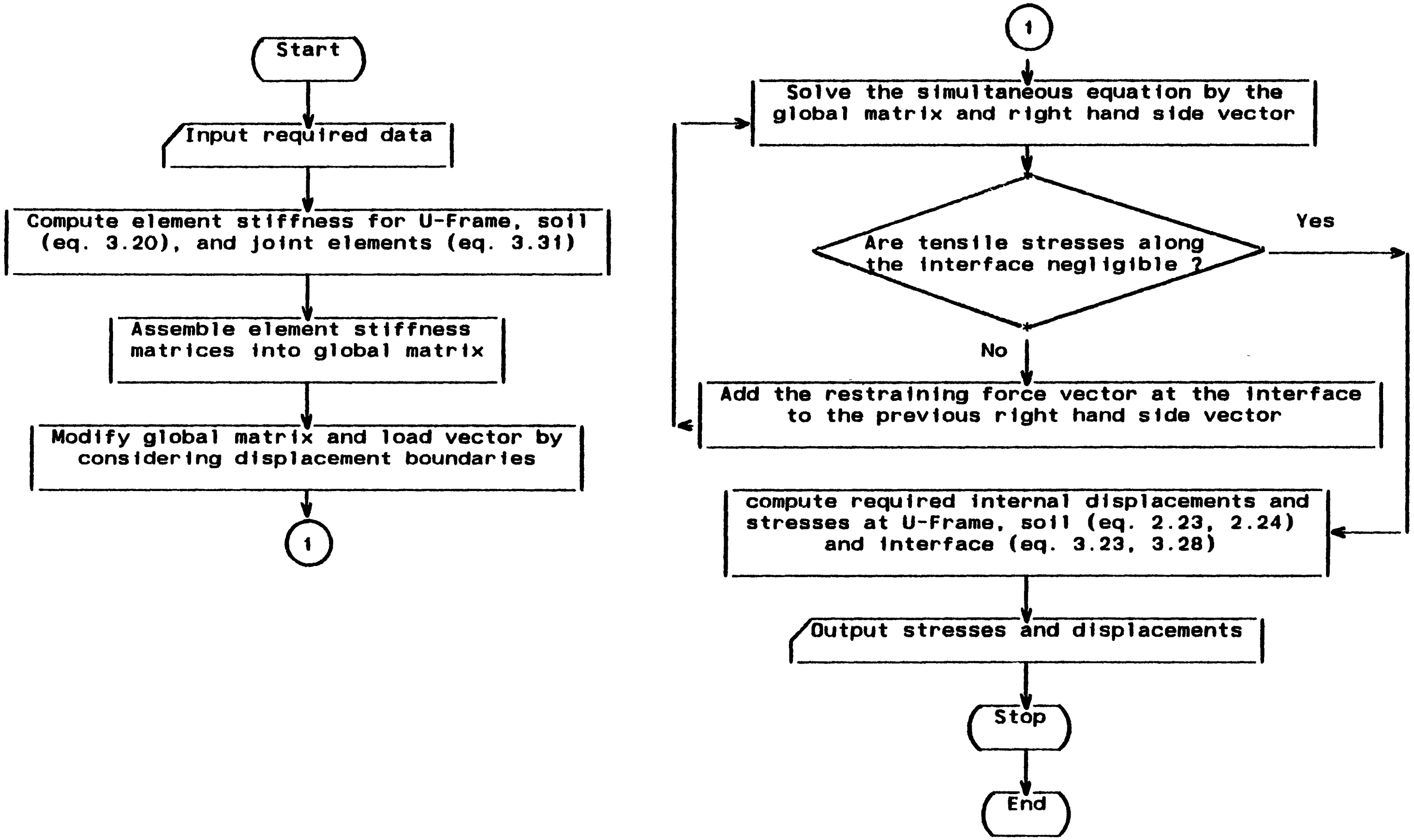


Figure 15. Flowcharts of Program FINITE

### 5.2.1 Numbering Scheme in Soil-Structure System

The following are the restrictions imposed on numbering nodes and elements in the soil-structure system.

1. Each node and element (including internal cells) must be assigned a positive number in sequence starting from the number "one". The U-Frame structure is numbered first, followed by the interface between structure and surrounding soil, then the soil mass. Note that the U-Frame structure must be counted before the interface which, in turn, is numbered before soil mass, otherwise, the computer programs do not run properly.

2. The element incidences should be designated in a counterclockwise direction for isoparametric quadrilateral, joint, and linear boundary elements. Moreover, in order to establish the local coordinate system for each joint element, the two nodes on the side attached to the soil mass are designated first.

### 5.2.2 Data Format

All data input are read in free field formats. Data items should be separated by one or more blanks/commas. No restrictions are imposed to tell integer numbers from real numbers; however, the exponential forms for real or integer numbers are prohibited.

### 5.2.3 Predefined Data File

In addition to the numbering rules and format requirements in subsections 5.2.1 and 5.2.2, the following pertain to the input data description in the next two subsections.

1. The only accepted unit for length is "inch" and the only accepted unit for force is "kip".

2. A line of input may require both alphanumeric and numeric data items. Alphanumeric data items are enclosed in single quotes.

3. A line of input may require a keyword. The acceptable abbreviation for the keyword is indicated by underlined captital letters, namely, the acceptable abbreviation for the keyword "TITle" is "TIT".

4. Items designated by uppercase letters and numbers without quotes indicate numeric data values. Numeric data values are either real or integer according to standard FORTRAN variable naming conventions.

5. Data items enclosed in brackets [ ] may not be required.

### 5.2.4 Input Data Description for BOUFIN

A. Symmetry Conditions--One line for indentifying whether the vertical axis is the line of symmetry

- a. Contents

  'TYPe' 1(or 2)

- b. Definitions

  '1' = unsymmetrical about the vertical axis

  '2' = symmetrical about the vertical axis.

B. Heading--Several lines for identifying the problem

- a. Contents

'TITle' M

. . . . . .

. . . . . . M lines

. . . . . .

b. Definition

M = number of lines for any alphanumeric information, eighty characters (including any imbedded blanks) in each line.

C. Structure Input:

1. Structural Properties--One line

a. Contents

'STRuctural' 'PROperties' E V D

b. Definitions

E = modulus of elasticity for structure

V = Poisson's ratio for structure

D = density of structure.

2. Node Coordinates--As many lines as required

a. Contents

'NODe' 'COOrdinates' M

N XN YN

. . . M lines

. . .

b. Definitions

M = number of nodes in structure

N = node number in structure

XN = X coordinate of node N

YN = Y coordinate of node N.

3. Element Connectivities--As many lines as required

a. Contents

```
'ELEment'  'CONnectivities'   M1   M2
N    IN1    IN2    IN3    IN4
.     .      .      .      .   M1 lines
.     .      .      .      .
N    IN1    IN2    IN3    IN4
.     .      .      .      .   M2 lines
.     .      .      .      .
```

b. Definitions

M1 = number of isoparametric quadrilateral elements in structure

M2 = number of joint elements

N = element number in structure

IN1 = the first node number in element N

IN2 = the second node number in element N

IN3 = the third node number in element N

IN4 = the fourth node number in element N.

4. Prescribed Displacements--As many lines as required

a. Contents

```
'PREscribed' 'DISplacements'   M
N   ['X'   DXN]   ['Y'  DYN]
.     .     .       .     .   M lines
.     .     .       .     .
```

b. Definitions

M = number of nodes with prescribed displacements in structure

N = node number in structure

DXN = prescribed horizontal displacement at node N

DYN = prescribed vertical displacement at node N.

5. Prescribed Non-zero Concentrated Loads--As many lines as required

a. Contents

'PREscribed' 'LOAds' M
N ['X' FXN] ['Y' FYN]
. . . . . M lines
. . . . .

b. Definitions

M = number of nodes with prescribed non-zero concentrated loads in structure
N = node number in structure
FXN = prescribed horizontal load at node N
FYN = prescribed vertical load at node N.

D. Interface Input:

1. Interface Properties--One line

a. Contents

'INTerface' 'PROperties' SN SS

b. Definitions

SN = unit normal stiffness at interface
SS = unit shear stiffness at interface.

2. Node Coordinates--As many lines as required, same as C.2.

3. Element Connectivities--As many lines as required

a. Contents

'ELEment' 'CONnectivities' M

N IN1 IN2

. . . M lines

. . .

b. Definitions

M = number of elements at interface

N = element number at interface

IN1 = the first node number in element N

IN2 = the second node number in element N.

4. Prescribed Displacements--As many lines as required, same as C.4.

5. Prescribed Non-zero Concentrated Loads--As many lines as required, same as C.5.

E. Soil Input:

1. Soil Properties--One line

a. Contents

'SOIl' 'PROperties' E V D

b. Definitions

E = modulus of elasticity for soil

V = Poisson's ratio for soil

D = density of soil.

2. Node Coordinates--As many lines as required

a. Contents

'NODe' 'COOrdinates' M1 M2

N XN YN

. . . M1 lines

. . .

```
N      XN      YN
.       .       .   M2 lines
.       .       .
```

b. Definitions

M1 = number of nodes along the boundary of soil mass

M2 = number of nodes inside the soil mass used for integration

N = node number in soil mass

XN = X coordinate of node N

YN = Y coordinate of node N.

3. Element Connectivities--As many lines as required

a. Contents

```
'ELEment'   'CONnectivities'   M1   M2
N     IN1     IN2
.      .       .   M1 lines
.      .       .
N     IN1     IN2     IN3     IN4
.      .       .       .       .   M2 lines
.      .       .       .       .
```

b. Definitions

M1 = number of linear elements along the boundary of soil mass

M2 = number of internal cells used for integration

N = element number

IN1 = the first node number in element N

IN2 = the second node number in element N

IN3 = the third node number in element N

IN4 = the forth node number in element N.

4. Prescribed Displacements--As many lines as required, same as C.4.

5. Prescribed Non-zero Stresses--As many lines as required

a. Contents

'PREscribed' 'STResses' M

N ['X' TXN] ['Y' TYN]

. . . . . M lines

. . . . .

b. Definitions

M = number of nodes with prescribed non-zero stresses in soil mass

N = node number in soil mass

TXN = prescribed horizontal stress

TYN = prescribed vertical stress.

## 5.2.5 Input Data Description for FINITE

A. Heading--Several lines for identifying the problem

a. Contents

'TITle' M

. . . . . .

. . . . . . M lines

. . . . . .

b. Definition

M = number of lines for any alphanumeric information, eighty characters(including any imbedded blanks) in each line.

B. Structure Input:

1. Structural Properties--One line

a. Contents

'STRuctural' 'PROperties' E V D

b. Definitions

E = modulus of elasticity for structure

V = Poisson's ratio for structure

D = density of structure.

2. Node Coordinates--As many lines as required

a. Contents

'NODe' 'COOrdinates' M

N XN YN

. . . M lines

. . .

b. Definitions

M = number of nodes in structure

N = node number in structure

XN = X coordinate of node N

YN = Y coordinate of node N.

3. Element Connectivities--As many lines as required

a. Contents

'ELEment' 'CONnectivities' M1 M2

N IN1 IN2 IN3 IN4

. . . . . M1 lines

. . . . .

```
N    IN1    IN2    IN3    IN4
.     .      .      .      .     M2 lines
.     .      .      .      .
```

b. Definitions

M1 = number of isoparametric quadrilateral elements in structure

M2 = number of joint elements

N = element number in structure

IN1 = the first node number in element N

IN2 = the second node number in element N

IN3 = the third node number in element N

IN4 = the fourth node number in element N.

4. Prescribed Displacements--As many lines as required

a. Contents

```
'PREscribed' 'DISplacements'   M
N   ['X'   DXN]   ['Y'  DYN]
.     .     .        .     .    M lines
.     .     .        .     .
```

b. Definitions

M = number of nodes with prescribed displacements in structure

N = node number in structure

DXN = prescribed horizontal displacement at node N

DYN = prescribed vertical displacement at node N.

5. Prescribed Non-zero Concentrated Loads--As many lines as required

a. Contents

```
'PREscribed'  'LOAds'   M
N   ['X'   FXN]   ['Y'  FYN]
.     .     .       .    .   M lines
.     .     .       .    .
```

b. Definitions

M = number of nodes with prescribed non-zero concentrated load in structure

N = node number in structure

FXN = prescribed horizontal load at node N

FYN = prescribed vertical load at node N.

C. Interface Input:

1. Interface Properties--One line

a. Contents

```
'INTerface'  'PROperties'  SN   SS
```

b. Definitions

SN = unit normal stiffness at interface

SS = unit shear stiffness at interface.

2. Node Coordinates--As many lines as required, same as B.2.

3. Element Connectivities--As many lines as required

a. Contents

```
'ELEment'  'CONnectivities'   M
 N    IN1    IN2
 .     .      .   M lines
 .     .      .
```

b. Definitions

M = number of elements at interface

N = element number at interface

IN1 = the first node number in element N

IN2 = the second node number in element N.

4. Prescribed Displacements--As many lines as required, same as B.4.

5. Prescribed Non-zero Concentrated Loads--As many lines as required, same as B.5.

D. Soil Input:

1. Soil Properties--One line

   a. Contents

      'SOIl' 'PROperties' E V D

   b. Definitions

      E = modulus of elasticity for soil

      V = Poisson's ratio for soil

      D = density of soil.

2. Node Coordinates--As many lines as required, same as B.2.

3. Element Connectivities--As many lines as required

   a. Contents

```
'ELEment'  'CONnectivities'   M
N    IN1    IN2    IN3    IN4
.     .      .      .      .    M lines
.     .      .      .      .
```

   b. Definitions

      M = number of isoparametric quadrilateral elements in soil

      N = element number in soil

      IN1 = the first node number in element N

IN2 = the second node number in element N
IN3 = the third node number in element N
IN4 = the fourth node number in element N.

4. Prescribed Displacements--As many lines as required, same as B.4.

5. Prescribed Non-zero Concentrated Loads--As many lines as required, same as B.5.

## 5.3 Output Information

Output data are provided in two parts in programs BOUFIN and FINITE. The first part is the echoprint which contains a tabular listing of all input data for heading, structure, interface, and soil sections, respectively. The second part contains the complete results for the specified Soil Streucture Interaction problem. The quantities involved in the results are displacements (inch) and stresses (ksi). The positive senses for displacements and stresses are shown in Figure 16.

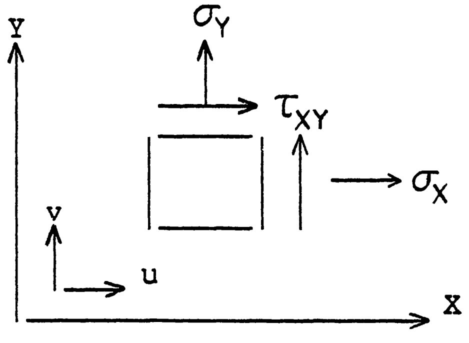


Figure 16. Positive Senses for Stresses and Displacements

# CHAPTER VI

# COMPARISON OF RESULTS

## 6.1 Introduction

In order to illustrate the solution capability of program BOUFIN, the accuracy of program FINITE must be assured before comparison. For this purpose, test problems without performing iterations on joint elements were solved by the general purpose program STRUDL (1) and by program FINITE; identical results were obtained. An additional test problem where tensile stresses were present along the soil/structure interface was analyzed by program FINITE. The results of this solution must satisfy the limitation of no tensile stresses across soil/structure interface. The convergence of normal and tangential stresses of joint elements in each iteration is discussed in section 6.2. The accuracy of program FINITE is thus proven.

An example problem of a U-Frame structure was solved using programs BOUFIN and FINITE. The validity of the coupling of boundary and finite element methods was examined by comparing the results of nodal displacements and element stresses in the structure, at the interface, and in the soil mass. A further discussion for both numerical techniques is also included.

## 6.2 Load Transfer Test

The load transfer method (see subsection 3.4.4), which provides an effective treatment to simulate the real properties of joint elements, is incorporated in programs BOUFIN and FINITE. In the load transfer subroutine, the number of iterations is controlled by a designated small value. This value defines the tolerable limit of absolute difference for each displacement at the interface between two consecutive iterations.

A test problem analyzed by program FINITE with tolerable limit equal to 1 x $10^{-8}$ was solved to demonstrate the way that the joint elements approach the so called "no tension" state by the load transfer method. The configuration, system properties, and the numbering of joint elements of the test problem are shown in Figure 17. The results of the analysis are given in Appendix C. The normal and tangential stresses of joint elements in each iteration are listed in Tables I and II. Since joint element 1 exhibits tensile stress (i.e. positive normal stress) in the first solution, a number of iterations are executed to eliminate the undesirable stresses. Note that tensile stresses still develop at the end of each iteration but are much reduced when compared to the previous solution. After eight iterations the normal and tangential stresses at joint element 1 are reduced to negligible quantities while the stresses in the remainder of the joint elements are converged to definite values. Therefore, the "no tension state" in the interface is reached and

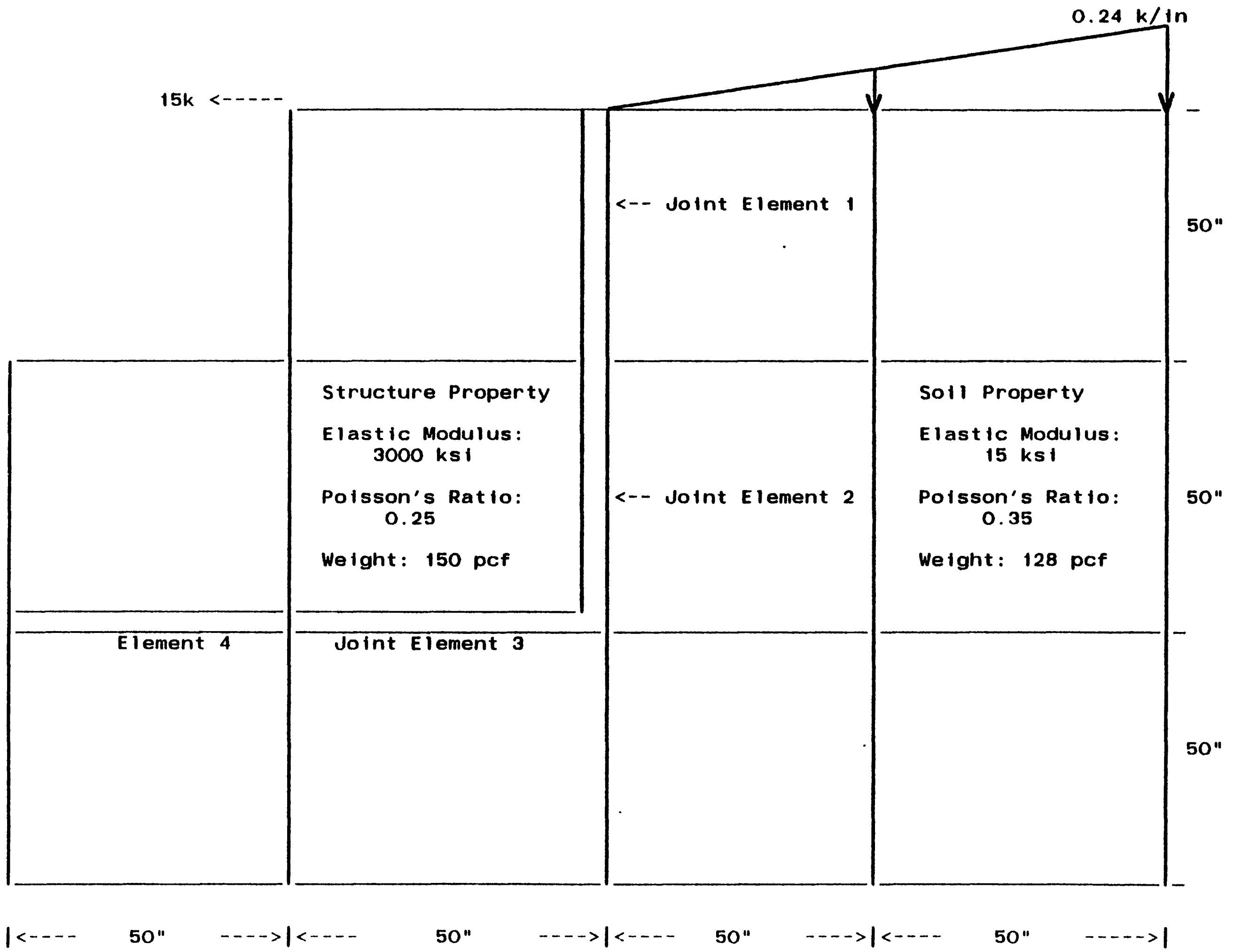


Figure 17. A Test Problem for Load Transfer Method

## TABLE I

## NORMAL STRESSES OF JOINT ELEMENTS FOR EACH ITERATION

| Iteration No. | Joint 1 (ksi) | Joint 2 (ksi) | Joint 3 (ksi) | Joint 4 (ksi) |
|---|---|---|---|---|
| 0 | 0.016047320 | -0.1177049 | -0.0250310 | -0.0610852 |
| 1 | 0.002262905 | -0.1196641 | -0.0203801 | -0.0621226 |
| 2 | 0.000322616 | -0.1199396 | -0.0196600 | -0.0622832 |
| 3 | 0.000046477 | -0.1199788 | -0.0195485 | -0.0623081 |
| 4 | 0.000006761 | -0.1199844 | -0.0195313 | -0.0623119 |
| 5 | 0.000000992 | -0.1199852 | -0.0195286 | -0.0623125 |
| 6 | 0.000000147 | -0.1199854 | -0.0195282 | -0.0623126 |
| 7 | 0.000000022 | -0.1199854 | -0.0195281 | -0.0623126 |
| 8 (Results) | 0.000000003 | -0.1200 | -0.01953 | -0.06231 |

# TABLE II

## TANGENTIAL STRESSES OF JOINT ELEMENTS ELEMENTS FOR EACH ITERATION

| Iteration No. | Joint 1 (ksi) | Joint 2 (ksi) | Joint 3 (ksi) | Joint 4 (ksi) |
|---|---|---|---|---|
| 0 | 0.056051976 | 0.0170428 | 0.0046481 | 0.0042666 |
| 1 | 0.008678506 | 0.0047507 | 0.0046240 | 0.0042536 |
| 2 | 0.001343685 | 0.0028476 | 0.0046204 | 0.0042516 |
| 3 | 0.000208041 | 0.0025529 | 0.0046199 | 0.0042513 |
| 4 | 0.000032211 | 0.0025073 | 0.0046198 | 0.0042512 |
| 5 | 0.000004987 | 0.0025002 | 0.0046198 | 0.0042512 |
| 6 | 0.000000772 | 0.0024991 | 0.0046198 | 0.0042512 |
| 7 | 0.000000119 | 0.0024990 | 0.0046198 | 0.0042512 |
| 8 (Results) | 0.000000018 | 0.002499 | 0.004620 | 0.004251 |

the validity of the load transfer method is verified.

## 6.3 Comparison of Example Solutions

### 6.3.1 Description of Example Soil-Structure System

A typical U-Frame structure was analyzed by programs BOUFIN and FINITE. The idealizations of the symmetric U-Frame structure, surrounding soil, and prescribed traction/displacement boundaries for the coupling of boundary and finite element methods and the finite element method alone are shown in Figures 18 and 19. The assumptions of negligible horizontal and vertical displacements at a sufficient distance from the structure center line and ground surface have been taken into account.

The specified soil-structure system is treated as a plane strain type problem since it involves a long body whose geometry and loading do not vary significantly in the longitudinal direction. The forces applied to the system are water pressure, weights of soil and structure, and prescribed loads/tractions. Water pressure is linearly distributed along the vertical wall of the U-Frame and becomes a constant at the floor slab. The properties of weight, elastic modulus, shear modulus, and Poisson's ratio for the U-Frame and soil are listed in Table III based on the assumptions that the U-Frame is composed of normal weight concrete with f'c = 3000 psi and that the surrounding soil is composed of dense sand.

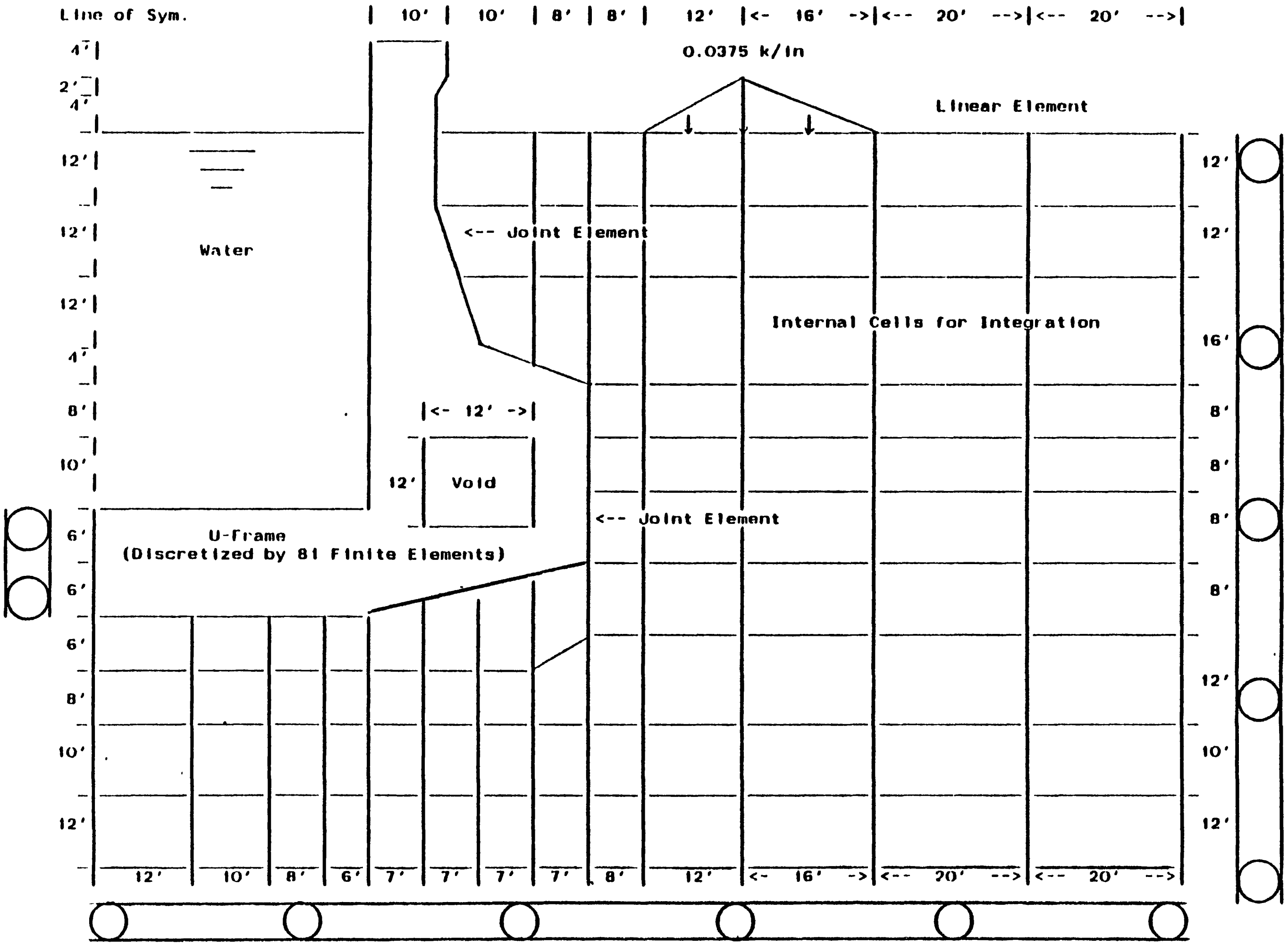


Figure 18. Idealization of U-Frame-Soil System by the Coupling of Boundary and Finite Element Methods

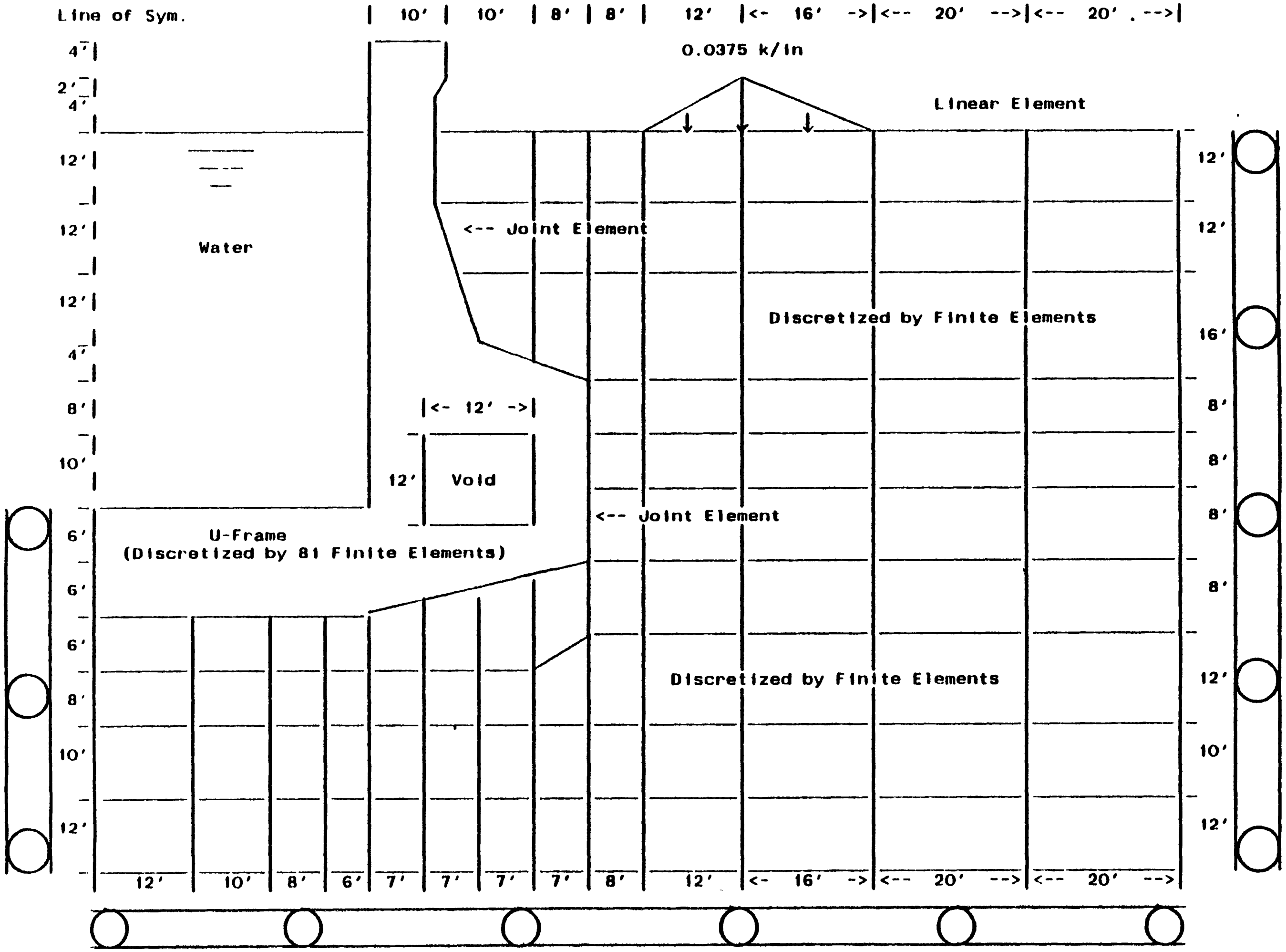


Figure 19. Idealization of U-Frame-Soil System by the Finite Element Method Alone

TABLE III

U-FRAME AND SOIL PROPERTIES USED IN THE PROPOSED SOIL-STRUCTURE SYSTEM

| | Weight | Poisson's Ratio | Elastic Modulus | Shear Modulus |
|---|---|---|---|---|
| U-Frame | 150 pcf | 0.25 | 3000 ksi | 1200 ksi |
| Soil | 128 pcf | 0.35 | 15 ksi | 5.56 ksi |

Because the unit normal and shear stiffnesses of joint elements are unavailable, the elastic modulus of U-Frame is taken as the unit normal stiffness and the shear modulus of soil is taken as the unit shear stiffness.

### 6.3.2 Verification of Solution Convergence

The numerical solutions of the coupling of boundary and finite element methods and the finite element method alone can be improved by increasing the mesh sizes of the soil-structure system. However, if the tradeoff between cost and accuracy is taken into consideration, the optimum case may be the solution close to the exact one with lower cost.

In order to acquire the optimum solutions for the example problem, several computer runs using programs BOUFIN and FINITE were conducted and a critical point at the soil mass near the corner of the U-Frame was chosen to test solution convergence. When the displacements and stresses at the specified critical point are close in two consecutive runs the results obtained from the finer mesh were used for comparison.

### 6.3.3 Comparison of Nodal Displacements and Element Stresses

The example U-Frame structure was analyzed with 209 nodes and 221 elements by program BOUFIN, and 204 nodes and 185 elements by program FINITE when the solutions converged. Data input and output information are presented in Chapter V. Computer codings and input listings of programs BOUFIN and FINITE are given in Appendices A and B. The printout sheets for program BOUFIN with 209 nodes and 221 elements, and for program FINITE with 204 nodes and 185 elements are listed in Appendices D and E. All computations were carried out on the IBM 3081D computer.

Nodal displacements and element stresses calculated from programs BOUFIN and FINITE for the U-Frame-soil system are presented in the following tables. Element stresses are evaluated at the centroid of the element.

Tables IV, V, and VI list the results of nodal displacements for the U-Frame structure, the interface, and surrounding soil mass, respectively. Tables VII, VIII, and IX are the results of element stresses for the U-Frame structure, the interface (joint elements), and surrounding soil mass, respectively.

## TABLE IV

## RESULTS OF NODAL DISPLACEMENTS IN U-FRAME STRUCTURE

| Node No. | X (inch) | Y (inch) | u (inch) | | v (inch) | |
|---|---|---|---|---|---|---|
| | | | FINITE | BOUFIN | FINITE | BOUFIN |
| 1 | 0 | 576 | 0.000 | 0.000 | -1.203 | -1.200 |
| 2 | 0 | 540 | 0.000 | 0.000 | -1.202 | -1.199 |
| 3 | 0 | 504 | 0.000 | 0.000 | -1.202 | -1.199 |
| 4 | 0 | 468 | 0.000 | 0.000 | -1.202 | -1.199 |
| 5 | 0 | 432 | 0.000 | 0.000 | -1.203 | -1.200 |
| 6 | 144 | 576 | 0.014 | 0.014 | -1.221 | -1.218 |
| 7 | 144 | 540 | 0.006 | 0.006 | -1.220 | -1.217 |
| 8 | 144 | 504 | -0.002 | -0.002 | -1.219 | -1.216 |
| 9 | 144 | 468 | -0.011 | -0.011 | -1.219 | -1.216 |
| 10 | 144 | 432 | -0.019 | -0.019 | -1.220 | -1.217 |
| 11 | 264 | 576 | 0.025 | 0.025 | -1.261 | -1.258 |
| 12 | 264 | 540 | 0.010 | 0.010 | -1.260 | -1.257 |
| 13 | 264 | 504 | -0.004 | -0.004 | -1.259 | -1.257 |
| 14 | 264 | 468 | -0.019 | -0.019 | -1.259 | -1.257 |
| 15 | 264 | 432 | -0.033 | -0.033 | -1.260 | -1.257 |
| 16 | 360 | 576 | 0.031 | 0.031 | -1.307 | -1.305 |
| 17 | 360 | 540 | 0.012 | 0.012 | -1.307 | -1.304 |
| 18 | 360 | 504 | -0.006 | -0.006 | -1.306 | -1.304 |
| 19 | 360 | 468 | -0.024 | -0.024 | -1.306 | -1.304 |
| 20 | 360 | 432 | -0.043 | -0.043 | -1.306 | -1.304 |
| 21 | 432 | 1392 | 0.579 | 0.594 | -1.353 | -1.350 |
| 22 | 432 | 1272 | 0.488 | 0.500 | -1.353 | -1.350 |
| 23 | 432 | 1128 | 0.379 | 0.389 | -1.354 | -1.351 |
| 24 | 432 | 984 | 0.279 | 0.286 | -1.355 | -1.352 |
| 25 | 432 | 840 | 0.188 | 0.191 | -1.355 | -1.353 |
| 26 | 432 | 768 | 0.144 | 0.146 | -1.355 | -1.353 |
| 27 | 432 | 696 | 0.101 | 0.103 | -1.354 | -1.352 |
| 28 | 432 | 576 | 0.033 | 0.034 | -1.350 | -1.348 |
| 29 | 432 | 540 | 0.013 | 0.013 | -1.348 | -1.346 |
| 30 | 432 | 504 | -0.007 | -0.007 | -1.348 | -1.345 |
| 31 | 432 | 468 | -0.027 | -0.027 | -1.347 | -1.345 |
| 32 | 432 | 432 | -0.047 | -0.047 | -1.347 | -1.345 |
| 33 | 456 | 1392 | 0.579 | 0.594 | -1.372 | -1.369 |
| 34 | 456 | 1272 | 0.488 | 0.500 | -1.371 | -1.369 |
| 35 | 456 | 1128 | 0.379 | 0.389 | -1.371 | -1.369 |
| 36 | 462 | 984 | 0.279 | 0.285 | -1.375 | -1.373 |
| 37 | 468 | 840 | 0.188 | 0.191 | -1.378 | -1.376 |
| 38 | 468 | 768 | 0.144 | 0.146 | -1.377 | -1.375 |
| 39 | 480 | 696 | 0.101 | 0.103 | -1.382 | -1.380 |
| 40 | 480 | 624 | 0.061 | 0.062 | -1.380 | -1.378 |
| 41 | 480 | 576 | 0.034 | 0.034 | -1.378 | -1.376 |
| 42 | 480 | 528 | 0.006 | 0.006 | -1.376 | -1.375 |

TABLE IV (Continued)

| Node No. | X (inch) | Y (inch) | u (inch) |  | v (inch) |  |
|---|---|---|---|---|---|---|
|  |  |  | FINITE | BOUFIN | FINITE | BOUFIN |
| 43 | 504 | 480 | -0.022 | -0.022 | -1.389 | -1.388 |
| 44 | 516 | 450 | -0.040 | -0.040 | -1.396 | -1.395 |
| 45 | 480 | 1392 | 0.579 | 0.594 | -1.390 | -1.388 |
| 46 | 480 | 1272 | 0.488 | 0.500 | -1.390 | -1.388 |
| 47 | 480 | 1128 | 0.379 | 0.389 | -1.389 | -1.387 |
| 48 | 492 | 984 | 0.279 | 0.285 | -1.395 | -1.393 |
| 49 | 504 | 840 | 0.188 | 0.191 | -1.400 | -1.399 |
| 50 | 504 | 768 | 0.144 | 0.147 | -1.399 | -1.397 |
| 51 | 528 | 696 | 0.101 | 0.103 | -1.412 | -1.410 |
| 52 | 528 | 648 | 0.075 | 0.076 | -1.408 | -1.407 |
| 53 | 528 | 600 | 0.048 | 0.049 | -1.406 | -1.405 |
| 54 | 528 | 552 | 0.020 | 0.021 | -1.404 | -1.403 |
| 55 | 576 | 504 | -0.009 | -0.009 | -1.433 | -1.432 |
| 56 | 600 | 468 | -0.032 | -0.032 | -1.448 | -1.447 |
| 57 | 504 | 1392 | 0.579 | 0.594 | -1.408 | -1.407 |
| 58 | 504 | 1272 | 0.488 | 0.500 | -1.408 | -1.406 |
| 59 | 504 | 1128 | 0.379 | 0.389 | -1.406 | -1.405 |
| 60 | 522 | 984 | 0.279 | 0.285 | -1.415 | -1.414 |
| 61 | 540 | 840 | 0.188 | 0.191 | -1.423 | -1.422 |
| 62 | 540 | 768 | 0.144 | 0.147 | -1.421 | -1.420 |
| 63 | 564 | 696 | 0.100 | 0.102 | -1.436 | -1.435 |
| 64 | 600 | 552 | 0.019 | 0.020 | -1.448 | -1.447 |
| 65 | 648 | 504 | -0.011 | -0.011 | -1.478 | -1.478 |
| 66 | 684 | 486 | -0.022 | -0.023 | -1.501 | -1.501 |
| 67 | 552 | 1392 | 0.579 | 0.594 | -1.445 | -1.444 |
| 68 | 552 | 1344 | 0.543 | 0.556 | -1.445 | -1.444 |
| 69 | 528 | 1320 | 0.524 | 0.537 | -1.427 | -1.425 |
| 70 | 528 | 1272 | 0.488 | 0.500 | -1.427 | -1.425 |
| 71 | 528 | 1128 | 0.379 | 0.389 | -1.423 | -1.422 |
| 72 | 552 | 984 | 0.279 | 0.286 | -1.435 | -1.434 |
| 73 | 576 | 840 | 0.188 | 0.191 | -1.445 | -1.444 |
| 74 | 600 | 768 | 0.144 | 0.147 | -1.459 | -1.458 |
| 75 | 600 | 696 | 0.100 | 0.102 | -1.458 | -1.458 |
| 76 | 672 | 816 | 0.172 | 0.175 | -1.502 | -1.502 |
| 77 | 672 | 768 | 0.143 | 0.146 | -1.502 | -1.502 |
| 78 | 636 | 696 | 0.100 | 0.102 | -1.480 | -1.480 |
| 79 | 672 | 696 | 0.100 | 0.102 | -1.501 | -1.501 |
| 80 | 672 | 648 | 0.072 | 0.074 | -1.499 | -1.500 |
| 81 | 672 | 600 | 0.044 | 0.046 | -1.497 | -1.498 |
| 82 | 672 | 552 | 0.018 | 0.018 | -1.494 | -1.494 |
| 83 | 720 | 744 | 0.128 | 0.131 | -1.530 | -1.531 |
| 84 | 720 | 672 | 0.086 | 0.088 | -1.528 | -1.529 |
| 85 | 720 | 600 | 0.045 | 0.046 | -1.525 | -1.526 |
| 86 | 720 | 528 | 0.003 | 0.003 | -1.524 | -1.525 |
| 87 | 768 | 792 | 0.157 | 0.160 | -1.561 | -1.561 |

TABLE IV (Continued)

| Node No. | X (inch) | Y (inch) | u (inch) FINITE | u (inch) BOUFIN | v (inch) FINITE | v (inch) BOUFIN |
|---|---|---|---|---|---|---|
| 88 | 768 | 696 | 0.100 | 0.102 | -1.557 | -1.558 |
| 89 | 768 | 600 | 0.045 | 0.046 | -1.554 | -1.555 |
| 90 | 768 | 504 | -0.011 | -0.011 | -1.552 | -1.554 |

# TABLE V

## RESULTS OF NODAL DISPLACEMENTS AT INTERFACE

| Node No. | x (inch) | Y (inch) | u (inch) | | v (inch) | |
|---|---|---|---|---|---|---|
| | | | FINITE | BOUFIN | FINITE | BOUFIN |
| 1 | 528 | 1272 | 0.488 | 0.500 | -1.427 | -1.426 |
| 2 | 528 | 1128 | 0.379 | 0.389 | -1.423 | -1.423 |
| 3 | 552 | 984 | 0.280 | 0.286 | -1.437 | -1.436 |
| 4 | 576 | 840 | 0.188 | 0.191 | -1.445 | -1.444 |
| 5 | 672 | 816 | 0.170 | 0.174 | -1.502 | -1.502 |
| 6 | 768 | 792 | 0.157 | 0.160 | -1.561 | -1.561 |
| 7 | 768 | 696 | 0.100 | 0.102 | -1.560 | -1.562 |
| 8 | 768 | 600 | 0.045 | 0.046 | -1.555 | -1.554 |
| 9 | 768 | 504 | -0.011 | -0.011 | -1.552 | -1.554 |
| 10 | 684 | 486 | -0.021 | -0.022 | -1.500 | -1.501 |
| 11 | 600 | 468 | -0.031 | -0.031 | -1.448 | -1.447 |
| 12 | 516 | 450 | -0.039 | -0.039 | -1.396 | -1.395 |
| 13 | 432 | 432 | -0.047 | -0.047 | -1.347 | -1.345 |
| 14 | 360 | 432 | -0.042 | -0.043 | -1.306 | -1.304 |
| 15 | 264 | 432 | -0.033 | -0.033 | -1.260 | -1.257 |
| 16 | 144 | 432 | -0.019 | -0.019 | -1.220 | -1.217 |
| 17 | 0 | 432 | 0.000 | -0.000 | -1.203 | -1.200 |

## TABLE VI

## RESULTS OF NODAL DISPLACEMENTS IN SOIL MASS

| Node No. | x (inch) | Y (inch) | u (inch) |  | v (inch) |  |
|---|---|---|---|---|---|---|
|  |  |  | FINITE | BOUFIN | FINITE | BOUFIN |
| 1 | 0 | 360 | 0.000 | +0.000 | -1.042 | -1.041 |
| 2 | 0 | 264 | 0.000 | +0.000 | -0.804 | -0.803 |
| 3 | 0 | 144 | 0.000 | +0.000 | -0.466 | -0.466 |
| 4 | 0 | 0 | 0.000 | +0.000 | 0.000 | 0.000 |
| 5 | 144 | 0 | -0.035 | -0.035 | 0.000 | 0.000 |
| 6 | 264 | 0 | -0.058 | -0.059 | 0.000 | 0.000 |
| 7 | 360 | 0 | -0.071 | -0.072 | 0.000 | 0.000 |
| 8 | 432 | 0 | -0.077 | -0.078 | 0.000 | 0.000 |
| 9 | 516 | 0 | -0.081 | -0.083 | 0.000 | 0.000 |
| 10 | 600 | 0 | -0.084 | -0.085 | 0.000 | 0.000 |
| 11 | 684 | 0 | -0.085 | -0.087 | 0.000 | 0.000 |
| 12 | 768 | 0 | -0.086 | -0.089 | 0.000 | 0.000 |
| 13 | 864 | 0 | -0.088 | -0.091 | 0.000 | 0.000 |
| 14 | 1008 | 0 | -0.088 | -0.092 | 0.000 | 0.000 |
| 15 | 1200 | 0 | -0.078 | -0.081 | 0.000 | 0.000 |
| 16 | 1440 | 0 | -0.046 | -0.047 | 0.000 | 0.000 |
| 17 | 1680 | 0 | 0.000 | -0.000 | 0.000 | 0.000 |
| 18 | 1680 | 144 | 0.000 | 0.000 | -0.567 | -0.567 |
| 19 | 1680 | 264 | 0.000 | 0.000 | -0.990 | -0.991 |
| 20 | 1680 | 408 | 0.000 | 0.000 | -1.438 | -1.440 |
| 21 | 1680 | 504 | 0.000 | 0.000 | -1.699 | -1.700 |
| 22 | 1680 | 600 | 0.000 | 0.000 | -1.928 | -1.928 |
| 23 | 1680 | 696 | 0.000 | 0.000 | -2.123 | -2.123 |
| 24 | 1680 | 792 | 0.000 | 0.000 | -2.283 | -2.283 |
| 25 | 1680 | 984 | 0.000 | 0.000 | -2.490 | -2.491 |
| 26 | 1680 | 1128 | 0.000 | 0.000 | -2.551 | -2.553 |
| 27 | 1680 | 1272 | 0.000 | 0.000 | -2.543 | -2.544 |
| 28 | 1440 | 1272 | 0.131 | 0.129 | -2.544 | -2.544 |
| 29 | 1200 | 1272 | 0.289 | 0.273 | -2.609 | -2.607 |
| 30 | 1008 | 1272 | 0.438 | 0.452 | -2.705 | -2.742 |
| 31 | 864 | 1272 | 0.537 | 0.571 | -2.262 | -2.261 |
| 32 | 768 | 1272 | 0.569 | 0.592 | -1.991 | -1.994 |
| 33 | 672 | 1272 | 0.564 | 0.588 | -1.770 | -1.768 |

# TABLE VII

## RESULTS OF ELEMENT STRESSES IN U-FRAME STRUCTURE

| Elm. No. | Centroid (inch) |  | Sigma X (ksi) |  | Sigma Y (ksi) |  | Tau XY (ksi) |  |
|---|---|---|---|---|---|---|---|---|
|  | X | Y | FINITE | BOUFIN | FINITE | BOUFIN | FINITE | BOUFIN |
| 1 | 72 | 558 | .2125 | .2138 | -.0277 | -.0280 | -.0070 | -.0070 |
| 2 | 72 | 522 | .0274 | .0281 | -.0354 | -.0348 | -.0087 | -.0086 |
| 3 | 72 | 486 | -.1569 | -.1572 | -.0433 | -.0430 | -.0085 | -.0085 |
| 4 | 72 | 450 | -.3419 | -.3431 | -.0506 | -.0505 | -.0070 | -.0070 |
| 5 | 204 | 558 | .1899 | .1920 | -.0273 | -.0274 | -.0184 | -.0182 |
| 6 | 204 | 522 | .0192 | .0200 | -.0341 | -.0340 | -.0265 | -.0261 |
| 7 | 204 | 486 | -.1491 | -.1495 | -.0438 | -.0436 | -.0271 | -.0266 |
| 8 | 204 | 450 | -.3204 | -.3220 | -.0528 | -.0523 | -.0186 | -.0183 |
| 9 | 312 | 558 | .1310 | .1344 | -.0234 | -.0238 | -.0221 | -.0219 |
| 10 | 312 | 522 | -.0054 | -.0039 | -.0348 | -.0345 | -.0458 | -.0450 |
| 11 | 312 | 486 | -.1308 | -.1317 | -.0456 | -.0453 | -.0476 | -.0470 |
| 12 | 312 | 450 | -.2580 | -.2612 | -.0532 | -.0531 | -.0286 | -.0283 |
| 13 | 396 | 558 | .0440 | .0496 | -.0628 | -.0611 | -.0448 | -.0433 |
| 14 | 396 | 522 | -.0291 | -.0279 | -.0654 | -.0641 | -.0592 | -.0583 |
| 15 | 396 | 486 | -.1050 | -.1066 | -.0604 | -.0596 | -.0559 | -.0556 |
| 16 | 396 | 450 | -.1758 | -.1800 | -.0605 | -.0603 | -.0318 | -.0318 |
| 17 | 444 | 1332 | .0004 | .0003 | -.0038 | -.0042 | .0011 | .0010 |
| 18 | 444 | 1200 | -.0025 | -.0024 | .0020 | .0017 | .0086 | .0085 |
| 19 | 446 | 1056 | -.0071 | -.0072 | -.0112 | -.0121 | .0031 | .0041 |
| 20 | 448 | 912 | -.0141 | -.0144 | -.0137 | -.0086 | -.0074 | -.0065 |
| 21 | 450 | 804 | -.0155 | -.0154 | -.0406 | -.0335 | -.0054 | -.0065 |
| 22 | 453 | 732 | -.0199 | -.0198 | -.0654 | -.0615 | -.0000 | -.0004 |
| 23 | 456 | 648 | -.0302 | -.0303 | -.1251 | -.1206 | -.0273 | -.0268 |
| 24 | 456 | 579 | -.0181 | -.0145 | -.1245 | -.1223 | -.0261 | -.0256 |
| 25 | 456 | 537 | -.0200 | -.0178 | -.1018 | -.1008 | -.0299 | -.0306 |
| 26 | 465 | 495 | -.0796 | -.0810 | -.0794 | -.0786 | -.0295 | -.0301 |
| 27 | 471 | 458 | -.1212 | -.1250 | -.0624 | -.0617 | -.0238 | -.0246 |
| 28 | 468 | 1332 | .0021 | .0019 | -.0060 | -.0059 | .0015 | .0013 |
| 29 | 468 | 1200 | -.0011 | -.0010 | -.0161 | -.0154 | .0116 | .0114 |
| 30 | 472 | 1056 | -.0104 | -.0100 | -.0262 | -.0256 | .0046 | .0059 |
| 31 | 482 | 912 | -.0164 | -.0172 | -.0450 | -.0424 | -.0095 | -.0079 |
| 32 | 486 | 804 | -.0105 | -.0110 | -.0639 | -.0609 | -.0136 | -.0155 |
| 33 | 495 | 732 | -.0348 | -.0332 | -.0985 | -.0975 | -.0208 | -.0221 |
| 34 | 504 | 666 | -.0373 | -.0370 | -.1659 | -.1662 | -.0454 | -.0453 |
| 35 | 504 | 612 | -.0038 | -.0037 | -.1330 | -.1348 | -.0102 | -.0098 |
| 36 | 504 | 564 | -.0113 | -.0086 | -.1176 | -.1192 | -.0016 | -.0024 |
| 37 | 522 | 516 | -.0641 | -.0639 | -.0511 | -.0506 | -.0058 | -.0065 |
| 38 | 549 | 476 | -.1122 | -.1161 | -.0560 | -.0563 | -.0169 | -.0179 |
| 39 | 492 | 1332 | .0039 | .0037 | -.0076 | -.0072 | .0002 | .0003 |
| 40 | 492 | 1200 | .0003 | .0007 | -.0352 | -.0331 | .0111 | .0109 |
| 41 | 500 | 1056 | -.0134 | -.0127 | -.0626 | -.0627 | .0062 | .0073 |
| 42 | 515 | 912 | -.0200 | -.0217 | -.0763 | -.0761 | -.0066 | -.0042 |
| 43 | 522 | 804 | -.0147 | -.0162 | -.0759 | -.0758 | -.0203 | -.0228 |

TABLE VII (Continued)

| Elm. No. | Centroid (inch) |  | Sigma X (ksi) |  | Sigma Y (ksi) |  | Tau XY (ksi) |  |
|---|---|---|---|---|---|---|---|---|
|  | X | Y | FINITE | BOUFIN | FINITE | BOUFIN | FINITE | BOUFIN |
| 44 | 534 | 732 | -.0455 | -.0417 | -.0957 | -.0970 | -.0497 | -.0518 |
| 45 | 588 | 528 | -.0666 | -.0651 | -.0321 | -.0326 | -.0132 | -.0135 |
| 46 | 627 | 491 | -.0860 | -.0898 | -.0656 | -.0662 | -.0249 | -.0256 |
| 47 | 534 | 1362 | +.0000 | .0001 | -.0043 | -.0042 | -.0001 | -.0000 |
| 48 | 516 | 1314 | .0067 | .0062 | -.0058 | -.0060 | -.0040 | -.0034 |
| 49 | 516 | 1200 | .0022 | .0028 | -.0559 | -.0518 | .0071 | .0071 |
| 50 | 526 | 1056 | -.0151 | -.0140 | -.0997 | -.1011 | .0075 | .0083 |
| 51 | 548 | 912 | -.0250 | -.0277 | -.1103 | -.1126 | .0030 | .0057 |
| 52 | 564 | 804 | -.0331 | -.0354 | -.0671 | -.0728 | -.0142 | -.0164 |
| 53 | 576 | 732 | -.0338 | -.0291 | -.0289 | -.0298 | -.0236 | -.0244 |
| 54 | 660 | 534 | -.0798 | -.0772 | -.0760 | -.0773 | -.0398 | -.0404 |
| 55 | 705 | 506 | -.0441 | -.0433 | -.0844 | -.0818 | -.0077 | -.0078 |
| 56 | 630 | 798 | -.0527 | -.0553 | -.0348 | -.0382 | .0055 | .0075 |
| 57 | 627 | 732 | -.0334 | -.0292 | -.0183 | -.0189 | .0025 | .0045 |
| 58 | 708 | 780 | -.0387 | -.0358 | -.0621 | -.0573 | -.0072 | -.0028 |
| 59 | 675 | 726 | -.0289 | -.0259 | -.0516 | -.0518 | .0124 | .0150 |
| 60 | 696 | 690 | -.0210 | -.0207 | -.1025 | -.0998 | .0066 | .0081 |
| 61 | 744 | 726 | -.0271 | -.0271 | -.1106 | -.0981 | -.0083 | -.0071 |
| 62 | 696 | 630 | -.0071 | -.0075 | -.1284 | -.1249 | -.0150 | -.0154 |
| 63 | 744 | 642 | -.0271 | -.0295 | -.1225 | -.1209 | -.0185 | -.0185 |
| 64 | 696 | 570 | -.0608 | -.0603 | -.1542 | -.1519 | -.0449 | -.0452 |
| 65 | 744 | 558 | -.0403 | -.0390 | -.0897 | -.0762 | -.0130 | -.0129 |

## TABLE VIII

## RESULTS OF ELEMENT STRESSES AT INTERFACE

| Elm. No. | Centroid (inch) | | Normal Stress (ksi) | | Tangential Stress (ksi) | |
|---|---|---|---|---|---|---|
| | X | Y | FINITE | BOUFIN | FINITE | BOUFIN |
| 1 | 528 | 1200 | -.0099 | -.0074 | .0020 | .0016 |
| 2 | 540 | 1056 | -.0113 | -.0234 | .0066 | .0061 |
| 3 | 564 | 912 | -.0196 | -.0162 | .0069 | .0060 |
| 4 | 624 | 828 | -.0215 | -.0353 | -.0046 | -.0041 |
| 5 | 720 | 804 | -.0786 | -.0716 | -.0046 | -.0040 |
| 6 | 768 | 744 | -.0227 | -.0261 | .0071 | .0109 |
| 7 | 768 | 648 | -.0343 | -.0359 | .0090 | .0082 |
| 8 | 768 | 552 | -.0256 | -.0269 | -.0020 | -.0027 |
| 9 | 726 | 495 | -.0847 | -.0536 | -.0024 | -.0012 |
| 10 | 642 | 477 | -.0483 | -.0283 | -.0044 | -.0035 |
| 11 | 558 | 459 | -.0629 | -.0509 | -.0053 | -.0058 |
| 12 | 474 | 441 | -.0445 | -.0403 | -.0036 | -.0037 |
| 13 | 396 | 432 | -.0792 | -.0705 | -.0010 | -.0010 |
| 14 | 312 | 432 | -.0523 | -.0495 | -.0011 | -.0011 |
| 15 | 204 | 432 | -.0556 | -.0516 | -.0007 | -.0008 |
| 16 | 72 | 432 | -.0532 | -.0404 | -.0002 | .0004 |

## TABLE IX

## RESULTS OF ELEMENT STRESSES IN SOIL MASS

| Elm. No. | Centroid (inch) | | Sigma X (ksi) | | Sigma Y (ksi) | | Tau XY (ksi) | |
|---|---|---|---|---|---|---|---|---|
| | X | Y | FINITE | BOUFIN | FINITE | BOUFIN | FINITE | BOUFIN |
| 1 | 72 | 396 | -.0330 | -.0479 | -.0562 | -.0358 | -.0003 | -.0012 |
| 2 | 72 | 312 | -.0374 | -.0374 | -.0628 | -.0627 | -.0003 | -.0003 |
| 3 | 72 | 204 | -.0424 | -.0424 | -.0713 | -.0711 | -.0003 | -.0003 |
| 4 | 72 | 72 | -.0479 | -.0480 | -.0815 | -.0812 | -.0001 | -.0001 |
| 5 | 204 | 396 | -.0333 | -.0404 | -.0574 | -.0479 | -.0009 | -.0011 |
| 6 | 204 | 312 | -.0374 | -.0374 | -.0640 | -.0639 | -.0010 | -.0009 |
| 7 | 204 | 204 | -.0421 | -.0422 | -.0724 | -.0723 | -.0007 | -.0007 |
| 8 | 204 | 72 | -.0476 | -.0477 | -.0824 | -.0822 | -.0003 | -.0003 |
| 9 | 312 | 396 | -.0337 | -.0357 | -.0595 | -.0569 | -.0014 | -.0015 |
| 10 | 312 | 312 | -.0372 | -.0372 | -.0660 | -.0658 | -.0013 | -.0013 |
| 11 | 312 | 204 | -.0417 | -.0417 | -.0739 | -.0738 | -.0008 | -.0008 |
| 12 | 312 | 72 | -.0472 | -.0472 | -.0835 | -.0835 | -.0003 | -.0003 |
| 13 | 396 | 396 | -.0342 | -.0341 | -.0628 | -.0622 | -.0017 | -.0017 |
| 14 | 396 | 312 | -.0367 | -.0368 | -.0678 | -.0678 | -.0010 | -.0010 |
| 15 | 396 | 204 | -.0414 | -.0414 | -.0751 | -.0751 | -.0006 | -.0006 |
| 16 | 396 | 72 | -.0469 | -.0469 | -.0845 | -.0845 | -.0002 | -.0002 |
| 17 | 474 | 401 | -.0331 | -.0334 | -.0629 | -.0629 | .0003 | .0008 |
| 18 | 474 | 312 | -.0368 | -.0368 | -.0686 | -.0687 | -.0000 | -.0000 |
| 19 | 474 | 204 | -.0413 | -.0413 | -.0759 | -.0759 | -.0002 | -.0002 |
| 20 | 474 | 72 | -.0467 | -.0468 | -.0853 | -.0853 | -.0001 | -.0001 |
| 21 | 558 | 410 | -.0329 | -.0329 | -.0617 | -.0617 | .0006 | .0005 |
| 22 | 558 | 312 | -.0371 | -.0370 | -.0688 | -.0688 | .0003 | .0003 |
| 23 | 558 | 204 | -.0414 | -.0414 | -.0765 | -.0764 | .0001 | .0001 |
| 24 | 558 | 72 | -.0467 | -.0467 | -.0859 | -.0859 | +.0000 | +.0000 |
| 25 | 600 | 1200 | .0014 | .0012 | -.0015 | -.0025 | -.0062 | -.0062 |
| 26 | 606 | 1056 | -.0138 | -.0137 | -.0130 | -.0116 | -.0074 | -.0079 |
| 27 | 618 | 906 | -.0167 | -.0163 | -.0202 | -.0219 | -.0058 | -.0061 |
| 28 | 642 | 419 | -.0328 | -.0328 | -.0615 | -.0618 | .0009 | .0008 |
| 29 | 642 | 312 | -.0375 | -.0371 | -.0693 | -.0691 | .0003 | .0006 |
| 30 | 642 | 204 | -.0414 | -.0415 | -.0768 | -.0768 | .0004 | .0004 |
| 31 | 642 | 72 | -.0468 | -.0468 | -.0864 | -.0863 | .0001 | .0001 |
| 32 | 720 | 1200 | -.0050 | -.0050 | -.0057 | -.0052 | -.0041 | -.0040 |
| 33 | 720 | 1056 | -.0119 | -.0135 | -.0159 | -.0164 | -.0079 | -.0082 |
| 34 | 720 | 894 | -.0200 | -.0177 | -.0340 | -.0341 | -.0069 | -.0078 |
| 35 | 726 | 440 | -.0331 | -.0323 | -.0614 | -.0614 | .0013 | .0017 |
| 36 | 726 | 324 | -.0366 | -.0369 | -.0685 | -.0683 | .0009 | .0011 |
| 37 | 726 | 204 | -.0417 | -.0418 | -.0770 | -.0769 | .0005 | .0005 |
| 38 | 726 | 72 | -.0470 | -.0470 | -.0868 | -.0867 | .0001 | .0002 |
| 39 | 816 | 1200 | -.0096 | -.0108 | -.0084 | -.0078 | -.0053 | -.0053 |
| 40 | 816 | 1056 | -.0136 | -.0142 | -.0216 | -.0210 | -.0095 | -.0094 |
| 41 | 816 | 888 | -.0231 | -.0207 | -.0355 | -.0336 | -.0133 | -.0123 |
| 42 | 816 | 744 | -.0236 | -.0242 | -.0230 | -.0288 | -.0111 | -.0107 |
| 43 | 816 | 648 | -.0285 | -.0291 | -.0266 | -.0268 | -.0052 | -.0051 |
| 44 | 816 | 552 | -.0269 | -.0265 | -.0296 | -.0366 | .0017 | .0016 |

TABLE IX (Continued)

| Elm. No. | Centroid (inch) | | Sigma X (ksi) | | Sigma Y (ksi) | | Tau XY (ksi) | |
|---|---|---|---|---|---|---|---|---|
| | X | Y | FINITE | BOUFIN | FINITE | BOUFIN | FINITE | BOUFIN |
| 45 | 816 | 456 | -.0355 | -.0347 | -.0578 | -.0541 | .0038 | .0030 |
| 46 | 816 | 336 | -.0371 | -.0374 | -.0669 | -.0665 | .0015 | .0012 |
| 47 | 816 | 204 | -.0422 | -.0421 | -.0772 | -.0770 | .0005 | .0004 |
| 48 | 816 | 72 | -.0472 | -.0472 | -.0871 | -.0870 | .0001 | .0001 |
| 49 | 936 | 1200 | -.0170 | -.0163 | -.0213 | -.0220 | -.0071 | -.0080 |
| 50 | 936 | 1056 | -.0149 | -.0144 | -.0268 | -.0272 | -.0077 | -.0075 |
| 51 | 936 | 888 | -.0201 | -.0206 | -.0315 | -.0321 | -.0064 | -.0065 |
| 52 | 936 | 744 | -.0234 | -.0232 | -.0367 | -.0373 | -.0033 | -.0033 |
| 53 | 936 | 648 | -.0235 | -.0241 | -.0401 | -.0423 | -.0026 | -.0025 |
| 54 | 936 | 552 | -.0275 | -.0271 | -.0482 | -.0489 | -.0024 | -.0024 |
| 55 | 936 | 456 | -.0319 | -.0325 | -.0551 | -.0562 | -.0018 | -.0017 |
| 56 | 936 | 336 | -.0377 | -.0375 | -.0668 | -.0666 | -.0003 | -.0004 |
| 57 | 936 | 204 | -.0423 | -.0423 | -.0775 | -.0774 | +.0000 | -.0001 |
| 58 | 936 | 72 | -.0473 | -.0473 | -.0876 | -.0875 | +.0000 | +.0000 |
| 59 | 1104 | 1200 | -.0191 | -.0207 | -.0208 | -.0187 | .0037 | .0053 |
| 60 | 1104 | 1056 | -.0160 | -.0159 | -.0257 | -.0273 | .0010 | .0016 |
| 61 | 1104 | 888 | -.0193 | -.0193 | -.0333 | -.0342 | -.0004 | -.0002 |
| 62 | 1104 | 744 | -.0223 | -.0223 | -.0412 | -.0422 | -.0003 | -.0003 |
| 63 | 1104 | 648 | -.0245 | -.0245 | -.0477 | -.0482 | -.0008 | -.0007 |
| 64 | 1104 | 552 | -.0274 | -.0276 | -.0538 | -.0544 | -.0014 | -.0012 |
| 65 | 1104 | 456 | -.0315 | -.0315 | -.0605 | -.0608 | -.0016 | -.0014 |
| 66 | 1104 | 336 | -.0365 | -.0365 | -.0689 | -.0693 | -.0012 | -.0011 |
| 67 | 1104 | 204 | -.0419 | -.0418 | -.0790 | -.0791 | -.0006 | -.0006 |
| 68 | 1104 | 72 | -.0471 | -.0472 | -.0889 | -.0886 | -.0002 | -.0001 |
| 69 | 1320 | 1200 | -.0144 | -.0217 | -.0074 | -.0027 | .0016 | .0018 |
| 70 | 1320 | 1056 | -.0180 | -.0182 | -.0199 | -.0188 | .0028 | .0027 |
| 71 | 1320 | 888 | -.0205 | -.0207 | -.0322 | -.0321 | .0019 | .0022 |
| 72 | 1320 | 744 | -.0234 | -.0235 | -.0425 | -.0427 | .0011 | .0013 |
| 73 | 1320 | 648 | -.0258 | -.0259 | -.0494 | -.0496 | .0005 | .0006 |
| 74 | 1320 | 552 | -.0286 | -.0286 | -.0563 | -.0564 | -.0001 | .0001 |
| 75 | 1320 | 456 | -.0317 | -.0318 | -.0629 | -.0632 | -.0004 | -.0003 |
| 76 | 1320 | 336 | -.0362 | -.0362 | -.0714 | -.0717 | -.0006 | -.0005 |
| 77 | 1320 | 204 | -.0413 | -.0413 | -.0809 | -.0812 | -.0005 | -.0004 |
| 78 | 1320 | 72 | -.0465 | -.0470 | -.0906 | -.0900 | -.0002 | +.0000 |
| 79 | 1560 | 1200 | -.0122 | -.0192 | -.0052 | -.0037 | .0001 | .0003 |
| 80 | 1560 | 1056 | -.0172 | -.0173 | -.0166 | -.0167 | .0008 | .0007 |
| 81 | 1560 | 888 | -.0214 | -.0214 | -.0305 | -.0303 | .0009 | .0009 |
| 82 | 1560 | 744 | -.0245 | -.0246 | -.0420 | -.0418 | .0006 | .0007 |
| 83 | 1560 | 648 | -.0268 | -.0269 | -.0495 | -.0493 | .0004 | .0004 |
| 84 | 1560 | 552 | -.0294 | -.0295 | -.0567 | -.0566 | .0002 | .0002 |
| 85 | 1560 | 456 | -.0322 | -.0324 | -.0638 | -.0638 | -.0000 | .0001 |
| 86 | 1560 | 336 | -.0363 | -.0364 | -.0725 | -.0727 | -.0001 | -.0000 |
| 87 | 1560 | 204 | -.0411 | -.0412 | -.0821 | -.0823 | -.0001 | -.0001 |
| 88 | 1560 | 72 | -.0462 | -.0468 | -.0918 | -.0912 | -.0001 | .0001 |

## 6.4 Discussions and Conclusions

Tables IV through IX present the nodal displacements and element stresses for the example problem analyzed in a one inch width by programs BOUFIN and FINITE. The results obtained from the test run of program BOUFIN indicate excellent agreement with those from program FINITE. In addition, because the iterative scheme for simulating the interface behavior is adopted in both programs, the limitation of no tensile stresses across the soil/structure interface is satisfied in program BOUFIN. The accuracy of the coupling of boundary and finite element methods in solving Soil Structure Interaction problems is proven.

As the boundary element matrices are fully populated, the computer cost of program BOUFIN is greater than that of program FINITE in solving a problem with similar sizes of system matrices. This indicates that the coupling of boundary and finite element methods is less efficient computationally than the finite element method alone. However, due to the fact that the dimensionality of the boundary element region is reduced, the equations generated by the coupling of boundary and finite element methods are fewer than the equations generated by the finite element method alone in acquiring the same accuracy. This advantage is more evident for complex two- or three-dimensional continuum problems. For instance, in the example given in section 6.3, the size of the system matrix in BOUFIN (324x324) is much smaller than the size of the system matrix in FINITE (408x408);

hence, the computer execution time needed in program BOUFIN is about three-fourths of the computer execution time needed in program FINITE.

Another advantage of the coupling of boundary and finite element methods, which is not prominent in this study, is the required time in data preparation. The input data required for program BOUFIN can be simplified significantly if the domain integrals in evaluating body force terms are transformed into boundary integrals (31). In constrast, a large amount of data is needed for program FINITE. This is an important point as many man-hours are lost in preparing and checking finite element data.

# CHAPTER VII

# SUMMARY AND CONCLUSIONS

## 7.1 Summary

The direct boundary element method has been formulated in which the symmetry condition is introduced. The displacement approach of the finite element method has been reviewed where the joint stiffness formulation is included. The coupling of boundary and finite element methods is derived for both equivalent boundary and finite element approaches.

A computer program based on the coupling of boundary and finite element methods was developed to analyze the elastic behavior of a U-Frame structure. Another computer program based on the finite element method alone was developed to solve the same soil-structure system for comparison. Both structure and soil mass were assumed to be linearly elastic, isotropic, and homogeneous. The limitation of no transfer of tensile stresses across soil/structure interface was ensured by employing iterative schemes on joint elements.

## 7.2 Conclusions

An example problem of U-Frame structure has been analyzed using the two numerical methods indicated above. Comparison of the results of nodal displacements and element stresses in the soil-structure system obtained from both techniques demonstrates the accuracy and validity of the coupling of boundary and finite element methods. Due to the fact that the dimensionality of the boundary element region is reduced, the equations generated by the coupling of boundary and finite element methods are fewer than the equations generated by the finite element method alone in acquiring the same accuracy. Therefore, the coupling of boundary and finite element methods is more efficient than the finite element method alone in solving a complex Soil Structure Interaction problem.

## 7.3 Recommendations

The coupling of boundary and finite methods is the proposed approach to analyze a U-Frame structure. In this study, boundary elements were applied to simulate the behavior of surrounding soil and internal cells were used to calculate body force terms by a domain integral. The process in computing body force terms adopted is the traditional boundary element method which requires soil mass to be divided into integration cells since the domain integral must be evaluated numerically. This process greatly increases the amount of required data preparation and causes the boundary element method to lose much of its advantage over the finite element method.

Danson (31) has presented a numerical technique recently to improve the efficiency in evaluating body force terms. According to his method, the body force terms are expressed by a simple function of gravity inertia such that the domain integral can be transformed to a boundary integral which may be evaluated at the same time as the other boundary integrals.

The present study concentrates on the linear isotropic stress analysis of a soil-structure system. However, in many practical applications, the soil-structure system is non-homogeneous and/or non-linear. Brebbia (32) has proposed a solution to the system with non-homogeneous materials. Therefore, any future work should place emphasis upon the boundary integral formulation of inelastic problems.

# A SELECTED BIBLIOGRAPHY

(1) ICES STRUDL-II, Engineering User's Manual, Vol. 2, Cranston: Users Group INC., 1979.

(2) Anderson, W. J., MSC/NASTRAN, Inrteractive Training Program. New York: John Wiley & Sons, 1983.

(3) Academic American Encyclopedia, Dan Bury, Conneticut: Grolier Incoporated, 1983, pp. 94-95.

(4) Brebbia, C. A., J. C. F. Telles, and L. C. Wrobel. Boundary Element Techniques, New York: Springer-Verlag, 1984, pp. 400-407.

(5) Osias, J. R., R. B. Wilson, and L. A. Seitelman. "Combined Boundary Integral Equation Finite Element Analysis of Solids." Proceedings, 1st Symposium on Innovative Numerical Analysis in Applied Engineering Science, Versailles, France, 1977, pp. 23-27.

(6) Zienkiewicz, O. C., D. W. Kelly, and P. Bettes. "The Coupling of the Finite Element Method and Boundary Solution Procedures." International Journal for Numerical Methods in Engineering, Vol. 11 (1977), pp. 355-375.

(7) Lachat, J. C. and J. O. Watson. "Effective Numerical Treatment of Boundary Integral Equations: A Formulation for Three-Dimensional Elastostatics." International Journal for Numerical Methods in Engineering, Vol. 10 (1976), pp. 991-1005.

(8) Banerjee, P. K. "Integral Equation Methods for Analysis of Piecewise Non-Homogeneous Three-Dimensional Elastic Solids of Arbitrary Shape." International Journal of Mechanical Science, Vol. 18 (1976), pp. 293-303.

(9) Sokolnikoff, I. S. Mathematical Theory of Elasticity. 2nd Ed., New York: McGraw-Hill, 1956, pp. 336-339.

(10) Crouch, S. L. and A. M. Starfield. Boundary Element Methods in Solid Mechanics. Boston: Allen & Unwin, 1983, pp. 110-111.

(11) Danson, D., "Linear Isotropic Elasticity with Body Forces", Progress in Boundary Element Methods, Vol. 2, Ed. C. A. Brebbia, London: Pentech Press, 1983, pp. 102-115.

(12) Brebbia, C. A. The Boundary Element Method for Engineers. 2nd Ed., London: Pentech Press, 1980, pp. 142.

(13) Brebbia, C. A., J. C. F. Telles, and L. C. Wrobel. Boundary Element Techniques, New York: Springer-Verlag, 1984, pp. 190.

(14) Brebbia, C. A. The Boundary Element Method for Engineers. 2nd Ed., London: Pentech Press, 1980, pp. 130.

(15) Brebbia, C. A. The Boundary Element Method for Engineers. 2nd Ed., London: Pentech Press, 1980, pp. 142.

(16) Brebbia, C. A. and S. Walker. Boundary Element Techniques in Engineering. London and Boston: Newnes-Butterworths, 1980, pp. 133.

(17) Chaudonneret, M. "On the Discontinuity of the Stress Vector in the Boundary Integral Equation Method for Elastic Analysis." Recent Advances in Boundary Element Methods, Ed. C. A. Brebbia, London: Pentech Press, 1978, pp. 185-194.

(18) Brebbia, C. A. The Boundary Element Method for Engineers. 2nd Ed., London: Pentech Press, 1980, pp. 145-156.

(19) Brebbia, C. A., J. C. F. Telles, and L. C. Wrobel. Boundary Element Techniques, New York: Springer-Verlag, 1984, pp. 204-206.

(20) Ghaboussi, J., E. L. Wilson, and J. Isenberg. "Finite Element for Rock Joints and Interfaces." Journal of the Soil Mechanics and Foundation Divisions, ASCE, Vol. 99, no. SM 10 (1973), pp. 833-847.

(21) Goodman, R. E., R. L. Taylor, and T. L. Brekke. "A Model for the Mechanics of Jointed Rock." Journal of the Soil Mechanics and Foundation Divisions, ASCE, Vol. 94, no. SM 3 (1968), pp. 637-659.

(22) Zienkiewicz, O. C., et al. "Analysis of Nonlinear Problems in Rock Mechanics with Particular Reference to Jointed Rock Systems." Proceedings of the 2nd Congress of the International Society for Rock Mechanics, Belgrade, Yugoslavia, 1970.

(23) Zienkiewicz, O. C., S. Valliappan, and I. P. King. "Stress Analysis of Rock as a "No Tension" Material." Geotechnique, Vol. 18, no. 1 (1968), pp. 56-66.

(24) Desai, C. S. and J. F. Abel. Introduction to the Finite Element Method. New York: Van Nostrand Reinhold Company, 1972, pp. 10-15.

(25) Desai, C. S. and J. T. Christian. Numerical Methods in Geotechnical Engineering. New York: McGraw-Hill, 1977, pp. 18-31.

(26) Zienkiewicz, O. C. The Finite Element Method in Engineering Science. New York: McGraw-Hill, 1971, pp. 103-129.

(27) Goodman, R. E. Methods of Geological Engineering. New York: West Publishing Company, 1976, pp. 320-328.

(28) Brebbia, C. A. and S. Walker. Boundary Element Techniques in Engineering. London and Boston: Newnes-Butterworths, 1980, pp. 202-207.

(29) Brebbia, C. A., J. C. F. Telles, and L. C. Wrobel. Boundary Element Techniques, New York: Springer-Verlag, 1984, pp. 401-407.

(30) Brebbia, C. A., J. C. F. Telles, and L. C. Wrobel. Boundary Element Techniques, New York: Springer-Verlag, 1984, pp. 435-436.

(31) Danson, D. J. "A Boundary Element Formulation of Problems in Linear Isotropic Elasticity with Body Forces." Boundary Element Methods. Proceedings of the Third International Seminar. Ed. C. A. Brebbia. CML Publications: Spring-Verlag New York, 1981, pp. 105-122.

(32) Brebbia, C. A. The Boundary Element Method for Engineers. 2nd Ed., London: Pentech Press, 1980, pp. 169-174.

# APPENDIX A

# COMPUTER CODINGS AND INPUT LISTINGS OF PROGRAM BOUFIN

```
C
C
C ==================================================================== C
C                                                                      C
C       THIS PROGRAM IS APPLIED TO EVALUATE THE ELASTIC BEHAVIOR       C
C               OF A U-FRAME STRUCTURE BY THE COUPLING OF              C
C                  BOUNDARY AND FINITE ELEMENT METHODS                 C
C                                                                      C
C ==================================================================== C
C
C     X, Y. NODE COORDINATES
C     E. ELASTICITY MATRIX FOR STRUCTURE
C     ICONNE  ELEMENT INCIDENCE VECTOR
C     A  SYSTEM MATRIX
C     IDCOLU  SPECIFIED BOUNDARY CONDITIONS, 0 -` LOAD, 1 -> DISP
C     COLUMN  SPECIFIED BOUNDARY VALUE
C     BODY  BODY FORCE VECTOR
C     STR. STRESS VECTOR
C     DISP  DISPLACEMENT VECTOR
C     ELSTIF  ELEMENT STIFFNESS MATRIX
C     NNSTR2. 2 X NUMBER OF NODES IN STRUCTURE
C     NNINT4  4 X NUMBER OF NODES IN INTERFACE
C     NNSOL2. 2 X NUMBER OF NODES IN SOIL
C     NESTR  NUMBER OF STRUCTURE ELEMENTS
C     NEJON. NUMBER OF JOINT ELEMENTS
C     NEINT. NUMBER OF INTERFACE ELEMENTS
C     NESOL  NUMBER OF SOIL ELEMENTS
C     NEREG. NUMBER OF INTEGRATION CELLS
C     HH, GG. MATRIX GENERATED IN BOUNDARY ELEMENT REGION
C     XM, XM1  VECTOR IN STORING SOLUTIONS
C
C----------------------------
C
C          MAIN PROGRAM
C
C----------------------------
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(250), Y(250), E(3,3), ICONNE(250,4), IDCOLU(420)
      DIMENSION COLUMN(420), A(420,420), BODY(420), XM1(420)
      DIMENSION STR(3,1), DISP(2,1), HH(200,200), GG(200,200), XM(420)
      DIMENSION ELSTIF(8,8)
C
      COMMON /CO/ X, Y
      COMMON /NNN/NNSTR2,NNINT4,NNSOL2,NESTR,NEJON,NEINT, NESOL, NEREG
      COMMON /MMM/ NEB, NEL, NEREG1, NEREG2
      COMMON /IO/IRE,IWR
C
C----      SET POINTERS AND INPUT TITLE
C
           IRE = 5
           IWR = 6
      WRITE (IWR,9)
C
C----      CALL SUBROUTINE "INPUT"
C
           IER = 0
      CALL INPUT(ICONNE,IDCOLU,COLUMN,IUNKNO,ITYPE,IER)
      IF (IER  EQ  0) GO TO 1000
           GO TO 10000
C
 1000      DO 5 M = 1, IUNKNO
                BODY(M) = 0
                XM1(M) = 0
                DO 8 N = 1, IUNKNO
                     A(M,N) = 0
    8           CONTINUE
    5      CONTINUE
C
C----      FINITE ELEMENT FORMULATION
C
      CALL FINITE(E,ICONNE,A,BODY)
C
C----      BOUNDARY CONDITION IN THE STRUCTURE AND INTERFACE
C
           IK = NNSTR2 + NNINT4 / 2
           DO 10 J = 1, IK
           IF (IDCOLU(J)  EQ  0) GO TO 20
           CALL CLEAN(J,COLUMN(J),A,BODY)
                BODY(J) = COLUMN(J)
                GO TO 10
   20           BODY(J) = BODY(J) + COLUMN(J)
   10      CONTINUE
C
C----      MATRIX "M" FORMULATION
C
           JB = NESTR + NEJON + 1
           JL = NESTR + NEJON + NEINT
C
           DO 30 JJ = JB, JL
                IB = ICONNE(JJ,1)
                IL = ICONNE(JJ,2)
                XLENG = DSQRT((X(IL)-X(IB)) ** 2 + (Y(IL)-Y(IB)) ** 2)
                KK = 2 * IB - 1
                LL = 2 * IL - 1
                MM = KK + NNINT4 / 2
                NN = LL + NNINT4 / 2
C
                A(KK,MM) = A(KK,MM) + XLENG / 3.
                A(LL,MM) = A(LL,MM) + XLENG / 6
                A(KK+1,MM+1) = A(KK+1,MM+1) + XLENG / 3.
                A(LL+1,MM+1) = A(LL+1,MM+1) + XLENG / 6
                A(KK,NN) = A(KK,NN) + XLENG / 6
                A(LL,NN) = A(LL,NN) + XLENG / 3.
                A(KK+1,NN+1) = A(KK+1,NN+1) + XLENG / 6
                A(LL+1,NN+1) = A(LL+1,NN+1) + XLENG / 3
   30      CONTINUE
C
C----      BOUNDARY ELEMENT FORMULATION
C----      ITYPE = 1 UNSYMMETRY
C----      ITYPE = 2 SYMMETRY ABOUT Y AXIS
C
      CALL BOUND(ICONNE,HH,GG,BODY,ITYPE,IFA,NIF,IDCOLU,COLUMN,CC)
C
           IROW = NNSTR2 + NNINT4 / 2
           ICOL1 = NNSTR2
           ICOL2 = IROW
           MROW = NNSOL2 + NNINT4 / 2
           MCOL = NNINT4 / 2
C
C----      ASSEMBLE  MATRIX "HH" AND "GG" INTO MATRIX "A"
C
           DO 60 M = 1, MROW
                DO 70 N = 1, MCOL
                     A(IROW+M,ICOL1+N) = HH(M,N)
                     A(IROW+M,ICOL2+N) = - GG(M,N)
   70           CONTINUE
   60      CONTINUE
C
           MCOL = MCOL + 1
           DO 80 M = 1, MROW
                DO 110 N = MCOL, MROW
```

```
                     A(IROW+M,ICOL2+N) = HH(M,N)
  110               CONTINUE
   80          CONTINUE
C
C----          SOME TERMS IN MATRIX A ARE MULTIPLIED BY CC OR CD TO AVOID
C----          NUMERICAL ERROR
C
               MM = NNSTR2 + 1
               NN = MM + NNINT4 / 2
               LL = IUNKNO - NNSOL2
               CD = A(1,1) / A(NN,MM)
               CC = CC * CD
               DO 40 J = MM, IUNKNO
                    DO 50 K = 1, IUNKNO
                    IF ((J  LT  NN)  OR  (J  GT  LL)) GO TO 55
                         A(K,J) = CC * A(K,J)
                         GO TO 50
   55                    A(K,J) = CD * A(K,J)
   50               CONTINUE
   40          CONTINUE
C
               IT = 0
               MB = NESTR + 1
               ML = NESTR + NEJON
C
C----          CALAULATE THE INVERSE OF GLOBAL MATRIX
C
      CALL INVER(A,420,IUNKNO)
C
C----          XM  STORE SOLUTION OF UNKNOWNS
C
 2000 CALL PROD1(A,BODY,XM,420,IUNKNO,IDCOLU,CC,CD)
C
C----          CHECK THE CONVERGENCE OF DISPLACEMENTS AT INTERFACE
C
      IF (IT  EQ  11) GO TO 190
               IB = NNSTR2 + 1
               IL = NNSTR2 + NNINT4 / 2
C
               DO 120 K = IB, IL
               IF (DABS(XM(K) - XM1(K))  GT  0 0000001) GO TO 130
  120          CONTINUE
C
               GO TO 190
C
C----          RESTART THE PROLEM  STORE "XM" TO "XM1"
C
  130          DO 140 I = 1, IUNKNO
                         XM1(I) = XM(I)
  140          CONTINUE
C
C----          ITERATIVE ROUTINE
C
               IT = IT + 1
C
               CALL ITER(XM,BODY,ICONNE,MB,ML)
C
               GO TO 2000
C
C----          REORDER AND STORE DISPLACEMENTS TO XM, TRACTIONS TO COLUMN
C
  190          M = NNSTR2 + NNINT4 / 2 + 1
               N = NNSTR2 + NNINT4
               DO 530 K = M, N
                    L = K - NNINT4 / 2
                    COLUMN(L) = XM(K)
```

```
  530          CONTINUE
C
               M = N + 1
C
               DO 550 K = M, IUNKNO
                    L = K - NNINT4 / 2
               IF (IDCOLU(K)  EQ  0) GO TO 540
                    COLUMN(L) = XM(K)
                    XM(L) = COLUMN(K)
                    GO TO 550
  540               COLUMN(L) = COLUMN(K)
                    XM(L) = XM(K)
  550          CONTINUE
C
C----          STRUCTURE OUTPUT
C
               NNSTR = NNSTR2 / 2
      WRITE (IWR,90)
               DO 570 L = 1, NNSTR
               WRITE (IWR,94) L, XM(2*L-1), XM(2*L)
  570          CONTINUE
      WRITE (IWR,97)
               DO 580 JJ = 1, NESTR
                    I = ICONNE(JJ,1)
                    J = ICONNE(JJ,2)
                    K = ICONNE(JJ,3)
                    L = ICONNE(JJ,4)
               CALL SOLVE(I,J,K,L,E,STR,DISP,XM)
               WRITE (IWR,904) JJ,(DISP(M,1),M=1,2),(STR(M,1),M=1,3)
  580          CONTINUE
      WRITE (IWR,99)
               DO 585 JJ = MB, ML
                    I = ICONNE(JJ,1)
                    J = ICONNE(JJ,2)
                    K = ICONNE(JJ,3)
                    L = ICONNE(JJ,4)
               CALL FIND(I,J,K,L,STR,DISP,XM)
               WRITE (IWR,906) JJ,I,J,(DISP(M,1),M=1,2),(STR(M,1),M=1,2)
  585          CONTINUE
C
C----          INTERFACE OUTPUT
C
               WRITE (IWR,907)
               M = NNSTR + 1
               N = NNSTR + NNINT4 / 4
C
               DO 590 K = M, N
               WRITE (IWR,914) K,XM(2*K-1),XM(2*K),COLUMN(2*K-1),COLUMN(2*K)
  590          CONTINUE
C
C----          SOIL OUTPUT
C
      WRITE (IWR,917)
               M = N + 1
               N = M + NNSOL2 / 2 - 1
C
               DO 600 K = M, N
               WRITE (IWR,914) K,XM(2*K-1),XM(2*K),COLUMN(2*K-1),COLUMN(2*K)
  600          CONTINUE
C
      WRITE (IWR,97)
C
      IF (NEREG  EQ  0) GO TO 10000
               DO 200 JJ = NEREG1, NEREG2
                    I = ICONNE(JJ,1)
                    J = ICONNE(JJ,2)
```

```
                K = ICONNE(JJ,3)
                L = ICONNE(JJ,4)
                XI = X(I)
                XJ = X(J)
                XK = X(K)
                XL = X(L)
                YI = Y(I)
                YJ = Y(J)
                YK = Y(K)
                YL = Y(L)
                XS =  25 * (XI + XJ + XK + XL)
                YS =  25 * (YI + YJ + YK + YL)
C
                DO 230 ISY = 1, IFA, NIF
                IF (ISY  EQ  1) GO TO 235
                     XS = - XS
                     GO TO 240
 235                 DISP(1,1) = 0
                     DISP(2,1) = 0
                     STR(1,1) = 0
                     STR(2,1) = 0
                     STR(3,1) = 0
  240           CALL CALCU(ICONNE,XS,YS,COLUMN,XM,STR,DISP,ISY)
  230           CONTINUE
           WRITE(IWR,904)JJ,(DISP(M,1),M=1,2),STR(1,1),STR(3,1),STR(2,1)
  200      CONTINUE
C
      WRITE(IWR,924)
C
    9 FORMAT(1H1,///,40X,'COUPLING  OF  FINITE  AND  BOUNDARY  ELEMENT'
     &              '  METHODS',
     &            //,40X,'                         APPLIED  TO',
     &            //,42X,'     SOIL-STRUCTURE  INTERACTION  PROBLEMS')
C
   90 FORMAT(/////,36X,'**********************',
     &           /,36X,'*  STRUCTURE OUTPUT  *',
     &           /,36X,'**********************',
     &     ////,41X,'NODAL DISPLACEMENT ',
     &  //,34X,'NODE',7X,'U',13X,'V')
C
   94 FORMAT(34X,I3,2(3X,E11 4))
C
   97 FORMAT(////,51X,'ELEMENT DISPLACEMENT AND STRESS',//,
     &  34X,'ELEMENT',4X,'U',13X,'V',12X,'SXX',11X,'SYY',10X,'SXY')
C
   99 FORMAT(////,47X,'RELATIVE DISPLACEMENT AND LOCAL STRESS ',
     &       'AT JOINT ELEMENT',//,34X,'ELEMENT',5X,'LOCAL X AXIS',
     &       7X,'TAN  DISP',5X,'NOR  DISP',8X,'TAU',10X,'SIGMA')
C
  904 FORMAT(34X,I3,5(3X,E11 4))
C
  906 FORMAT(34X,I3,6X,'NODE',I3,' TO NODE',I3,4(3X,E11 4))
C
  907 FORMAT(/////,36X,'**********************',
     &           /,36X,'*  INTERFACE OUTPUT  *',
     &           /,36X,'**********************',
     &     ////,43X,'DISPLACEMENT AND TRACTION AT INTERFACE',
     &  //,34X,'NODE',7X,'U',13X,'V',12X,'TXX',11X,'TYY')
C
  914 FORMAT(34X,I3,4(3X,E11 4))
C
  917 FORMAT(/////,36X,'******************',
     &           /,36X,'*  SOIL  OUTPUT  *',
     &           /,36X,'******************',
     &     ////,43X,'DISPLACEMENT AND TRACTION AT BOUNDARY NODE',
     &  //,34X,'NODE',7X,'U',13X,'V',12X,'TXX',11X,'TYY')
```

```
C
  924 FORMAT(1H1)
C
10000 STOP
      END
C----------------------------
C
C          SUBPROGRAM "INVER"
C
C----------------------------
C
      SUBROUTINE INVER(A,NX,N)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(NX,NX)
C
           DO 100 K = 1, N
C
                DO 20 J = 1, N
                IF (J  EQ  K) GO TO 20
                     A(K,J) = A(K,J) / A(K,K)
   20           CONTINUE
C
                A(K,K) = 1  / A(K,K)
C
                DO 30 I = 1, N
                IF(I  EQ  K) GO TO 30
C
                     DO 40 J = 1, N
                     IF (J .EQ  K) GO TO 40
                          A(I,J) = A(I,J) - A(K,J) * A(I,K)
   40                CONTINUE
   30           CONTINUE
C
                DO 50 I = 1, N
                IF (I  EQ  K) GO TO 50
                     A(I,K) = -A(I,K) * A(K,K)
   50           CONTINUE
  100      CONTINUE
C
      RETURN
      END
C----------------------------
C
C          SUBPROGRAM "ITER"
C
C----------------------------
C
      SUBROUTINE ITER(XM,BODY,ICONNE,MB,ML)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION XM(420), BODY(420), ELSTIF(8,8), X(250)
      DIMENSION Y(250), ICONNE(250,4)
C
      COMMON /CO/ X,Y
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
C
           DO 10 JJ = MB, ML
                I = ICONNE(JJ,1)
                J = ICONNE(JJ,2)
                K = ICONNE(JJ,3)
                L = ICONNE(JJ,4)
                XLENG = DSQRT((X(J) - X(I)) ** 2 + (Y(J) - Y(I)) ** 2)
                COST = (X(J) - X(I)) / XLENG
                SINT = (Y(J) - Y(I)) / XLENG
C
```

```
C----     RELATIVE DISPLACEMENT AT CENTROID OF JOINT ELEMENT
C
                U= O 5 * (XM(2*K-1) + XM(2*L-1) - XM(2*I-1) - XM(2*J-1))
                V = O 5 * (XM(2*K) + XM(2*L) - XM(2*I) - XM(2*J))
C
C----     TRANSFORM TO LOCAL DISPLACEMENT
C
                WS = U * COST + V * SINT
                WN = - U * SINT + V * COST
C
                TAU = WS * EKS
                SIGMA = WN * EKN
           IF (SIGMA  LE  O) GO TO 10
C
C----     RESTRAINING FORCE IS - SIGMA * LENGTH AND - TAU * LENGTH
C
                TX = (COST * TAU - SIGMA * SINT) * XLENG / 2
                TY = (SINT * TAU + SIGMA * COST) * XLENG / 2
                BODY(2*I-1) = BODY(2*I-1) + TX
                BODY(2*I) = BODY(2*I) + TY
                BODY(2*J-1) = BODY(2*J-1) + TX
                BODY(2*J) = BODY(2*J) + TY
C
                BODY(2*K-1) = BODY(2*K-1) - TX
                BODY(2*K) = BODY(2*K) - TY
                BODY(2*L-1) = BODY(2*L-1) - TX
                BODY(2*L) = BODY(2*L) - TY
C
   10      CONTINUE
C
      RETURN
      END
C----------------------------
C
C          SUBPROGRAM "FIND"
C
C----------------------------
C
      SUBROUTINE FIND(I,J,K,L,STR,DISP,XM)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION STR(3,1), DISP(2,1), XM(420), D(2), X(250), Y(250)
C
      COMMON /CO/ X,Y
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
C
           XD = X(J) - X(I)
           YD = Y(J) - Y(I)
           D(1) = (-XM(2*I-1) - XM(2*J-1) + XM(2*K-1) + XM(2*L-1)) * O 5
           D(2) = (- XM(2*I) - XM(2*J) + XM(2*K) + XM(2*L)) * O 5
C
           XLENG = DSQRT(XD * XD + YD * YD)
           COST = XD / XLENG
           SINT = YD / XLENG
C
C----      TRANSFORM DISPLACEMENT TO LOCAL COORDINATES AND FIND STRESS
C
           DISP(1,1) = COST * D(1) + SINT * D(2)
           DISP(2,1) = -SINT * D(1) + COST * D(2)
           STR(1,1) = EKS * DISP(1,1)
           STR(2,1) = EKN * DISP(2,1)
 1000 RETURN
      END
C----------------------------
C
C          SUBPROGRAM "FINITE"
```

```
C
C----------------------------
C
      SUBROUTINE FINITE(E,ICONNE,A,BODY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ELSTIF(8,8), A(420,420)
      DIMENSION ICONNE(250,4), BODY(420), E(3,3), DETJAC(4)
C
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
      COMMON /NNN/NNSTR2,NNINT4,NNSOL2,NESTR,NEJON,NEINT,NESOL,NEREG
C
           CONST = ESTR / ((1. + PSTR) * (1  - 2  * PSTR))
           E(1,1) = CONST * (1  - PSTR)
           E(2,2) = E(1,1)
           E(3,3) = CONST *  5 * (1  - 2  * PSTR)
           E(1,2) = CONST * PSTR
           E(2,1) = E(1,2)
           E(2,3) = O
           E(3,2) = E(2,3)
           E(1,3) = O
           E(3,1) = O
C
           N = NESTR + NEJON
           DO 100 JJ = 1, N
                I = ICONNE(JJ,1)
                J = ICONNE(JJ,2)
                K = ICONNE(JJ,3)
                L = ICONNE(JJ,4)
           IF (JJ  GT  NESTR) GO TO 50
C
           CALL STIFF(I,J,K,L,E,ELSTIF,DETJAC)
C
                BODY(2*I) = BODY(2*I) - WSTR * DETJAC(1)
                BODY(2*J) = BODY(2*J) - WSTR * DETJAC(2)
                BODY(2*K) = BODY(2*K) - WSTR * DETJAC(3)
                BODY(2*L) = BODY(2*L) - WSTR * DETJAC(4)
                GO TO 70
C
   50      CALL SJOINT(I,J,K,L,EKS,EKN,ELSTIF)
C
   70      CALL ASSEM(ELSTIF,I,J,K,L,A)
C
  100      CONTINUE
      RETURN
      END
C----------------------------
C
C          SUBPROGRAM "SJOINT"
C
C----------------------------
C
      SUBROUTINE SJOINT(I,J,K,L,ES,EN,ELSTIF)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION SQ4(8,8), ELSTIF(8,8), TRAN(8,8), TEMP(8,8), X(250)
      DIMENSION Y(250)
C
      COMMON /CO/ X, Y
C
C----      CLEAR TRANSFORMATION MATRIX AND LOCAL JOINT STIFFNESS MATRIX
C
           DO 10 M = 1, 8
                DO 20 N = 1, 8
                     TRAN(M,N) = O
                     SQ4(M,N) = O
```

```
   20            CONTINUE
   10       CONTINUE
C
            XD = X(J) - X(I)
            YD = Y(J) - Y(I)
            XLENG = DSQRT(XD * XD + YD * YD)
C
C----       FORMULATION OF TRANSPOSE "TRAN"
C
            TRAN(1,1) = XD / XLENG
            TRAN(2,2) = XD / XLENG
            TRAN(3,3) = XD / XLENG
            TRAN(4,4) = XD / XLENG
            TRAN(5,5) = XD / XLENG
            TRAN(6,6) = XD / XLENG
            TRAN(7,7) = XD / XLENG
            TRAN(8,8) = XD / XLENG
            TRAN(1,2) = - YD / XLENG
            TRAN(2,1) = YD / XLENG
            TRAN(3,4) = - YD / XLENG
            TRAN(4,3) = YD / XLENG
            TRAN(5,6) = - YD / XLENG
            TRAN(6,5) = YD / XLENG
            TRAN(7,8) = - YD / XLENG
            TRAN(8,7) = YD / XLENG
C
C----       FORMULATION OF LOACL JOINT ELEMENT STIFFNESS
C
            SQ4(1,1) = XLENG / 3  * ES
            SQ4(1,3) = XLENG / 6  * ES
            SQ4(1,5) = - XLENG / 6  * ES
            SQ4(1,7) = - XLENG / 3  * ES
C
            SQ4(3,1) = XLENG / 6  * ES
            SQ4(3,3) = XLENG / 3  * ES
            SQ4(3,5) = - XLENG / 3  * ES
            SQ4(3,7) = - XLENG / 6  * ES
C
            SQ4(5,1) = - XLENG / 6  * ES
            SQ4(5,3) = - XLENG / 3  * ES
            SQ4(5,5) = XLENG / 3  * ES
            SQ4(5,7) = XLENG / 6  * ES
C
            SQ4(7,1) = - XLENG / 3  * ES
            SQ4(7,3) = - XLENG / 6  * ES
            SQ4(7,5) = XLENG / 6  * ES
            SQ4(7,7) = XLENG / 3  * ES
C
            SQ4(2,2) = XLENG / 3  * EN
            SQ4(2,4) = XLENG / 6  * EN
            SQ4(2,6) = - XLENG / 6  * EN
            SQ4(2,8) = - XLENG / 3  * EN
C
            SQ4(4,2) = XLENG / 6  * EN
            SQ4(4,4) = XLENG / 3  * EN
            SQ4(4,6) = - XLENG / 3  * EN
            SQ4(4,8) = - XLENG / 6  * EN
C
            SQ4(6,2) = - XLENG / 6  * EN
            SQ4(6,4) = - XLENG / 3  * EN
            SQ4(6,6) = XLENG / 3  * EN
            SQ4(6,8) = XLENG / 6  * EN
C
            SQ4(8,2) = - XLENG / 3  * EN
            SQ4(8,4) = - XLENG / 6  * EN
            SQ4(8,6) = XLENG / 6  * EN
```

```
            SQ4(8,8) = XLENG / 3  * EN
C
C----       TRANSFORMATION TO GLOBAL JOINT STIFFNESS MATRIX
C
      CALL PROD(TRAN,SQ4,TEMP,8,8,8)
C
            TRAN(1,2) = - TRAN(1,2)
            TRAN(2,1) = - TRAN(2,1)
            TRAN(3,4) = - TRAN(3,4)
            TRAN(4,3) = - TRAN(4,3)
            TRAN(5,6) = - TRAN(5,6)
            TRAN(6,5) = - TRAN(6,5)
            TRAN(7,8) = - TRAN(7,8)
            TRAN(8,7) = - TRAN(8,7)
C
      CALL PROD(TEMP,TRAN,ELSTIF,8,8,8)
C
      RETURN
      END
C-----------------------------
C
C           SUBPROGRAM "STIFF"
C
C-----------------------------
C
.     SUBROUTINE STIFF(I,J,K,L,E,ELSTIF,DETJAC)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION E(3,3), SQ4(8,8), B(3,8), DUMM(8,3), DETJAC(4)
      DIMENSION XJINV(2,2), XM1(2,8), XM2(2,8), ELSTIF(8,8)
      DIMENSION AALPHA(4), ABETA(4), BT(8,3), X(250), Y(250)
C
      COMMON /CO/ X, Y
      COMMON /ALPH/ AALPHA, ABETA
C
C----       IGNORE WEIGHT, BECAUSE WEIGHT = 1 FOR 4 POINT INTEGRATION
C
            XI = X(I)
            XJ = X(J)
            XK = X(K)
            XL = X(L)
C
            YI = Y(I)
            YJ = Y(J)
            YK = Y(K)
            YL = Y(L)
C
            DO 10 IR = 1, 8
                 DO 20 IC = 1, 8
                      ELSTIF(IR,IC) = O
   20            CONTINUE
   10       CONTINUE
C
            IPOINT = 1
   15       ALPHA = AALPHA(IPOINT)
            BETA  = ABETA(IPOINT)
C
            DXDA = -  25 * (1 - BETA) * XI +  25 * (1 - BETA) * XJ
     &             +  25 * (1 + BETA) * XK -  25 * (1 + BETA) * XL
C
            DYDA = -  25 * (1 - BETA) * YI +  25 * (1 - BETA) * YJ
     &             +  25 * (1 + BETA) * YK -  25 * (1 + BETA) * YL
C
            DXDB = -  25 * (1 - ALPHA) * XI -  25 * (1 + ALPHA) * XJ
     &             +  25 * (1 + ALPHA) * XK +  25 * (1 - ALPHA) * XL
C
```

```
        DYDB = -  25 * (1 - ALPHA) * YI -  25 * (1 + ALPHA) * YJ
     &         +  25 * (1 + ALPHA) * YK +  25 * (1 - ALPHA) * YL
C
C----   DXDA, DYDA, DXDB AND DYDB ARE TERMS IN JACOBIAN
C
        XJDET = DXDA * DYDB - DXDB * DYDA
        XJINV(1,1) = DYDB / XJDET
        XJINV(1,2) = - DYDA / XJDET
        XJINV(2,1) = - DXDB / XJDET
        XJINV(2,2) = DXDA / XJDET
        DETJAC(IPOINT) = DABS(XJDET)
C
        DO 30 M = 1, 4
            XM1(1,2*M) = 0
            XM1(2,2*M) = 0
            XM2(1,2*M-1) = 0
            XM2(2,2*M-1) = 0
   30   CONTINUE
C
        XM1(1,1) = -  25 * (1 - BETA)
        XM2(1,2) = XM1(1,1)
        XM1(1,3) = +  25 * (1 - BETA)
        XM2(1,4) = XM1(1,3)
        XM1(1,5) = +  25 * (1 + BETA)
        XM2(1,6) = XM1(1,5)
        XM1(1,7) = -  25 * (1 + BETA)
        XM2(1,8) = XM1(1,7)
C
        XM1(2,1) = -  25 * (1 - ALPHA)
        XM2(2,2) = XM1(2,1)
        XM1(2,3) = -  25 * (1 + ALPHA)
        XM2(2,4) = XM1(2,3)
        XM1(2,5) = +  25 * (1 + ALPHA)
        XM2(2,6) = XM1(2,5)
        XM1(2,7) = +  25 * (1 - ALPHA)
        XM2(2,8) = XM1(2,7)
C
        DO  40 M = 1, 8
            B(1,M) = 0
            B(2,M) = 0
            B(3,M) = 0
C
            DO 50 N = 1, 2
                B(1,M) = B(1,M) + XJINV(1,N) * XM1(N,M)
                B(2,M) = B(2,M) + XJINV(2,N) * XM2(N,M)
                B(3,M) = B(3,M) + XJINV(2,N) * XM1(N,M)
     &                          + XJINV(1,N) * XM2(N,M)
   50       CONTINUE
C
            BT(M,1) = B(1,M)
            BT(M,2) = B(2,M)
            BT(M,3) = B(3,M)
   40   CONTINUE
C
    CALL PROD(BT,E,DUMM,8,3,3)
C
    CALL PROD(DUMM,B,SQ4,8,3,8)
C
        DO 60 M = 1, 8
            DO 70 N = 1, 8
                ELSTIF(M,N) = ELSTIF(M,N)+SQ4(M,N) * DETJAC(IPOINT)
   70       CONTINUE
   60   CONTINUE
C
    IF (IPOINT  EQ  4 ) GO TO 1000
C
```

```
        IPOINT = IPOINT + 1
        GO TO 15
C
 1000 RETURN
      END
C----------------------------
C
C       SUBPROGRAM "ASSEM"
C
C----------------------------
C
      SUBROUTINE ASSEM(ELSTIF,I,J,K,L,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ELSTIF(8,8), JR(4), NR(8), A(420,420)
C
          JR(1) = I
          JR(2) = J
          JR(3) = K
          JR(4) = L
C
          DO 10 M = 1, 4
              IX = 2 * M - 1
              IY = IX + 1
              NR(IX) = 2 * JR(M) -1
              NR(IY) = 2 * JR(M)
   10     CONTINUE
C
          DO 20 M = 1, 8
              DO 30 N = 1,8
                  A(NR(M),NR(N)) = A(NR(M),NR(N)) + ELSTIF(M,N)
   30         CONTINUE
   20     CONTINUE
C
      RETURN
      END
C----------------------------
C
C       SUBPROGRAM "SOLVE"
C
C----------------------------
C
      SUBROUTINE SOLVE(I,J,K,L,E,STR,DISP,BODY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION E(3,3), STR(3,1), STRAIN(3,1), BODY(420)
      DIMENSION B(3,8), DISP(2,1), XJINV(2,2), XM1(2,8), XM2(2,8)
      DIMENSION X(250), Y(250), DD(8,1)
C
      COMMON /CO/ X, Y
C
          XI = X(I)
          XJ = X(J)
          XK = X(K)
          XL = X(L)
C
          YI = Y(I)
          YJ = Y(J)
          YK = Y(K)
          YL = Y(L)
C
          DD(1,1) = BODY(2*I-1)
          DD(2,1) = BODY(2*I)
          DD(3,1) = BODY(2*J-1)
          DD(4,1) = BODY(2*J)
          DD(5,1) = BODY(2*K-1)
```

```
      DD(6,1) = BODY(2*K)
      DD(7,1) = BODY(2*L-1)
      DD(8,1) = BODY(2*L)
C
      DISP(1,1) = 0
      DISP(2,1) = 0
C
      DO 15 K = 1, 4
          DISP(1,1) = DISP(1,1) +  25 * DD(2*K-1,1)
          DISP(2,1) = DISP(2,1) +  25 * DD(2*K,1)
 15   CONTINUE
C
      DXDA = -  25 * XI +  25 * XJ +  25 * XK -  25 * XL
C
      DYDA = -  25 * YI +  25 * YJ +  25 * YK -  25 * YL
C
      DXDB = -  25 * XI -  25 * XJ +  25 * XK +  25 * XL
C
      DYDB = -  25 * YI -  25 * YJ +  25 * YK +  25 * YL
C
C---- DXDA, DYDA, DXDB AND DYDB ARE TERMS IN JACOBIAN
C
      XJDET = DXDA * DYDB - DXDB * DYDA
      XJINV(1,1) = DYDB / XJDET
      XJINV(1,2) = - DYDA / XJDET
      XJINV(2,1) = - DXDB / XJDET
      XJINV(2,2) = DXDA / XJDET
C
      DO 30 M = 1, 4
          XM1(1,2*M) = 0
          XM1(2,2*M) = 0
          XM2(1,2*M-1) = 0
          XM2(2,2*M-1) = 0
 30   CONTINUE
C
      XM1(1,1) = - .25
      XM2(1,2) = XM1(1,1)
      XM1(1,3) = +  25
      XM2(1,4) = XM1(1,3)
      XM1(1,5) = +  25
      XM2(1,6) = XM1(1,5)
      XM1(1,7) = -  25
      XM2(1,8) = XM1(1,7)
C
      XM1(2,1) = -  25
      XM2(2,2) = XM1(2,1)
      XM1(2,3) = -  25
      XM2(2,4) = XM1(2,3)
      XM1(2,5) = +  25
      XM2(2,6) = XM1(2,5)
      XM1(2,7) = +  25
      XM2(2,8) = XM1(2,7)
C
      DO  40 M = 1, 8
          B(1,M) = 0
          B(2,M) = 0
          B(3,M) = 0
C
          DO 50 N = 1, 2
              B(1,M) = B(1,M) + XJINV(1,N) * XM1(N,M)
              B(2,M) = B(2,M) + XJINV(2,N) * XM2(N,M)
              B(3,M) = B(3,M) + XJINV(2,N) * XM1(N,M)
     &                        + XJINV(1,N) * XM2(N,M)
 50       CONTINUE
C
 40   CONTINUE
```

```
C
      CALL PROD(B,DD,STRAIN,3,8,1)
C
      CALL PROD(E,STRAIN,STR,3,3,1)
C
      RETURN
      END
C----------------------------
C
C           SUBPROGRAM "CLEAN"
C
C----------------------------
C
      SUBROUTINE CLEAN(J,V,A,BODY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(420,420),BODY(420)
C
      COMMON /NNN/NNSTR2,NNINT4,NNSOL2,NESTR,NEJON,NEINT, NESOL, NEREG
C
          IK = NNSTR2 + NNINT4 + NNSOL2
          DO 10 K = 1, IK
              BODY(K) = BODY(K) - A(K,J) * V
              A(J,K) = 0
              A(K,J) = 0.
 10       CONTINUE
C
          A(J,J) = 1
      RETURN
      END
C----------------------------
C
C           SUBPROGRAM "BOUND"
C
C----------------------------
C
      SUBROUTINE BOUND(ICONNE,HH,GG,BODY,ITYPE,IFA,NIF,IDCOLU,COLUMN,CC)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION GI(3,6), OME(3,6), D(2,2), HH(200,200), GG(200,200)
      DIMENSION H(2,4), G(2,4), USTAR(2,2), AALPHA(4), ABETA(4)
      DIMENSION ICONNE(250,4), BODY(420), X(250), Y(250), IDCOLU(420)
      DIMENSION COLUMN(420)
C
      COMMON /CO/ X, Y
      COMMON /NNN/NNSTR2,NNINT4,NNSOL2,NESTR,NEJON,NEINT, NESOL, NEREG
      COMMON /CONS/ C1, C2, C3, C4, C5, C6, C7, GI, OME, D
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
      COMMON /ALPH/ AALPHA, ABETA
      COMMON /MMM/ NEB, NEL, NEREG1, NEREG2
C
          GE = ESOL / (2  * (1  + PSOL))
          C2 = 3  - 4  * PSOL
          C3 = 1  / ((1  - PSOL) * 12 56637062)
          C4 = 1  - 2. * PSOL
          C6 = 2  * C3 * GE
          C7 = 1  - 4  * PSOL
          C1 = C3 / (2  * GE)
          C5 = C1 / 2.
          CC = 2  * GE / (1. - PSOL)
C
          D(1,1) = 1
          D(1,2) = 0
          D(2,1) = 0
          D(2,2) = 1
C
```

```
          NNB = NNSTR2 / 2 + 1
          NNL = NNB + NNSOL2 / 2 + NNINT4 / 4 - 1
          KKK = 2 * (NNL - NNB + 1)
          DO 10 J = 1, KKK
                DO 15 L = 1, KKK
                      HH(J,L) = 0
                      GG(J,L) = 0
   15           CONTINUE
   10     CONTINUE
C
          NEB = NESTR + NEJON + 1
          NEL = NEB + NEINT + NESOL - 1
C
          IFA = 1
          NIF = 1
          IF (ITYPE  EQ  1) GO TO 23
C
          IFA = 3
          NIF = 2
C
   23     DO 990 ISY = 1, IFA, NIF
                NODE = NNB
 1000           XS = X(NODE)
                YS = Y(NODE)
C
          IF (ISY  EQ  1) GO TO 55
                XS = -XS
C
   55           DO 100 J = NEB, NEL
                      IB = ICONNE(J,1)
                      IL = ICONNE(J,2)
                      DXB = DABS(X(IB)-XS)
                      DYB = DABS(Y(IB)-YS)
                      DXL = DABS(X(IL)-XS)
                      DYL = DABS(Y(IL)-YS)
                      ICOD = 1
                IF ((ISY  NE  1)  AND  (DABS(XS)  GT  0 001)) GO TO 40
C
                IF ((DXB  GT  0 001)  OR  (DYB  GT  0 001)) GO TO 25
                      ICOD = 2
                      GO TO  40
C
   25           IF((DXL  GT  0 001)  OR  (DYL  GT  0 001)) GO TO 40
                      ICOD = 3
C
   40           CALL FUNC(ICOD,H,G,IB,IL,XS,YS,USTAR,XX,YY,ISY)
C
C----      SET REQUIRED POINTERS IN TERMS OF LOCAL MATRIX "HH", "GG"
C
                      M = 2 * (NODE - NNB) + 1
                      N = 2 * (IB - NNB) + 1
                      L = 2 * (IL - NNB) + 1
C
                      HH(M,N) = HH(M,N) + H(1,1)
                      HH(M,N+1) = HH(M,N+1) + H(1,2)
                      HH(M,L) = HH(M,L) + H(1,3)
                      HH(M,L+1) = HH(M,L+1) + H(1,4)
C
                      HH(M+1,N) = HH(M+1,N) + H(2,1)
                      HH(M+1,N+1) = HH(M+1,N+1) + H(2,2)
                      HH(M+1,L) = HH(M+1,L) + H(2,3)
                      HH(M+1,L+1) = HH(M+1,L+1) + H(2,4)
C
                      GG(M,N) = GG(M,N) + G(1,1)
                      GG(M,N+1) = GG(M,N+1) + G(1,2)
                      GG(M,L) = GG(M,L) + G(1,3)
                      GG(M,L+1) = GG(M,L+1) + G(1,4)
C
                      GG(M+1,N) = GG(M+1,N) + G(2,1)
                      GG(M+1,N+1) = GG(M+1,N+1) + G(2,2)
                      GG(M+1,L) = GG(M+1,L) + G(2,3)
                      GG(M+1,L+1) = GG(M+1,L+1) + G(2,4)
C
                IF (ISY  EQ  1) GO TO 90
                      H(1,1) = - H(1,1)
                      H(2,1) = - H(2,1)
                      H(1,3) = - H(1,3)
                      H(2,3) = - H(2,3)
   90                 HH(M,M) = HH(M,M)- H(1,1)- H(1,3)
                      HH(M,M+1) = HH(M,M+1)- H(1,2)- H(1,4)
                      HH(M+1,M) = HH(M+1,M) - H(2,1) - H(2,3)
                      HH(M+1,M+1) = HH(M+1,M+1) - H(2,2) - H(2,4)
  100           CONTINUE
          IF (NEREG  EQ  0) GO TO 220
                NEREG1 = NEL + 1
                NEREG2 = NEREG1 + NEREG - 1
C
                DO 200 JJ = NEREG1, NEREG2
                      I = ICONNE(JJ,1)
                      J = ICONNE(JJ,2)
                      K = ICONNE(JJ,3)
                      L = ICONNE(JJ,4)
                      XI = X(I)
                      XJ = X(J)
                      XK = X(K)
                      XL = X(L)
                      YI = Y(I)
                      YJ = Y(J)
                      YK = Y(K)
                      YL = Y(L)
C
                      IPOINT = 1
  300                 AL = AALPHA(IPOINT)
                      BE = ABETA(IPOINT)
                      XX =  25*(1-AL)*(1-BE)*XI +  25*(1+AL)*(1-BE)*XJ
     &                   +  25*(1+AL)*(1+BE)*XK +  25*(1-AL)*(1+BE)*XL
                      YY =  25*(1-AL)*(1-BE)*YI +  25*(1+AL)*(1-BE)*YJ
     &                   +  25*(1+AL)*(1+BE)*YK + .25*(1-AL)*(1+BE)*YL
C
                      DXDA = -  25 * (1 - BE) * XI +  25 * (1 - BE) * XJ
     &                      +  25 * (1 + BE) * XK -  25 * (1 + BE) * XL
                      DYDA = -  25 * (1 - BE) * YI +  25 * (1 - BE) * YJ
     &                      +  25 * (1 + BE) * YK -  25 * (1 + BE) * YL
C
                      DXDB = -  25 * (1 - AL) * XI -  25 * (1 + AL) * XJ
     &                      +  25 * (1 + AL) * XK +  25 * (1 - AL) * XL
C
                      DYDB = -  25 * (1 - AL) * YI -  25 * (1 + AL) * YJ
     &                      + .25 * (1 + AL) * YK +  25 * (1 - AL) * YL
C
                      XJDET = DXDA * DYDB - DXDB * DYDA
                      XJA = DABS(XJDET)
                      ICOD = 4
C
                CALL FUNC(ICOD,H,G,IB,IL,XS,YS,USTAR,XX,YY,ISY)
C
C----      SET REQUIRED POINTERS IN TERMS OF GLOBAL MATRIX "A"
C
                      MM = 2 * NODE - 1 + NNINT4 / 2
C
                      BODY(MM) = BODY(MM) - WSOL * USTAR(1,2) * XJA
                      BODY(MM+1) = BODY(MM+1) - WSOL * USTAR(2,2) * XJA
```

```
                     IPOINT = IPOINT + 1
                IF (IPOINT  LE  4) GO TO 300
C
  200           CONTINUE
C
  220           NODE = NODE + 1
           IF (NODE  LE  NNL) GO TO 1000
C
  990      CONTINUE
C
C----      TAKE CARE OF GIVEN BOUNDARY CONDITION FOR "HH" AND "GG"
C
           ILOC = KKK - NNSOL2 + 1
C
           DO 350 K = ILOC, KKK
                IGLO = K + NNSTR2 + NNINT4 / 2
           IF (IDCOLU(IGLO)  EQ  O) GO TO 400
C
C----      THE TERMS IN GG ARE MULTIPLIED BY CC TO AVOID NUMERICAL ERROR
C
                DO 380 N = 1, KKK
                     IG = N + NNSTR2 + NNINT4 / 2
                     BODY(IG) = BODY(IG) - HH(N,K) * COLUMN(IGLO)
                     HH(N,K) = - GG(N,K) * CC
  380           CONTINUE
C
                GO TO 350
C
  400           DO 450 M = 1, KKK
                     IG = M + NNSTR2 + NNINT4 / 2
                     BODY(IG) = BODY(IG) + GG(M,K) * COLUMN(IGLO)
  450           CONTINUE
C
  350      CONTINUE
              RETURN
              END
C-----------------------------
C
C          SUBPROGRAM "FUNC"
C
C-----------------------------
C
      SUBROUTINE FUNC(ICOD,H,G,IB,IL,XS,YS,USTAR,XX,YY,ISY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION USTAR(2,2), G(2,4), H(2,4), GI(3,6), OME(3,6)
      DIMENSION PSTAR(2,2), DRD(2), DND(2), PHISC(2), XY(2)
      DIMENSION X(250), Y(250), D(2,2)
      COMMON /CO/ X, Y
      COMMON /CONS/ C1, C2, C3, C4, C5, C6, C7, GI, OME, D
           DO 10 KK = 1, 2
                DO 20 L = 1, 4
                     G(KK,L) = O
                     H(KK,L) = O
   20           CONTINUE
   10      CONTINUE
C
           XY(1) = X(IL) - X(IB)
           XY(2) = Y(IL) - Y(IB)
           XLENG = (XY(1) * XY(1) + XY(2) * XY(2)) ** O 5
           DND(1) = XY(2) / XLENG
           DND(2) = - XY(1) / XLENG
      IF ((ICOD  EQ  2)  OR  (ICOD  EQ  3)) GO TO 2000
C
      IF (ICOD  EQ  1) GO TO 1000
C
```

```
  500      XDD = XX - XS
           YDD = YY - YS
           GO TO 25
C
 1000      SEL = O 5 * DSQRT((2  * XS - X(IB) - X(IL)) ** 2 + (2  *
     &           YS - Y(IB) - Y(IL)) ** 2) / XLENG
      IF (SEL  LE  1 5) GO TO 115
      IF (SEL  LE  5 5) GO TO 125
           L = 1
           NPOINT = 2
           GO TO 135
C
  115      L = 3
           NPOINT = 6
           GO TO 135
C
  125      L = 2
           NPOINT = 4
  135      IPOINT = 1
  700      XDD = (X(IB)+X(IL)) / 2 - XS + XY(1) / 2  * GI(L,IPOINT)
           YDD = (Y(IB)+Y(IL)) / 2 - YS + XY(2) / 2  * GI(L,IPOINT)
   25      R = (XDD * XDD + YDD * YDD) ** O 5
           DRD(1) = XDD / R
           DRD(2) = YDD / R
           DRDN = DRD(1) * DND(1) + DRD(2) * DND(2)
C
C----      COMPUTE MATRICES H AND G
C
           DO 30 I = 1, 2
                DO 40 J = 1, 2
C
                     USTAR(I,J) = -C1*(C2*DLOG(R)*D(I,J)-DRD(I)*DRD(J))
                IF (ICOD  EQ  4) GO TO 40
                     PSTAR(I,J) = -C3*((C4*D(I,J)+2 *DRD(I)*DRD(J))*DRDN
     &                                 +C4*(DRD(J)*DND(I)-DRD(I)*DND(J)))/R
   40      CONTINUE
   30           CONTINUE
C
      IF (ICOD  EQ  4) GO TO 900
C
           PHISC(1) =  25 * (1.-GI(L,IPOINT)) * XLENG * OME(L,IPOINT)
           PHISC(2) =  25 * (1 +GI(L,IPOINT)) * XLENG * OME(L,IPOINT)
C
           G(1,1) = G(1,1) + USTAR(1,1) * PHISC(1)
           G(1,2) = G(1,2) + USTAR(1,2) * PHISC(1)
           G(1,3) = G(1,3) + USTAR(1,1) * PHISC(2)
           G(1,4) = G(1,4) + USTAR(1,2) * PHISC(2)
C
           G(2,1) = G(2,1) + USTAR(2,1) * PHISC(1)
           G(2,2) = G(2,2) + USTAR(2,2) * PHISC(1)
           G(2,3) = G(2,3) + USTAR(2,1) * PHISC(2)
           G(2,4) = G(2,4) + USTAR(2,2) * PHISC(2)
C
           H(1,1) = H(1,1) + PSTAR(1,1) * PHISC(1)
           H(1,2) = H(1,2) + PSTAR(1,2) * PHISC(1)
           H(1,3) = H(1,3) + PSTAR(1,1) * PHISC(2)
           H(1,4) = H(1,4) + PSTAR(1,2) * PHISC(2)
C
           H(2,1) = H(2,1) + PSTAR(2,1) * PHISC(1)
           H(2,2) = H(2,2) + PSTAR(2,2) * PHISC(1)
           H(2,3) = H(2,3) + PSTAR(2,1) * PHISC(2)
           H(2,4) = H(2,4) + PSTAR(2,2) * PHISC(2)
      IF (IPOINT  EQ  NPOINT) GO TO 900
           IPOINT = IPOINT + 1
           GO TO 700
C
```

```
 2000        AL = C5 * C2 * XLENG
             AA = AL * (0 5 - DLOG(XLENG))
C
             DO 70 I = 1, 2
                   DO 80 J = 1, 4
                         IT = (J / 2) * 2 + 2 - J
                         G(I,J) = C5 * XY(I) * XY(IT) / XLENG
                   IF (IT  NE  I) GO TO 80
C
                         G(I,J) = G(I,J) + AA
   80              CONTINUE
   70        CONTINUE
             IAA = -2
       IF (ICOD  NE  3) GO TO 90
C
             IAA = 0
   90        G(1,3+IAA) = G(1,3+IAA) + AL
             G(2,4+IAA) = G(2,4+IAA) + AL
             H(1,2-IAA) = C3 * C4 * (1  + IAA)
             H(2,1-IAA) = - H(1,2-IAA)
  900  IF (ISY  EQ  1) GO TO 10000
C
             USTAR(1,2) = - USTAR(1,2)
             DO 130 J = 1, 4
             H(1,J) = - H(1,J)
             G(1,J) = - G(1,J)
  130        CONTINUE
10000  RETURN
       END
C-----------------------------
C
C            SUBPROGRAM "CALCU"
C
C-----------------------------
C
       SUBROUTINE CALCU(ICONNE,XS,YS,COLUMN,BODY,STR,DISP,ISY)
C
       IMPLICIT REAL*8(A-H,O-Z)
       DIMENSION GI(3,6), OME(3,6), D(2,2), DISP(2,1), S(2,1), T(2,1)
       DIMENSION AALPHA(4), ABETA(4), B(3,1), XY(2), PHI(2), DD(2,2,2)
       DIMENSION ICONNE(250,4), BODY(420), X(250), Y(250), STR(3,1)
       DIMENSION COLUMN(420), A(3,1), SSS(3,4), DDD(3,4), P(4,1), Q(4,1)
       DIMENSION H(2,4), G(2,4), USTAR(2,2)
C
       COMMON /CO/ X, Y
       COMMON /CONS/ C1, C2, C3, C4, C5, C6, C7, GI, OME, D
       COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
       COMMON /ALPH/ AALPHA, ABETA
       COMMON /MMM/ NEB, NEL, NEREG1, NEREG2
C
             IFLAG = 1
             ICOD = 1
C
             DO 100 J = NEB, NEL
                   IB = ICONNE(J,1)
                   IL = ICONNE(J,2)
                   P(1,1) = COLUMN(2*IB-1)
                   P(2,1) = COLUMN(2*IB)
                   P(3,1) = COLUMN(2*IL-1)
                   P(4,1) = COLUMN(2*IL)
                   Q(1,1) = BODY(2*IB-1)
                   Q(2,1) = BODY(2*IB)
                   Q(3,1) = BODY(2*IL-1)
                   Q(4,1) = BODY(2*IL)
C
             CALL FORM(IFLAG,IB,IL,XX,YY,XS,YS,DDD,SSS,DD,ISY)
```

```
C
             CALL PROD(DDD,P,A,3,4,1)
             CALL PROD(SSS,Q,B,3,4,1)
                   STR(1,1) = STR(1,1) + A(1,1) - B(1,1)
                   STR(2,1) = STR(2,1) + A(2,1) - B(2,1)
                   STR(3,1) = STR(3,1) + A(3,1) - B(3,1)
             CALL FUNC(ICOD,H,G,IB,IL,XS,YS,USTAR,XX,YY,ISY)
             CALL PROD(G,P,S,2,4,1)
             CALL PROD(H,Q,T,2,4,1)
                   DISP(1,1) = DISP(1,1) + S(1,1) - T(1,1)
                   DISP(2,1) = DISP(2,1) + S(2,1) - T(2,1)
  100        CONTINUE
C
             IFLAG= 2
             ICOD = 4
             DO 200 JJ = NEREG1, NEREG2
                   I = ICONNE(JJ,1)
                   J = ICONNE(JJ,2)
                   K = ICONNE(JJ,3)
                   L = ICONNE(JJ,4)
                   XI = X(I)
                   XJ = X(J)
                   XK = X(K)
                   XL = X(L)
                   YI = Y(I)
                   YJ = Y(J)
                   YK = Y(K)
                   YL = Y(L)
C
                   IPOINT = 1
  300              AL = AALPHA(IPOINT)
                   BE = ABETA(IPOINT)
                   XX =  25*(1-AL)*(1-BE)*XI +  25*(1+AL)*(1-BE)*XJ
     &                + .25*(1+AL)*(1+BE)*XK + .25*(1-AL)*(1+BE)*XL
                   YY =  25*(1-AL)*(1-BE)*YI +  25*(1+AL)*(1-BE)*YJ
     &                +  25*(1+AL)*(1+BE)*YK +  25*(1-AL)*(1+BE)*YL
C
                   DXDA = -  25 * (1 - BE) * XI +  25 * (1 - BE) * XJ
     &                   +  25 * (1 + BE) * XK - .25 * (1 + BE) * XL
                   DYDA = -  25 * (1 - BE) * YI +  25 * (1 - BE) * YJ
     &                   +  25 * (1 + BE) * YK -  25 * (1 + BE) * YL
C
                   DXDB = -  25 * (1 - AL) * XI -  25 * (1 + AL) * XJ
     &                   +  25 * (1 + AL) * XK +  25 * (1 - AL) * XL
C
                   DYDB = -  25 * (1 - AL) * YI -  25 * (1 + AL) * YJ
     &                   +  25 * (1 + AL) * YK +  25 * (1 - AL) * YL
C
                   XJDET = DXDA * DYDB - DXDB * DYDA
                   XJA = DABS(XJDET)
C
             CALL FORM(IFLAG,IB,IL,XX,YY,XS,YS,DDD,SSS,DD,ISY)
C
                   STR(1,1) = STR(1,1) - DD(2,1,1) * WSOL * XJA
                   STR(2,1) = STR(2,1) - DD(2,1,2) * WSOL * XJA
                   STR(3,1) = STR(3,1) - DD(2,2,2) * WSOL * XJA
C
             CALL FUNC(ICOD,H,G,IB,IL,XS,YS,USTAR,XX,YY,ISY)
C
                   DISP(1,1) = DISP(1,1) - USTAR(1,2) * WSOL * XJA
                   DISP(2,1) = DISP(2,1) - USTAR(2,2) * WSOL * XJA
                   IPOINT = IPOINT + 1
                   IF (IPOINT  LE  4) GO TO 300
C
  200        CONTINUE
       RETURN
```

```
      END
C-----------------------------
C
C           SUBPROGRAM "PROD1"
C
C-----------------------------
C
      SUBROUTINE PROD1(A,B,C,M,IN,IDCOLU,CC,CD)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(M,M), B(M), C(M), IDCOLU(M)
C
      COMMON /NNN/ NNSTR2,NNINT4,NNSOL2,NESTR,NEJON,NEINT,NESOL,NEREG
C
            DO 10 I = 1, IN
                        C(I) = O
                        DO 20 K = 1, IN
                              C(I) = C(I) + A(I,K) * B(K)
   20                   CONTINUE
   10       CONTINUE
C
            MM = NNSTR2 + 1
            NN = MM + NNINT4 / 2 - 1
            DO 40 I = MM, NN
                  C(I) = CD * C(I)
   40       CONTINUE
C
            MM = NN + 1
            NN = MM + NNINT4 / 2 - 1
            DO 50 I = MM, NN
                  C(I) = CC * C(I)
   50       CONTINUE
C
            MM = NN + 1
            DO 60 I = MM, IN
            IF (IDCOLU(I)  EQ  O) GO TO 70
                  C(I) = CC * C(I)
                  GO TO 60
   70             C(I) = CD * C(I)
   60       CONTINUE
      RETURN
      END
C-----------------------------
C
C           SUBPROGRAM "FORM"
C
C-----------------------------
C
      SUBROUTINE FORM(IFLAG,IB,IL,XX,YY,XS,YS,DDD,SSS,DD,ISY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION GI(3,6), OME(3,6), D(2,2), SSS(3,4), DDD(3,4)
      DIMENSION XY(2), PHI(2), DD(2,2,2), SS(2,2,2), DND(2), DRD(2)
      DIMENSION X(250), Y(250)
C
      COMMON /CO/ X, Y
      COMMON /CONS/ C1, C2, C3, C4, C5, C6, C7, GI, OME, D
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
C
            DO 10 KK = 1, 3
                  DO 20 L = 1, 4
                        SSS(KK,L) = O
                        DDD(KK,L) = O
   20             CONTINUE
   10       CONTINUE
C
```

```
            XY(1) = X(IL) - X(IB)
            XY(2) = Y(IL) - Y(IB)
            XLENG = (XY(1) * XY(1) + XY(2) * XY(2)) ** O 5
      IF (IFLAG  EQ  2) GO TO 155
C
            DND(1) = XY(2) / XLENG
            DND(2) = - XY(1) / XLENG
            SEL = O 5 * DSQRT((2  * XS - X(IB) - X(IL)) ** 2 + (2  *
     &            YS - Y(IB) - Y(IL)) ** 2) / XLENG
      IF (SEL  LE  1 5) GO TO 115
C
      IF (SEL  LE  5 5) GO TO 125
C
            L = 1
            NPOINT = 2
            GO TO 135
C
  115       L = 3
            NPOINT = 6
            GO TO 135
C
  125       L = 2
            NPOINT = 4
  135       IPOINT = 1
  500       XX = (X(IB)+X(IL)) / 2  + XY(1) / 2  * GI(L,IPOINT)
            YY = (Y(IB)+Y(IL)) / 2  + XY(2) / 2  * GI(L,IPOINT)
            PHI(1) =  25 * (1.-GI(L,IPOINT))*XLENG*OME(L,IPOINT)
            PHI(2) =  25 * (1 +GI(L,IPOINT))*XLENG*OME(L,IPOINT)
C
  155       XDD = XX - XS
            YDD = YY - YS
            R = (XDD * XDD + YDD * YDD) ** O 5
            DRD(1) = XDD / R
            DRD(2) = YDD / R
            DRDN = DRD(1) * DND(1) + DRD(2) * DND(2)
C
            DO 30 I = 1, 2
                  DO 40 J = 1, 2
                        DO 50 K = 1, 2
                              DD(K,I,J)=C3*(C4*(DRD(J)*D(K,I)+DRD(I)*D(K,J)-
     &                        DRD(K)*D(I,J))+2.*DRD(I)*DRD(J)*DRD(K))/R
                        IF (IFLAG  EQ  2) GO TO 50
C
                              B1=2.*DRDN*(C4*DRD(K)*D(I,J)+PSOL*(DRD(J)*
     &                        D(I,K)+DRD(I)*D(J,K))-4 *DRD(I)*DRD(J)*DRD(K))
C
                              B2=2 *PSOL*(DND(I)*DRD(J)*DRD(K)+DND(J)*
     &                        DRD(I)*DRD(K))
C
                              B3=C4*(2 *DND(K)*DRD(I)*DRD(J)+DND(J)*D(I,K)+
     &                        DND(I)*D(J,K))
C
                              SS(K,I,J)=C6*(B1+B2+B3-C7*DND(K)*D(I,J))/R**2
   50                   CONTINUE
   40             CONTINUE
   30       CONTINUE
C
      IF (IFLAG  EQ  2) GO TO 700
C
            DDD(1,1) = DDD(1,1) + PHI(1) * DD(1,1,1)
            DDD(1,2) = DDD(1,2) + PHI(1) * DD(2,1,1)
            DDD(1,3) = DDD(1,3) + PHI(2) * DD(1,1,1)
            DDD(1,4) = DDD(1,4) + PHI(2) * DD(2,1,1)
            DDD(2,1) = DDD(2,1) + PHI(1) * DD(1,1,2)
            DDD(2,2) = DDD(2,2) + PHI(1) * DD(2,1,2)
            DDD(2,3) = DDD(2,3) + PHI(2) * DD(1,1,2)
```

```
         DDD(2,4) = DDD(2,4) + PHI(2) * DD(2,1,2)
         DDD(3,1) = DDD(3,1) + PHI(1) * DD(1,2,2)
         DDD(3,2) = DDD(3,2) + PHI(1) * DD(2,2,2)
         DDD(3,3) = DDD(3,3) + PHI(2) * DD(1,2,2)
         DDD(3,4) = DDD(3,4) + PHI(2) * DD(2,2,2)
C
         SSS(1,1) = SSS(1,1) + PHI(1) * SS(1,1,1)
         SSS(1,2) = SSS(1,2) + PHI(1) * SS(2,1,1)
         SSS(1,3) = SSS(1,3) + PHI(2) * SS(1,1,1)
         SSS(1,4) = SSS(1,4) + PHI(2) * SS(2,1,1)
         SSS(2,1) = SSS(2,1) + PHI(1) * SS(1,1,2)
         SSS(2,2) = SSS(2,2) + PHI(1) * SS(2,1,2)
         SSS(2,3) = SSS(2,3) + PHI(2) * SS(1,1,2)
         SSS(2,4) = SSS(2,4) + PHI(2) * SS(2,1,2)
         SSS(3,1) = SSS(3,1) + PHI(1) * SS(1,2,2)
         SSS(3,2) = SSS(3,2) + PHI(1) * SS(2,2,2)
         SSS(3,3) = SSS(3,3) + PHI(2) * SS(1,2,2)
         SSS(3,4) = SSS(3,4) + PHI(2) * SS(2,2,2)
      IF (IPOINT  EQ  NPOINT) GO TO 700
         IPOINT = IPOINT + 1
         GO TO  500
C
  700 IF (ISY  EQ  1) GO TO 1000
C
         DD(2,1,2) = -DD(2,1,2)
         DO 25 J = 1, 4
            DDD(2,J) = - DDD(2,J)
            SSS(2,J) = - SSS(2,J)
   25    CONTINUE
 1000 RETURN
      END
C------------------------------
C
C        SUBPROGRAM "INPUT"
C
C------------------------------
C
      SUBROUTINE INPUT(ICONNE,IDCOLU,COLUMN,IUNKNO,ITYPE,IER)
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 CARD(80), WORD(6), TIT(3), STR(3), SOI(3), PRO(3)
      CHARACTER*1 INTE(3), NODCOO(6), PRELOA(6), PRESTR(6), PREDIS(6)
      CHARACTER*1 ELECON(6), TYP(3)
C
      DIMENSION VAR(5), ICONNE(250,4), IDCOLU(420)
      DIMENSION COLUMN(420), X(250), Y(250), AALPHA(4), ABETA(4)
      DIMENSION GI(3,6), OME(3,6), D(2,2)
C
      COMMON /CO/ X, Y
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
      COMMON /NNN/ NNSTR2,NNINT4,NNSOL2,NESTR,NEJON,NEINT,NESOL,NEREG
      COMMON /IO/IRE,IWR
      COMMON /ALPH/ AALPHA, ABETA
      COMMON /CONS/ C1, C2, C3, C4, C5, C6, C7, GI, OME, D
C
      DATA TIT/1HT,1HI,1HT/, STR/1HS,1HT,1HR/, SOI/1HS,1HO,1HI/
      DATA PRO/1HP,1HR,1HO/, INTE/1HI,1HN,1HT/, TYP/1HT,1HY,1HP/
      DATA NODCOO/1HN,1HO,1HD,1HC,1HO,1HO/
      DATA PREDIS/1HP,1HR,1HE,1HD,1HI,1HS/
      DATA PRELOA/1HP,1HR,1HE,1HL,1HO,1HA/
      DATA PRESTR/1HP,1HR,1HE,1HS,1HT,1HR/
      DATA ELECON/1HE,1HL,1HE,1HC,1HO,1HN/
C
C----    INITIALIZATION OF GAUSS INTEGRATING POINTS GI AND WEIGHTS OME
C
         AALPHA(1) = - 577350269189626
         AALPHA(2) = + 577350269189626
         AALPHA(3) = + 577350269189626
         AALPHA(4) = - 577350269189626
C
         ABETA(1)  = - 577350269189626
         ABETA(2)  = - 577350269189626
         ABETA(3)  = + 577350269189626
         ABETA(4)  = + 577350269189626
C
         GI(1,1) = - 0 577350269189626
         GI(1,2) = - GI(1,1)
C
         OME(1,1) = 1 000000000000000
         OME(1,2) = 1 000000000000000
C
         GI(2,1) = - 0 861136311594053
         GI(2,2) = - 0 339981043584856
         GI(2,3) = - GI(2,2)
         GI(2,4) = - GI(2,1)
C
         OME(2,1) = 0 347854845137454
         OME(2,2) = 0 652145154862546
         OME(2,3) = OME(2,2)
         OME(2,4) = OME(2,1)
C
         GI(3,1) = - 0.932469514203152
         GI(3,2) = - 0.661209386466265
         GI(3,3) = - 0.238619186083197
         GI(3,4) = - GI(3,3)
         GI(3,5) = - GI(3,2)
         GI(3,6) = - GI(3,1)
C
         OME(3,1) =  0 171324492379170
         OME(3,2) =  0.360761573048139
         OME(3,3) =  0 467913934572691
         OME(3,4) =  OME(3,3)
         OME(3,5) =  OME(3,2)
         OME(3,6) =  OME(3,1)
C
         IUNKNO = 0
         NETO = 0
C
C----    STRUCTURE  ICHECK = -1
C----    INTERFACE  ICHECK =  0
C----    SOIL       ICHECK =  1
C
         ICHECK = -1
C
C----    TYPE CARDS
C
         DO 2 J = 1, 3
            WORD(J) = TYP(J)
    2    CONTINUE
C
C----    ID  IDENTIFIER
C----    NW. NUMBER OF WORDS, 3 CHARACTERS PER WORD
C----    NV: NUMBER OF VARIABLES
C----    READ TYPE CARD
C
      READ (IRE,4) (CARD(K), K= 1,80)
         ID = 1
         NW = 1
         NV = 1
      CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
      IF (IER  EQ  0) GO TO 15
   11 WRITE (IWR,6)
```

```
C
             GO TO 10000
    15       ITYPE = VAR(1)
       IF (ITYPE  EQ  1) GO TO 13
       IF (ITYPE  EQ  2) GO TO 16
             IER = 1
    13 WRITE (IWR,7)
       GO TO 17
    16 WRITE (IWR,8)
C
C----        SET WORDS FOR TITLE CARD
C
    17       DO 10 J = 1, 3
                   WORD(J) = TIT(J)
    10       CONTINUE
C
C----        READ TITLE CARD
C
       READ (IRE,4) (CARD(K), K= 1,80)
       CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
       IF (IER  EQ  O) GO TO 20
C
       WRITE (IWR, 9)
             GO TO 10000
C
    20 WRITE (IWR, 93)
             JJ = VAR(1)
C
C----        READ INPUT CARDS OF TITLE
C
             DO 30 J = 1, JJ
             READ (IRE,4) (CARD(K), K=1,80)
             WRITE (IWR,96) (CARD(K), K=1,80)
    30       CONTINUE
C
C----        SET WORDS FOR PROPERTY CARDS
C
  5000       ID = 1
             NW = 2
       IF (ICHECK) 100, 200, 300
   100       DO 110 J = 1, 3
                   WORD(J) = STR(J)
   110       CONTINUE
             NV = 3
             GO TO 400
C
   200       DO 210 J = 1, 3
                   WORD(J) = INTE(J)
   210       CONTINUE
             NV = 2
             GO TO 400
C
   300       DO 310 J = 1, 3
                   WORD(J) = SOI(J)
   310       CONTINUE
             NV = 3
C
   400       DO 410 J = 1, 3
                   WORD(J+3) = PRO(J)
   410       CONTINUE
C
C----        READ PROPERTY CARD
C
       READ (IRE,4) (CARD(K), K= 1,80)
             NW = 2
       CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
```

```
       IF (IER  EQ  O) GO TO 415
C
       WRITE (IWR, 99)
             GO TO 10000
C
   415 IF (ICHECK) 420, 430, 440
   420       ESTR = VAR(1)
             PSTR = VAR(2)
             WSTR = VAR(3)
       WRITE (IWR,903) ESTR, PSTR, WSTR
             GO TO 450
C
   430       EKN  = VAR(1)
             EKS  = VAR(2)
       WRITE (IWR,906) EKN, EKS
             GO TO 450
C
   440       ESOL = VAR(1)
             PSOL = VAR(2)
             WSOL = VAR(3)
       WRITE (IWR,909) ESOL, PSOL, WSOL
C
C----        SET WORDS FOR COORDINATE CARDS
C
   450       DO 460 J = 1, 6
                   WORD(J) = NODCOO(J)
   460       CONTINUE
C
C----        READ COORDINATE CARD
C
       READ (IRE,4) (CARD(K), K= 1,80)
             NV = 1
       IF (ICHECK  NE  1) GO TO  468
             NV = 2
   468 CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
       IF (IER  EQ  O) GO TO 480
C
   470 WRITE (IWR, 913)
             GO TO 10000
C
   480       NN = VAR(1)
       IF (ICHECK) 490, 500, 510
C
C----        STRESSES OR LOADS ARE ASSUMED TO BE O AT THE BEGINNING
C----        IDCOLU = O STRESSES OR LOADS ARE PRESCRIBED
C----        IDCOLU = 1 DISPLACEMENTS ARE PRESCRIBED
C
   490       NNSTR2 = 2 * NN
             IUNKNO = IUNKNO + NNSTR2
       WRITE (IWR,916) NN
             LL = 1
             GO TO 520
C
   500       NNINT4 = 4 * NN
             IUNKNO = IUNKNO + NNINT4
       WRITE (IWR,919) NN
             LL = IUNKNO - NNINT4 + 1
             GO TO 520
C
   510       NNSOL2 = 2 * NN
             IUNKNO = IUNKNO + NNSOL2
             MM = VAR(2)
       WRITE (IWR,923) NN, MM
             NN = NN + MM
             LL = IUNKNO - NNSOL2 + 1
C
```

```
  520       DO 525 J = LL, IUNKNO
                 IDCOLU(J) = O
                 COLUMN(J) = O
  525       CONTINUE
C
       WRITE (IWR,926)
            ID = 2
            NV = 3
C
C----       READ INPUT CARDS OF COORDINATE
C
            DO 530 J = 1,NN
            READ (IRE,4) (CARD(K),K=1,80)
            CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
            IF (IER  NE  O) GO TO 470
C
                 KK = VAR(1)
                 X(KK) = VAR(2)
                 Y(KK) = VAR(3)
            WRITE (IWR,929) KK, X(KK), Y(KK)
  530       CONTINUE
C
C----       SET WORDS FOR CONNECTIVITY CARDS
C
            DO 540 J = 1, 6
                 WORD(J) = ELECON(J)
  540       CONTINUE
C
C----       READ CONNECTIVITY CARD
C
       READ (IRE,4) (CARD(K), K= 1,80)
            ID =1
            NW = 2
       IF (ICHECK  NE  O) GO TO 535
            NV = 1
            GO TO 555
  535       NV = 2
  555  CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
       IF (IER  EQ  O) GO TO 545
C
  550  WRITE (IWR, 933)
            GO TO 10000
C
C----       READ INPUT CARDS OF CONNECTIVITY
C
  545  IF (ICHECK) 560, 580, 600
C
  560       KK = VAR(1)
            NESTR = KK
            NEJON = VAR(2)
            NETO = NETO + NESTR + NEJON
       WRITE (IWR,936) KK, NEJON
            NN = KK + NEJON
            ID = 3
  562       NV = 5
C
            DO 565 J = 1, NN
            READ (IRE,4) (CARD(K),K=1,80)
            CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
            IF (IER  NE  O) GO TO 550
C
                 M = VAR(1)
C
                 DO 570 K = 1, 4
                      ICONNE(M,K) = VAR(K+1)
  570            CONTINUE
C           IF (ICHECK  EQ  -1) GO TO 575
            IF (J  LE  KK) GO TO 575
            WRITE (IWR,937) M, (ICONNE(M,K), K=1,4)
                 GO TO 565
  575       WRITE (IWR,938) M, (ICONNE(M,K), K=1,4)
  565       CONTINUE
C
            GO TO 620
C
  580       NM = VAR(1)
            NEINT = NM
            NETO = NETO + NEINT
       WRITE (IWR,939) NM
            ID = 3
  582       NV = 3
C
            DO 585 J = 1, NM
            READ (IRE,4) (CARD(K),K=1,80)
            CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
            IF (IER  NE  O) GO TO 550
C
            M = VAR(1)
                 DO 590 K = 1, 2
                      ICONNE(M,K) = VAR(K+1)
  590            CONTINUE
C
            WRITE (IWR,941) M, (ICONNE(M,K), K=1,2)
  585       CONTINUE
C
       IF (ICHECK  EQ  1) GO TO 605
C
            GO TO 620
C
  600       NM = VAR(1)
            NN = VAR(2)
            NESOL = NM
            NEREG = NN
            NETO = NETO + NESOL + NEREG
       WRITE (IWR,943) NM
            ID =3
            GO TO 582
C
  605  WRITE (IWR,944) NN
       IF (NN  EQ  O) GO TO 620
            GO TO 562
C
C----       SET WORDS FOR PRESCRIBED DISPLACEMENTS
C
  620       DO 640 J = 1, 6
                 WORD(J) = PREDIS(J)
  640       CONTINUE
C
C----       READ PRESCRIBED DISPLACEMENT CARD
C
       READ (IRE,4) (CARD(K), K= 1,80)
            ID = 1
            NW = 2
            NV = 1
       CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
       IF (IER  EQ  O) GO TO 645
C
  650  WRITE (IWR,946)
            GO TO 10000
C
  645       JJ = VAR(1)
       IF (ICHECK) 655  658  660
```

```
  655 WRITE (IWR,949) JJ
          GO TO 670
C
  658 WRITE (IWR,951) JJ
          GO TO 670
C
  660 WRITE (IWR,953) JJ
  670 IF (JJ  EQ 0) GO TO 800
          ID = 2
C
C----     READ INPUT CARDS OF PRESCRIBED DISPLACEMENTS
C
          DO 700 K = 1, JJ
          READ (IRE,4) (CARD(J), J=1, 80)
          CALL CONV2(IER,CARD,WORD,VAR,ID)
          IF (IER  NE  0) GO TO 650
C
               KK = VAR(1)
               IP = 1
          IF (ICHECK  EQ  1) GO TO 705
C
               MM = 2 * KK - 1
               GO TO 710
C
  705          MM = 2 * KK - 1 + NNINT4 / 2
  710     IF (WORD(IP)  EQ  ' ') GO TO 740
C
          IF (WORD(IP)  NE  'X') GO TO 720
C
               IDCOLU(MM) = 1
               COLUMN(MM) = VAR(IP+1)
               GO TO 730
C
  720          IDCOLU(MM+1) = 1
               COLUMN(MM+1) = VAR(IP+1)
  730          IP = IP + 1
               GO TO 710
C
  740     IF (IP  EQ  3) GO TO 760
C
          WRITE (IWR,956) KK, WORD(1), VAR(2)
               GO TO 700
C
  760     WRITE (IWR,959) KK, WORD(1), VAR(2), WORD(2), VAR(3)
  700     CONTINUE
C
C----     SET WORD FOR PRESCRIBED NON-ZERO LOADS OR STRESSES
C
  800 IF (ICHECK) 830, 830, 834
C
  830     DO 840 J = 1, 6
               WORD(J) = PRELOA(J)
  840     CONTINUE
          GO TO 838
C
  834     DO 848 J = 1, 6
               WORD(J) = PRESTR(J)
  848     CONTINUE
C
C----     READ PRESCRIBED LOAD OR STRESS CARDS
C
  838 READ (IRE,4) (CARD(K), K= 1,80)
          ID = 1
          NW = 2
          NV = 1
      CALL CONV1(IER,NW NV,CARD WORD VAR ID)
```

```
      IF (IER  EQ  0) GO TO 845
C
  855 WRITE (IWR,963)
          GO TO 10000
C
  845     JJ = VAR(1)
      IF (ICHECK) 860, 865, 870
C
  860 WRITE (IWR,966) JJ
          GO TO 880
C
  865 WRITE (IWR,968) JJ
          GO TO 880
C
  870 WRITE (IWR,969) JJ
  880 IF (JJ  EQ.0) GO TO 1200
          ID = 2
C
C----     READ INPUT CARDS OF PRESCRIBED LOADS OR STRESSES
C
          DO 900 K = 1, JJ
          READ (IRE,4) (CARD(J), J=1, 80)
          CALL CONV2(IER,CARD,WORD,VAR,ID)
          IF (IER .NE. 0) GO TO 855
C
               KK = VAR(1)
               IP = 1
          IF (ICHECK  EQ  1) GO TO 905
C
               MM = 2 * KK - 1
               GO TO 910
C
  905          MM = 2 * KK - 1 + NNINT4 / 2
  910     IF (WORD(IP)  EQ  ' ') GO TO 940
C
          IF (WORD(IP)  NE. 'X') GO TO 920
C
               COLUMN(MM) = VAR(IP+1)
               GO TO 930
C
  920          COLUMN(MM+1) = VAR(IP+1)
  930          IP = IP + 1
               GO TO 910
C
  940     IF (ICHECK) 925, 1200, 935
C
  925     IF (IP  EQ. 3) GO TO 960
C
          WRITE (IWR,973) KK, WORD(1), VAR(2)
               GO TO 900
C
  960     WRITE (IWR,976) KK, WORD(1), VAR(2), WORD(2), VAR(3)
               GO TO 900
C
  935     IF (IP  EQ  3) GO TO 965
C
          WRITE (IWR,979) KK, WORD(1), VAR(2)
               GO TO 900
C
  965     WRITE (IWR,983) KK, WORD(1), VAR(2), WORD(2), VAR(3)
  900     CONTINUE
C
 1200 IF (ICHECK) 1300, 1400, 10000
C
 1300     ICHECK = 0
          GO TO 5000
```

```
C
 1400       ICHECK = 1
            GO TO 5000
C
    4 FORMAT(80A1)
C
    6 FORMAT(///,26X,'******   TYPE CARD ARE LOST OR WRONG    ******')
C
    7 FORMAT(///,44X,'*********     UNSYMMETRICAL   CASE       *********')
C
    8 FORMAT(///,44X,'*********  SYMMETRICAL  ABOUT  Y  AXIS  *********')
C
    9 FORMAT(///,26X,'******   TITLE CARDS ARE LOST OR WRONG    ******')
C
   93 FORMAT(/////,26X,'PROBLEM IDENTIFICATION ',
     &             /,26X,'-----------------------',/)
C
   96 FORMAT(36X,80A1)
C
   99 FORMAT(//,26X,'******  PROPERTY CARDS ARE LOST OR WRONG  ******',
     &          /,26X,'                        OR                     ',
     &          /,26X,'**** NUMBER OF TITLE CARDS IS INCONSISTENT *****',
     &          /,26X,'                        OR                     ',
     &          /,26X,'*NUMBER OF STRUCTURE LOAD CARDS IS INCONSISTENT*',
     &          /,26X,'                        OR                     ',
     &          /,26X,'****   NUMBER OF INTERFACE CONNECTIVITY    *****',
     &          /,26X,'*****       CARDS IS INCONSISTENT          *****')
C
  903 FORMAT(///,26X,'STRUCTURE INPUT ',
     &           /,26X,'----------------',
     &          //,36X,'ELASTIC MODULUS = ',E11 4,'   KSI',
     &           /,36X,'POISSON RATIO   = ',E11 4,
     &           /,36X,'UNIT SELFWEIGHT = ',E11 4,'   KCI')
C
  906 FORMAT(///,26X,'INTERFACE INPUT ',
     &           /,26X,'----------------',
     &          //,36X,'UNIT NORMAL STIFFNESS         = ',E11 4,'   KPI',
     &           /,36X,'UNIT TANGENTIAL STIFFNESS     = ',E11 4,'   KPI')
C
  909 FORMAT(///,26X,'SOIL INPUT ',
     &           /,26X,'-----------',
     &          //,36X,'ELASTIC MODULUS = ', E11 4,'   KSI',
     &           /,36X,'POISSON RATIO   = ', E11 4,
     &           /,36X,'UNIT SELFWEIGHT = ', E11 4,'   KCI')
C
  913 FORMAT(//,26X,'****** COORDINATE CARDS ARE LOST OR WRONG ******')
C
  916 FORMAT(//,36X,'NUMBER OF NODES IN STRUCTURE = ',I3)
C
  919 FORMAT(//,36X,'NUMBER OF NODES IN INTERFACE = ',I3)
C
  923 FORMAT(//,36X,'NUMBER OF NODES IN SOIL         = ',I3,
     &          /,36X,'NUMBER OF NODES FOR INTEGRATION = ',I3)
C
  926 FORMAT(/,36X,'NODE',10X,'X COORDINATE',10X,'Y COORDINATE')
C
  929 FORMAT(36X,I3,11X,E11 4,11X,E11 4)
C
  933 FORMAT(//,26X,'***** CONNECTIVITY CARDS ARE LOST OR WRONG *****',
     &          /,26X,'                        OR                     ',
     &          /,26X,'** NUMBER OF COORDINATE CARDS IS INCONSISTENT **')
C
  936 FORMAT(//,36X,'NUMBER OF ELEMENTS IN STRUCTURE = ',I3,
     &          /,36X,'NUMBER OF JOINT ELEMENTS        = ',I3,
     &    //,36X,'ELEMENT NO',4X,'NODE 1',6X,'NODE 2',6X,'NODE 3',
     &           6X,'NODE 4')
```

```
C
  937 FORMAT(39X,I3,4(9X,I3),'    JOINT ELEMENT')
C
  938 FORMAT(39X,I3,4(9X,I3))
C
  939 FORMAT(//,36X,'NUMBER OF ELEMENTS IN INTERFACE = ',I3,
     &    //,36X,'ELEMENT NO',4X,'NODE 1',6X,'NODE 2')
C
  941 FORMAT(39X,I3,2(9X,I3))
C
  943 FORMAT(//,36X,'NUMBER OF ELEMENTS IN SOIL      = ',I3,
     &    //,36X,'ELEMENT NO',4X,'NODE 1',6X,'NODE 2')
C
  944 FORMAT(//,36X,'NUMBER OF ELEMENTS FOR INTEGRATION = ',I3,
     &    //,36X,'ELEMENT NO',4X,'NODE 1',6X,'NODE 2',6X,'NODE 3',
     &           6X,'NODE 4')
C
  946 FORMAT(//,26X,'***** DISPLACEMENT CARDS ARE LOST OR WRONG *****',
     &          /,26X,'                        OR                      ',
     &          /,26X,'* NUMBER OF CONNECTIVITY CARDS IS INCONSISTENT *')
C
  949 FORMAT(//,36X,'NUMBER OF PRESCRIBED DISPLACEMENT',
     &                 ' IN STRUCTURE = ',I3,/)
C
  951 FORMAT(//,36X,'NUMBER OF PRESCRIBED DISPLACEMENT',
     &                 ' IN INTERFACE = ',I3,/)
C
  953 FORMAT(//,36X,'NUMBER OF PRESCRIBED DISPLACEMENT',
     &                 ' IN SOIL = ',I3,/)
C
  956 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,'INCH')
C
  959 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,A1,' = ',E11 4,
     &        '     INCH')
C
  963 FORMAT(//,26X,'**** LOAD OR STRESS CARDS ARE LOST OR WRONG ****',
     &          /,26X,'                        OR                      ',
     &          /,26X,'* NUMBER OF DISPLACEMENT CARDS IS INCONSISTENT *')
C
  966 FORMAT(//,36X,'NUMBER OF PRESCRIBED NON-ZERO LOAD IN STRUCTURE',
     &                  ' = ',I3,/)
C
  968 FORMAT(//,36X,'NUMBER OF PRESCRIBED NON-ZERO LOAD IN INTERFACE',
     &                  ' = ',I3,/)
C
  969 FORMAT(//,36X,'NUMBER OF PRESCRIBED NON-ZERO TRACTION IN SOIL',
     &                  ' = ',I3,/)
C
  973 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,'KIPS')
C
  976 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,A1,' = ',E11 4,
     &        '    KIPS')
C
  979 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,'KPI')
C
  983 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,A1,' = ',E11 4,
     &        '     KPI')
C
10000 RETURN
      END
C----------------------------
C
C          SUBPROGRAM "CONV1"
C
C----------------------------
C
```

```
      SUBROUTINE CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 CARD(80), WORD(6)
      DIMENSION VAR(5)
C
C----     FOR MAJOR INPUT CARD (START WITH CHARACTER), ID = 1
C----     FOR COORDINATE, DISPLACEMENT OR LOAD, ID = 2 (MAYBE NEGATIVE)
C----     FOR CONNECTIVITY CARD ID = 3 (POSITIVE VALUE ONLY)
C
          J = 1
      IF (ID  NE 1) GO TO 65
C
C----     FIND CHARACTER
C
          DO 100 K = 1, NW
   20     IF (J  GE  77) GO TO 500
C
          IF ((CARD(J)  NE  ' ')  AND  (CARD(J)  NE  ',')) GO TO 30
C
              J = J + 1
              GO TO 20
C
   30     IF ((CARD(J)  NE  WORD(3*K-2))  OR  (CARD(J+1)  NE
    &     WORD(3*K-1))  OR  (CARD(J+2)  NE  WORD(3*K))) GO TO 500
C
              J = J + 3
   60     IF (J  GE  80) GO TO 500
C
          IF ((CARD(J)  EQ  ' ')  OR  (CARD(J)  EQ  ',')) GO TO 100
C
              J = J + 1
              GO TO 60
C
  100     CONTINUE
C
C----     FIND VARIABLE
C
   65     DO 200 K = 1, NV
              VAR(K) = 0
   70     IF (J GE 80) GO TO 500
C
          IF ((CARD(J)  NE  ' ')  AND  (CARD(J)  NE  ',')) GO TO 80
C
              J = J + 1
              GO TO 70
C
   80     CALL VALUE(IER,CARD,VAR(K),ID,J)
          IF (VAR(1)  LT  0 ) GO TO 500
C
  200     CONTINUE
C
          GO TO 1000
C
  500     IER = 1
 1000 RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "CONV2"
C
C-----------------------------
C
      SUBROUTINE CONV2(IER,CARD,WORD,VAR,ID)
C
      IMPLICIT REAL*8(A-H,O-Z)
```

```
      CHARACTER*1 CARD(80), WORD(6)
      DIMENSION VAR(5)
C
C----     INITIALIZATION
C
          WORD(1) = ' '
          WORD(2) = ' '
          VAR(1) = 0
          VAR(2) = 0
          VAR(3) = 0
          J = 1
          IC = 1
   10 IF (J  GE  80) GO TO 500
C
      IF ((CARD(J)  NE  ' ') .AND  (CARD(J)  NE  ',')) GO TO 100
C
          J = J + 1
          GO TO 10
C
  100 CALL VALUE(IER,CARD,VAR(IC),ID,J)
      IF ((IER  NE  0)  OR  (VAR(1)  LT  1 )) GO TO 500
C
   20 IF ((J  GE  80)  AND  (IC  NE  1)) GO TO  900
C
      IF ((J  GE  80)  AND  (IC  EQ  1)) GO TO 500
C
      IF ((CARD(J)  NE. ' ')  AND  (CARD(J)  NE  ',')) GO TO 200
C
          J = J + 1
          GO TO 20
C
  200 IF ((CARD(J)  NE  'X')  AND  (CARD(J)  NE. 'Y')) GO TO 500
C
          WORD(IC) = CARD(J)
          IC = IC + 1
          J = J + 1
          GO TO 10
C
  900 IF (WORD(1)  EQ  WORD(2)) GO TO 500
          WORD(3) = ' '
          GO TO 1000
  500     IER = 1
 1000 RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "PROD"
C
C-----------------------------
C
      SUBROUTINE PROD(A,B,C,M,L,N)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(M,L), B(L,N), C(M,N)
C
          DO 10 I = 1, M
              DO 20 J = 1, N
                  C(I,J) = 0
                  DO 30 K = 1, L
                      C(I,J) = C(I,J) + A(I,K) * B(K,J)
   30             CONTINUE
   20         CONTINUE
   10     CONTINUE
C
      RETURN
      END
```

```
C-----------------------------
C
C         SUBPROGRAM "VALUE"
C
C-----------------------------
C
      SUBROUTINE VALUE(IER,CARD,VAL,ID,J)
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 CARD(80)
      DIMENSION IDIGIT(16)
C
C----     ND  NUMBER  OF DIGIT
C----     IP. LOCATION OF " "
C----     INITIALIZATION
C
          SIGN = 1 0
          ND = 1
          IP = 0
          VAL = 0
      IF ((CARD(J)  NE  '-')  AND  (CARD(J)  NE  '+')) GO TO 10
C
      IF (CARD(J)  EQ  '+') GO TO 15
C
      IF (ID  NE  2) GO TO 500
C
          SIGN = -1
   15     J = J + 1
      IF (J  GT  80) GO TO 500
      IF (CARD(J)  EQ  ' ') GO TO 15
   10     KK = J
C
          DO 1000 K = KK, 80
          IF (CARD(K)  EQ  '0') GO TO 20
C
          IF (CARD(K)  EQ  '1') GO TO 30
C
          IF (CARD(K)  EQ  '2') GO TO 40
C
          IF (CARD(K)  EQ  '3') GO TO 50
C
          IF (CARD(K)  EQ  '4') GO TO 60
C
          IF (CARD(K)  EQ  '5') GO TO 70
C
          IF (CARD(K)  EQ  '6') GO TO 80
C
          IF (CARD(K) .EQ  '7') GO TO 90
C
          IF (CARD(K)  EQ  '8') GO TO 100
C
          IF (CARD(K)  EQ  '9') GO TO 110
C
          IF (CARD(K)  EQ  ' ') GO TO 120
C
          IF ((CARD(K)  NE  ' ')  AND  (CARD(K)  NE  ',')) GO TO 500
C
              J = K + 1
              GO TO 125
C
   20         IDIGIT(ND) = 0
              GO TO 115
C
   30         IDIGIT(ND) = 1
              GO TO 115
C
```

```
   40         IDIGIT(ND) = 2
              GO TO 115
C
   50         IDIGIT(ND) = 3
              GO TO 115
C
   60         IDIGIT(ND) = 4
              GO TO 115
C
   70         IDIGIT(ND) = 5
              GO TO 115
C
   80         IDIGIT(ND) = 6
              GO TO 115
C
   90         IDIGIT(ND) = 7
              GO TO 115
C
  100         IDIGIT(ND) = 8
              GO TO 115
C
  110         IDIGIT(ND) = 9
  115         ND = ND + 1
              GO TO 1000
C
  120         IP = ND
 1000     CONTINUE
C
  125 IF (IP  NE  0) GO TO 130
C
          IP = ND
C
  130     NM = ND - 1
C
          DO 200 K = 1, NM
              VAL = VAL + SIGN * IDIGIT(K) * 10  ** (IP-K-1)
  200     CONTINUE
C
          GO TO 2000
C
  500     IER = 1
 2000 RETURN
      END
$ENTRY
  TYPE 2
  TITLE 5
THIS PROBLEM IS TO TEST A U-FRAME STRUCTURE BY THE COUPLING OF BOUNDARY
AND FINITE ELEMENT METHODS.  IT INCLUDES 209 NODES AND 221 ELEMENTS
THE RESULTS ARE COMPARED WITH THOSE OBTAINED FROM THE FINITE ELEMENT
METHOD ALONE   NOTE THAT TWO SMALL ELEMENTS ARE USED AT EACH END OF
INTERFACE TO AVOID DISPLACEMENT DISCONTINUITIES
    STR PROP 3000  25  0 00008681
NODE COOR 90
 1 0  576
  2 0,  540
3  0          504
  4  0    468
 5   0   432
 6 144  576
  7 144,  540
8  144          504
  9  144    468
 10   144  432
11 264  576
 12 264,  540
13 264          504
```

```
  14 264     468
 15   264  432
16 360  576
 17 360,  540
18 360           504
  19 360     468
 20   360  432
    21      432      1392
22     432     1272
 23              432     +1128
  24     + 432,..       984
    25     432   840
  26   432   768
    27     432     696
   28      432       576
  29     432    540
30         432     504
31       432      468
  32   432,,     432
33       456      1392
34     456     1272
  35   456    1128
 36        462     984
   37    468         840
 38      468     768
39       480      696
 40      480                           624
 41    480      576
42       480       528
 43       504         480
 44         516,   450
   45       480      1392
  46      480,, 1272
 47    480    1128
48         492,..      984
  49     504   840
  50         504  768
  51          528      696
  52       528   648
  53       528                600
    54        528      552
      55       576        504
      56       600    468
      57       504       1392
    58       504    1272
      59       504      1128
    60       522  984
    61        540       840
   62        540      768
  63     564    696
  64    600    552
  65        648       504
    66      684       486
    67              552        1392
      68      552,    1344
        69       528      1320
     70       528  1272
   71     528       1128
  72        552        984
    73     576    840
      74     600      768
        75     600      696,
      76     672  816
      77        672      768
     78       636     696
    79      672      696
```

```
     80        672       648
    81       672    600
   82                672      552
  83            720                  744
    84      720   672
    84      720  600
    86     720       528
     87     768      792
        88      768      696
        89      768            600
            90      768       504
  ELE  CONNE   65     16
1    1    2   7   6
2     2    3     8      7
3     3    4     9     8
4   4    5    10     9
5      6    7      12     11
  6    7    8     13       12
 7     8    9    14       13
  8      9    10     15      14
    9      11     12     17        16
   10          12     13     18      17
  11         13     14     19      18
 12   14    15        20     19
13          16      17     29      28
    14        17      18     30   29
  15     18    19     31        30
 16     19       20   32      31
 17          21   22   34     33
    18          22     23     35     34
   19     23     24       36       35
20              24     25     37     36
   21   25     26     38     37
  22   26     27   39     38
    23      27    28   40     39
  24      28      29   41     40
     25   29   30       42       41
   26   30     31      43      42
27   31   32      44     43
      28          33     34     46     45
      29     34     35     47     46
      30     35     36            48     47
      31      36     37   49      48
         32      37     38      50     49
            33     38     39     51     50
            34     39   40     52   51
   35      40   41     53     52
   36   41         42        54     53
   37        42     43     55       54
   38      43     44     56     55
   39      45     46     58     57
      40      46     47     59      58
         41     47      48     60     59
         42       48   49     61   60
         43       49          50      62       61
    44      50     51     63     62
    45       54     55     65     64
      46      55     56   66      65
          47     57     69     68     67
           48    57     58     70      69
     49      58    59     71     70
   50      59       60          72     71
    51    60     61     73     72
    52       61       62     74     73
      53     62     63      75      74
       54        64   65     86     82
```

```
        55       65    66    90    86
        56    73    74    77      76
    57    74    75    78     77
      58    76      77     83    87
    59     77    78    79    83
      60     79  80    84     83
      61    83    84     88      87
    62    80  81    85     84
      63    84   85    89     88
    64    81    82     86      85
      65     85    86      90      89
      66    92    91     70    71
      67    93    92     71    72
      68    94    93     72    73
      69    95    94     73    76
      70    96    95     76    87
      71    97    96     87    88
      72    98    97     88    89
      73    99    98     89    90
      74   100    99     90    66
      75   101    100    66    56
      76   102    101    56    44
      77   103    102    44    32
      78   104    103    32    20
      79   105    104    20    15
      80   106    105    15    10
      81   107    106    10    5
PRE  DIS  5
 1 X O
   2    X O.
   3    X O.
   4   X O
   5    X O.
   PRE   LOAD       11
1   Y   -1 810
  6    Y   -3.318
 11 Y -2.714
  16      Y    - 2.111
     22    X   + O 0875
  23     X   O 7492
       24      X      1 498
       25      X    1 592
       26      X    1 310
       27      X    2 053
   28     X     1.422    Y -O 905
INT PROP  3000     5 56
 NODE COOR        17
91           528   1271
  92      528    1128
   93           552    984
94       576          840
         95          672     816
     96          768     792
        97       768    696
           98       768    600
             99       768        504
         100  684          486.
     101    600      468
     102    516      450.
     103    432        432.
     104   360      +     432
     105   264      432.
     106    144         432
     107    1        432.
  ELEM CONN       16
82       91         92
```

```
   83      92      93
84      93      94
   85      94      95
      86      95      96
      87      96          97
   88   97        98
 89   98       99
   90.   99    100
     91     100        101
     92     101      102
     93     102    103
     94        103       104
     95     104      105
     96   105      106
     97        106                107
  PRES  DISP  O
     PRES  LOAD    O
  SOI PROP 15    35   O 00007407
NODE COORDINATES 38   ....    64
 108      O     431
109    O      360
     110   O        264
111    O    144
  112   O       O.
113    O      O.
114   144   O
115                264.    O
   116         360      O
117        432      O
    118      516      O
119         600      O
 120   684       O
  121 768     O
    122       864        O
  123   1008         O
   124       1200        O
 125        1440       O
  126       1680           O
  127  1680           O
128                1680        144
    129       1680       264
         130       1680         408
  131       1680        504
    132    1680     600
       133   1680       696
          134    1680       792
         135     1680            984
    136        1680        1128
       137   1680           1272
          138     1680             1272
             139     1440      1272
          140     1200      1272
       141   1008                             1272
          142     864        1272
             143     768            1272
               144      672             1272
                 145       529        1272
            146     144      360
          147     144       264
        148      144       144
   149       264       360
  150       264     264
 151     264           144
   152       360            360
     153       360      264
   151       360      144
```

```
      155     432      360
         156      432      264
             157     432         144
           158    516      360
   159        516     264
160      516     144
   161     600      360
160      600     264
   163       600     144
  164       672     1128
165      672            984
   166      684      360
167      684      264
    168            684         144
   169      768      1128
  170     768       984
 171        768      408
172         768      408
   173             768         144
  174      864      1128
 175     864         984
   176      864      792
  177      864       696
     178      864      600
         179       864          504
            180       864          408
                  181      864     264
        182     864      144
     183     1008   1128
  184     1008      984
 185     1008     792
   186         1008     696
  187     1008     600
  188     1008          504
   189         1008         408
   190      1008        264
 191     1008        144
 192     1200            1128
   193      1200     984
      194       1200       792
         195      1200       696
     196      1200       600
        197    1200      504
   198    1200      408
 199    1200      264
200       1200      144
201          1440   1128
   202      1440        984
 203     1440         792
   204       1440        696
   205         1440      600
   206           1440         504
  207      1440      408
      208       1440        264
          209      1440     144
ELEMENT CONNECTIVITY  36.   88
98       107      108
     99      108       109
100   109       110
  101      110      111
 102   111       112
  103     113      114
   104      114     115
  105           115       116
      106     116      117
107    117       118
```

```
   108        118        119
     109        119         120
110       120        121
      111       121       122
   112        122     123
      113       123        124
 114     124        125
115          125         126
     116       127        128
        117      128        129
   118        129      130
        119      130       131
120       131        132
      121      132       133
     122        133       134
        123      134      135
124   135      136
 125       136       137
   126       138      139
      127       139       140
   128     140       141
      129   141       142
    130   142       143
   131     143        144
 132         144         145
     133    145       91
        134   108      109      146       106
           135      109      110      147      146
              136    110      111       148       147
           137      111      112      114       148
          138   106      146      149      105
              139   146      147       150      149
                140      147      148      151       150
             141      148      114      115       151
        142      105      149      152      104
     143   149      150       153       152
  144     150      151      154       153
     145    151      115      116       154
  146       104     152       155      103
     147    152      153       156      155
        148     153     154       157        156
           149     154       116      117      157
        150       103      155      158       102
   151       155       156      159       158
        152   156      157      160        159
        153            157       117      118        160
     154      102      158      161      101
     155      158      159      162      161
         156   159      160          163       162
           157      160       118      119       163
             158   91       92       164       144
             159      92   93      165       164
             160     93      94   95     165
         161     101    161      166      100
      162      161      162      167      166
    163      162     163       168      167
    164      163     119      120      168
  165     144     164      169      143
166     164     165      170      169
  167         165     95     96      170
    168     100      166      171        99
      169      166      167      172      171
      170      167    168      173      172
      171      168     120      121       173
         172     143      169      174      142
        173      169      170      175      174
```

```
      174   170   96     176   175
    175   96   97   177   176
    176   97   98        178     177
  177   98    99     179    178
 178    99   171   180    179
     179    171   172    181    180
       180   172   173    182     181
     181   173    121   122    182
     182    142    174    183     141
        183    174   175   184    183
     184   175   176   185   184
   185   176   177   186   185
  186    177    178    187     186
 187   178   179   188    187
   188    179   180   189   188
     189    180   181   190    189
     190   181   182   191    190
   191    182   122   123    191
   192   141   183   192    140
     193   183   184   193      192
     194   184   185        194    193
   195   185    186    195   194
  196   186   187   196   195
  197      187   188   197   196
  198  188   189   198   197
  199    189  190   199      198
200   190    191   200    199
   201    191   123   124     200
   202    140   192   201    139
   203     192   193   202   201
  204      193    194   203    202
    205   194      195     204     203
   206   195   196    205   204
     207   196   197   206     205
         208     197   198   207   206
      209 198   199   208   207
      210     199   200   209     208
      211     200   124   125   209
    212   139    201   136   137
      213     201   202    135    136
          214   202   203    134    135
        215    203    204    133     134
      216   204    205        132    133
      217   205   206   131   132
      218    206   207    130      131
      219      207   208   129  130
   220     208          209      128      129
   221   209    125    126    128
PRESCRIBED DISPLACEMENT 25
113   Y O
114 Y O
   115        Y O
  116         Y O
     117      Y O
118  Y O
119  Y O
120  Y O
   121  Y O
122  Y O
 123   Y O
    124  Y O
  125    Y O
126      Y O
127   X O
128   X O
129   X O
130   X O
131   X O
132   X O
133   X O
134   X O
135   X O
136   X O
137   X O
PRES STRESS     1
141   Y  -0 0375
$IBSYS
//
```

# APPENDIX B

# COMPUTER CODINGS AND INPUT LISTINGS OF PROGRAM FINITE

```
C
C
C ===================================================================== C
C                                                                       C
C     THIS PROGRAM IS APPLIED TO EVALUATE THE ELASTIC BEHAVIOR OF       C
C       A U-FRAME STRUCTURE BY THE FINITE ELEMENT MEHTOD ALONE          C
C                                                                       C
C ===================================================================== C
C
C     X, Y  NODE COORDINATES
C     STRE  ELASTICITY MATRIX FOR STRUCTURE
C     SOLE  ELASTICITY MATRIX FOR SOIL
C     ICONNE  ELEMENT INCIDENCE VECTOR
C     A  SYSTEM MATRIX
C     IDCOLU  SPECIFIED BOUNDARY CONDITIONS. 0 -> LOAD. 1 -> DISP
C     COLUMN  SPECIFIED BOUNDARY VALUES
C     BODY  BODY FORCE VECTOR
C     STR. STRESS VECTOR
C     DISP  DISPLACEMENT VECTOR
C     ELSTIF  ELEMENT STIFFNESS MATRIX
C     NNSTR2  2 X NUMBER OF NODES IN STRUCTURE
C     NNINT2  2 X NUMBER OF NODES IN INTERFACE
C     NNSOL2  2 X NUMBER OF NODES IN SOIL
C     NESTR  NUMBER OF STRUCTURE ELEMENTS
C     NEJON  NUMBER OF JOINT ELEMENTS
C     NEINT  NUMBER OF INTERFACE ELEMENTS
C     NESOL  NUMBER OF SOIL ELEMENTS
C     XM, XM1  VECTOR IN STORING SOLUTIONS
C
C-----------------------------
C
C          MAIN PROGRAM
C
C-----------------------------
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION X(250), Y(250), STRE(3,3), ICONNE(250,4), IDCOLU(420)
      DIMENSION COLUMN(420), A(420,420), BODY(420), STR(3,1), XM1(420)
      DIMENSION DISP(2,1),XM(420),ELSTIF(8,8),SOLE(3,3)
C
      COMMON /CO/ X, Y
      COMMON /NNN/ NNSTR2,NNINT2,NNSOL2,NESTR,NEJON,NEINT, NESOL
      COMMON /IO/IRE,IWR
C
C----      SET POINTERS AND INPUT TITLE
C
           IRE = 5
           IWR = 6
      WRITE (IWR,9)
C
C----      CALL SUBROUTINE "INPUT"
C
           IER = 0
      CALL INPUT(ICONNE,IDCOLU,COLUMN,IUNKNO,IER)
      IF (IER  EQ  0) GO TO 1000
           GO TO 10000
C
 1000      DO 5 M = 1, IUNKNO
                BODY(M) = 0
                XM1(M) = 0
                DO 8 N = 1, IUNKNO
                     A(M,N) = 0
    8           CONTINUE
    5      CONTINUE
C
C----      FINITE ELEMENT FORMULATION
C
```

```
      CALL FINITE(STRE,SOLE,ICONNE,A,BODY,NEB,NEL)
C
C----      TAKE BOUNDARY CONDITION AND NODAL FORCE VECTOR INTO ACCOUNT
C
           DO 10 J = 1, IUNKNO
           IF (IDCOLU(J)  EQ  0) GO TO 20
           CALL CLEAN(J,COLUMN(J),A,BODY,IUNKNO)
                BODY(J) = COLUMN(J)
                GO TO 10
   20           BODY(J) = BODY(J) + COLUMN(J)
   10      CONTINUE
C
           IT = 0
           MB = NESTR + 1
           ML = NESTR + NEJON
C
C----      CALCULATE THE INVERSE OF GLOBAL MATRIX
C
      CALL INVER(A,420,IUNKNO)
C
C----      XM  STORE DISPLACEMENTS
C
 2000 CALL PROD1(A,BODY,XM,420,IUNKNO)
C
C----      CHECK THE CONVERGENCE OF DISPLACEMENTS AT INTERFACE
C
      IF (IT  GT  10) GO TO 190
           IB = NNSTR2 + 1
           IL = NNSTR2 + NNINT2
C
           DO 120 K = IB, IL
           IF (DABS(XM(K) - XM1(K))  GT  0 0000001) GO TO 130
  120      CONTINUE
C
           GO TO 190
C
C----      RESTART THE PROLEM. RESTORE "XM" TO "XM1"
C
  130      DO 140 I = 1, IUNKNO
                     XM1(I) = XM(I)
  140      CONTINUE
C
C----      ITERATIVE ROUTINE
C
           IT = IT + 1
C
           CALL ITER(XM,BODY,ICONNE,MB,ML)
C
           GO TO 2000
C
C----      STRUCTURE OUTPUT
C
  190      NNSTR = NNSTR2 / 2
      WRITE (IWR,90)
           DO 570 L = 1, NNSTR
           WRITE (IWR,94) L, XM(2*L-1), XM(2*L)
  570      CONTINUE
      WRITE (IWR,97)
           DO 580 JJ = 1, NESTR
                I = ICONNE(JJ,1)
                J = ICONNE(JJ,2)
                K = ICONNE(JJ,3)
                L = ICONNE(JJ,4)
           CALL SOLVE(I,J,K,L,STRE,STR,DISP,XM)
           WRITE (IWR,904) JJ,(DISP(M,1),M=1,2),(STR(M,1),M=1,3)
  580      CONTINUE
```

```
      WRITE (IWR,99)
          DO 585 JJ = MB, ML
               I = ICONNE(JJ,1)
               J = ICONNE(JJ,2)
               K = ICONNE(JJ,3)
               L = ICONNE(JJ,4)
          CALL FIND(I,J,K,L,STR,DISP,XM)
          WRITE (IWR,906) JJ,I,J,(DISP(M,1),M=1,2),(STR(M,1),M=1,2)
  585     CONTINUE
C
C----     INTERFACE OUTPUT
C
          WRITE (IWR,907)
          M = NNSTR + 1
          N = NNSTR + NNINT2 / 2
C
          DO 590 K = M, N
          WRITE (IWR,94) K,XM(2*K-1),XM(2*K)
  590     CONTINUE
C
C----     SOIL OUTPUT
C
      WRITE (IWR,917)
          M = N + 1
          N = M + NNSOL2 / 2 - 1
C
          DO 600 K = M, N
          WRITE (IWR,94) K,XM(2*K-1),XM(2*K)
  600     CONTINUE
      WRITE (IWR,97)
C
          DO 200 JJ = NEB, NEL
               I = ICONNE(JJ,1)
               J = ICONNE(JJ,2)
               K = ICONNE(JJ,3)
               L = ICONNE(JJ,4)
          CALL SOLVE(I,J,K,L,SOLE,STR,DISP,XM)
          WRITE (IWR,904) JJ,(DISP(M,1),M=1,2),(STR(M,1),M=1,3)
  200     CONTINUE
C
      WRITE(IWR,924)
C
    9 FORMAT(1H1,///,40X,'                 FINITE  ELEMENT  METHOD ',
     &            //,40X,'                             APPLIED  TO',
     &            //,42X,'     SOIL-STRUCTURE  INTERACTION  PROBLEMS')
C
   90 FORMAT(/////,36X,'**********************',
     &           /,36X,'*  STRUCTURE OUTPUT  *',
     &           /,36X,'**********************',
     &      ////,41X,'NODAL DISPLACEMENT ',
     &  //,34X,'NODE',7X,'U',13X,'V')
C
   94 FORMAT(34X,I3,2(3X,E11 4))
C
   97 FORMAT(////,51X,'ELEMENT DISPLACEMENT AND STRESS',//,
     &  34X,'ELEMENT',4X,'U',13X,'V',12X,'SXX',11X,'SYY',10X,'SXY')
C
   99 FORMAT(////,47X,'RELATIVE DISPLACEMENT AND LOCAL STRESS ',
     &      'AT JOINT ELEMENT',//,34X,'ELEMENT',5X,'LOCAL X AXIS',
     &      7X,'TAN  DISP',5X,'NOR  DISP',8X,'TAU',10X,'SIGMA')
C
  904 FORMAT(34X,I3,5(3X,E11 4))
C
  906 FORMAT(34X,I3,6X,'NODE',I3,' TO NODE',I3,4(3X,E11 4))
C
  907 FORMAT(/////,36X,'**********************',
     &           /,36X,'*  INTERFACE OUTPUT  *',
     &           /,36X,'**********************',
     &      ////,41X,'NODAL DISPLACEMENT ',
     &  //,34X,'NODE',7X,'U',13X,'V')
C
  917 FORMAT(/////,36X,'******************',
     &           /,36X,'*  SOIL  OUTPUT  *',
     &           /,36X,'******************',
     &      ////,41X,'NODAL DISPLACEMENT ',
     &  //,34X,'NODE',7X,'U',13X,'V')
C
  924 FORMAT(1H1)
C
10000 STOP
      END
C-----------------------------
C
C          SUBPROGRAM "INVER"
C
C-----------------------------
C
      SUBROUTINE INVER(A,NX,N)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(NX,NX)
C
            DO 100 K = 1, N
C
                  DO 20 J = 1, N
                  IF (J  EQ  K) GO TO 20
                        A(K,J) = A(K,J) / A(K,K)
   20             CONTINUE
C
                  A(K,K) = 1  / A(K,K)
C
                  DO 30 I = 1, N
                  IF(I  EQ  K) GO TO 30
C
                        DO 40 J = 1, N
                        IF (J .EQ  K) GO TO 40
                              A(I,J) = A(I,J) - A(K,J) * A(I,K)
   40                   CONTINUE
   30             CONTINUE
C
                  DO 50 I = 1, N
                  IF (I  EQ  K) GO TO 50
                        A(I,K) = -A(I,K) * A(K,K)
   50             CONTINUE
  100       CONTINUE
C
      RETURN
      END
C-----------------------------
C
C          SUBPROGRAM "ITER"
C
C-----------------------------
C
      SUBROUTINE ITER(XM,BODY,ICONNE,MB,ML)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION XM(420), BODY(420), ELSTIF(8,8), X(250)
      DIMENSION Y(250), ICONNE(250,4)
C
      COMMON /CO/ X,Y
      COMMON /PROP/ESTR PSTR WSTR EKN EKS ESOL PSOL WSOL
```

```
C
          DO 10 JJ = MB, ML
               I = ICONNE(JJ,1)
               J = ICONNE(JJ,2)
               K = ICONNE(JJ,3)
               L = ICONNE(JJ,4)
               XLENG = DSQRT((X(J) - X(I)) ** 2 + (Y(J) - Y(I)) ** 2)
               COST = (X(J) - X(I)) / XLENG
               SINT = (Y(J) - Y(I)) / XLENG
C
C----     RELATIVE DISPLACEMENT AT CENTROID OF JOINT ELEMENT
C
               U= 0 5 * (XM(2*K-1) + XM(2*L-1) - XM(2*I-1) - XM(2*J-1))
               V = 0 5 * (XM(2*K) + XM(2*L) - XM(2*I) - XM(2*J))
C
C----     TRANSFORM TO LOCAL DISPLACEMENT
C
               WS = U * COST + V * SINT
               WN = - U * SINT + V * COST
               TAU = WS * EKS
               SIGMA = WN * EKN
C
          IF (SIGMA  LE  0) GO TO 10
C
C----     RESTRAINING FORCE IS - SIGMA * LENGTH * ANGLE OF TRANSFORM
C
               TX = (COST * TAU - SIGMA * SINT) * XLENG / 2
               TY = (SINT * TAU + COST * SIGMA) * XLENG / 2
               BODY(2*I-1) = BODY(2*I-1) + TX
               BODY(2*I) = BODY(2*I) + TY
               BODY(2*J-1) = BODY(2*J-1) + TX
               BODY(2*J) = BODY(2*J) + TY
C
               BODY(2*L-1) = BODY(2*L-1) - TX
               BODY(2*L) = BODY(2*L) - TY
               BODY(2*K-1) = BODY(2*K-1) -TX
               BODY(2*K) = BODY(2*K) - TY
C
   10     CONTINUE
C
      RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "FIND"
C
C-----------------------------
C
      SUBROUTINE FIND(I,J,K,L,STR,DISP,XM)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION STR(3,1), DISP(2,1), XM(420), D(2), X(250), Y(250)
C
      COMMON /CO/ X,Y
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
C
          XD = X(J) - X(I)
          YD = Y(J) - Y(I)
          D(1) = (-XM(2*I-1) - XM(2*J-1) + XM(2*K-1) + XM(2*L-1)) * 0 5
          D(2) = (- XM(2*I) - XM(2*J) + XM(2*K) + XM(2*L)) * 0 5
C
          XLENG = DSQRT(XD * XD + YD * YD)
          COST = XD / XLENG
          SINT = YD / XLENG
C
C----     TRANSFORM DISPLACEMENT TO LOCAL COORDINATES AND FIND STRESS
```

```
C
          DISP(1,1) = COST * D(1) + SINT * D(2)
          DISP(2,1) = - SINT * D(1) + COST * D(2)
          STR(1,1) = EKS * DISP(1,1)
          STR(2,1) = EKN * DISP(2,1)
 1000 RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "FINITE"
C
C-----------------------------
C
      SUBROUTINE FINITE(STRE,SOLE,ICONNE,A,BODY,NEB,NEL)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ELSTIF(8,8), A(420,420), STRE(3,3)
      DIMENSION ICONNE(250,4), BODY(420), SOLE(3,3), DETJAC(4)
C
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
      COMMON /NNN/NNSTR2,NNINT2,NNSOL2,NESTR,NEJON,NEINT, NESOL
C
          CONST = ESTR / ((1  + PSTR) * (1  - 2  * PSTR))
          STRE(1,1) = CONST * (1. - PSTR)
          STRE(2,2) = STRE(1,1)
          STRE(3,3) = CONST * .5 * (1  - 2  * PSTR)
          STRE(1,2) = CONST * PSTR
          STRE(2,1) = STRE(1,2)
          STRE(2,3) = 0
          STRE(3,2) = STRE(2,3)
          STRE(1,3) = 0
          STRE(3,1) = 0
          N = NESTR + NEJON
C
          DO 100 JJ = 1, N
               I = ICONNE(JJ,1)
               J = ICONNE(JJ,2)
               K = ICONNE(JJ,3)
               L = ICONNE(JJ,4)
C
          IF (JJ  GT. NESTR) GO TO 50
C
          CALL STIFF(I,J,K,L,STRE,ELSTIF,DETJAC)
C
               BODY(2*I) = BODY(2*I) - WSTR * DETJAC(1)
               BODY(2*J) = BODY(2*J) - WSTR * DETJAC(2)
               BODY(2*K) = BODY(2*K) - WSTR * DETJAC(3)
               BODY(2*L) = BODY(2*L) - WSTR * DETJAC(4)
               GO TO 70
C
   50     CALL SJOINT(I,J,K,L,EKS,EKN,ELSTIF)
C
   70     CALL ASSEM(ELSTIF,I,J,K,L,A)
C
  100     CONTINUE
C
          CONST1 = ESOL / ((1  + PSOL) * (1  - 2  * PSOL))
          SOLE(1,1) = CONST1 * (1  - PSOL)
          SOLE(2,2) = SOLE(1,1)
          SOLE(3,3) = CONST1 * .5 * (1  - 2  * PSOL)
          SOLE(1,2) = CONST1 * PSOL
          SOLE(2,1) = SOLE(1,2)
          SOLE(2,3) = 0
          SOLE(3,2) = SOLE(2,3)
          SOLE(1,3) = 0
          SOLE(3,1) = 0
```

```
          NEB = N + NEINT + 1
          NEL = NEB + NESOL - 1
C
          DO 200 JJ = NEB, NEL
              I = ICONNE(JJ,1)
              J = ICONNE(JJ,2)
              K = ICONNE(JJ,3)
              L = ICONNE(JJ,4)
C
          CALL STIFF(I,J,K,L,SOLE,ELSTIF,DETJAC)
C
          CALL ASSEM(ELSTIF,I,J,K,L,A)
C
              BODY(2*I) = BODY(2*I) - WSOL * DETJAC(1)
              BODY(2*J) = BODY(2*J) - WSOL * DETJAC(2)
              BODY(2*K) = BODY(2*K) - WSOL * DETJAC(3)
              BODY(2*L) = BODY(2*L) - WSOL * DETJAC(4)
  200     CONTINUE
      RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "SJOINT"
C
C-----------------------------
C
      SUBROUTINE SJOINT(I,J,K,L,ES,EN,ELSTIF)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION SQ4(8,8), ELSTIF(8,8), TRAN(8,8), TEMP(8,8), X(250)
      DIMENSION Y(250)
C
      COMMON /CO/ X, Y
C
C----     CLEAR TRANSFORMATION MATRIX AND LOCAL JOINT STIFFNESS MATRIX
C
          DO 10 M = 1, 8
              DO 20 N = 1, 8
                  TRAN(M,N) = 0
                  SQ4(M,N) = 0
   20         CONTINUE
   10     CONTINUE
C
          XD = X(J) - X(I)
          YD = Y(J) - Y(I)
          XLENG = DSQRT(XD * XD + YD * YD)
C
C----     FORMULATION OF TRANSPOSE "TRAN"
C
          TRAN(1,1) = XD / XLENG
          TRAN(2,2) = XD / XLENG
          TRAN(3,3) = XD / XLENG
          TRAN(4,4) = XD / XLENG
          TRAN(5,5) = XD / XLENG
          TRAN(6,6) = XD / XLENG
          TRAN(7,7) = XD / XLENG
          TRAN(8,8) = XD / XLENG
          TRAN(1,2) = - YD / XLENG
          TRAN(2,1) = YD / XLENG
          TRAN(3,4) = - YD / XLENG
          TRAN(4,3) = YD / XLENG
          TRAN(5,6) = - YD / XLENG
          TRAN(6,5) = YD / XLENG
          TRAN(7,8) = - YD / XLENG
          TRAN(8,7) = YD / XLENG
C
```

```
C----     FORMULATION OF LOACL JOINT ELEMENT STIFFNESS
C
          SQ4(1,1) = XLENG / 3  * ES
          SQ4(1,3) = XLENG / 6  * ES
          SQ4(1,5) = - XLENG / 6  * ES
          SQ4(1,7) = - XLENG / 3  * ES
C
          SQ4(3,1) = XLENG / 6  * ES
          SQ4(3,3) = XLENG / 3  * ES
          SQ4(3,5) = - XLENG / 3  * ES
          SQ4(3,7) = - XLENG / 6  * ES
C
          SQ4(5,1) = - XLENG / 6  * ES
          SQ4(5,3) = - XLENG / 3  * ES
          SQ4(5,5) = XLENG / 3  * ES
          SQ4(5,7) = XLENG / 6  * ES
C
          SQ4(7,1) = - XLENG / 3  * ES
          SQ4(7,3) = - XLENG / 6  * ES
          SQ4(7,5) = XLENG / 6  * ES
          SQ4(7,7) = XLENG / 3  * ES
C
          SQ4(2,2) = XLENG / 3  * EN
          SQ4(2,4) = XLENG / 6  * EN
          SQ4(2,6) = - XLENG / 6  * EN
          SQ4(2,8) = - XLENG / 3  * EN
C
          SQ4(4,2) = XLENG / 6  * EN
          SQ4(4,4) = XLENG / 3. * EN
          SQ4(4,6) = - XLENG / 3  * EN
          SQ4(4,8) = - XLENG / 6  * EN
C
          SQ4(6,2) = - XLENG / 6  * EN
          SQ4(6,4) = - XLENG / 3  * EN
          SQ4(6,6) = XLENG / 3. * EN
          SQ4(6,8) = XLENG / 6. * EN
C
          SQ4(8,2) = - XLENG / 3  * EN
          SQ4(8,4) = - XLENG / 6. * EN
          SQ4(8,6) = XLENG / 6  * EN
          SQ4(8,8) = XLENG / 3  * EN
C
C----     TRANSFORMATION TO GLOBAL JOINT STIFFNESS MATRIX
C
      CALL PROD(TRAN,SQ4,TEMP,8,8,8)
C
          TRAN(1,2) = - TRAN(1,2)
          TRAN(2,1) = - TRAN(2,1)
          TRAN(3,4) = - TRAN(3,4)
          TRAN(4,3) = - TRAN(4,3)
          TRAN(5,6) = - TRAN(5,6)
          TRAN(6,5) = - TRAN(6,5)
          TRAN(7,8) = - TRAN(7,8)
          TRAN(8,7) = - TRAN(8,7)
C
      CALL PROD(TEMP,TRAN,ELSTIF,8,8,8)
C
      RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "STIFF"
C
C-----------------------------
C
      SUBROUTINE STIFF(I,J,K,L,E,ELSTIF,DETJAC)
```

```
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION E(3,3), SQ4(8,8), B(3,8), DUMM(8,3), DETJAC(4)
      DIMENSION XJINV(2,2), XM1(2,8), XM2(2,8), ELSTIF(8,8)
      DIMENSION AALPHA(4), ABETA(4), BT(8,3), X(250), Y(250)
C
      COMMON /CO/ X, Y
      COMMON /ALPH/ AALPHA, ABETA
C
C----      IGNORE WEIGHT, BECAUSE WEIGHT = 1 FOR 4 POINT INTEGRATION
C
           XI = X(I)
           XJ = X(J)
           XK = X(K)
           XL = X(L)
C
           YI = Y(I)
           YJ = Y(J)
           YK = Y(K)
           YL = Y(L)
C
           DO 10 IR = 1, 8
                DO 20 IC = 1, 8
                     ELSTIF(IR,IC) = 0
   20           CONTINUE
   10      CONTINUE
C
           IPOINT = 1
   15      ALPHA = AALPHA(IPOINT)
           BETA  = ABETA(IPOINT)
C
           DXDA = -  25 * (1 - BETA) * XI +  25 * (1 - BETA) * XJ
     &            +  25 * (1 + BETA) * XK -  25 * (1 + BETA) * XL
C
           DYDA = -  25 * (1 - BETA) * YI +  25 * (1 - BETA) * YJ
     &            +  25 * (1 + BETA) * YK -  25 * (1 + BETA) * YL
C
           DXDB = -  25 * (1 - ALPHA) * XI -  25 * (1 + ALPHA) * XJ
     &            +  25 * (1 + ALPHA) * XK +  25 * (1 - ALPHA) * XL
C
           DYDB = -  25 * (1 - ALPHA) * YI -  25 * (1 + ALPHA) * YJ
     &            +  25 * (1 + ALPHA) * YK +  25 * (1 - ALPHA) * YL
C
C----      DXDA, DYDA, DXDB AND DYDB ARE TERMS IN JACOBIAN
C
           XJDET = DXDA * DYDB - DXDB * DYDA
           XJINV(1,1) = DYDB / XJDET
           XJINV(1,2) = - DYDA / XJDET
           XJINV(2,1) = - DXDB / XJDET
           XJINV(2,2) = DXDA / XJDET
           DETJAC(IPOINT) = DABS(XJDET)
C
           DO 30 M = 1, 4
                XM1(1,2*M) = 0
                XM1(2,2*M) = 0
                XM2(1,2*M-1) = 0
                XM2(2,2*M-1) = 0
   30      CONTINUE
C
           XM1(1,1) = -  25 * (1 - BETA)
           XM2(1,2) = XM1(1,1)
           XM1(1,3) = +  25 * (1 - BETA)
           XM2(1,4) = XM1(1,3)
           XM1(1,5) = +  25 * (1 + BETA)
           XM2(1,6) = XM1(1,5)
           XM1(1,7) = -  25 * (1 + BETA)
```

```
           XM2(1,8) = XM1(1,7)
C
           XM1(2,1) = -  25 * (1 - ALPHA)
           XM2(2,2) = XM1(2,1)
           XM1(2,3) = -  25 * (1 + ALPHA)
           XM2(2,4) = XM1(2,3)
           XM1(2,5) = +  25 * (1 + ALPHA)
           XM2(2,6) = XM1(2,5)
           XM1(2,7) = +  25 * (1 - ALPHA)
           XM2(2,8) = XM1(2,7)
C
           DO  40 M = 1, 8
                B(1,M) = 0
                B(2,M) = 0
                B(3,M) = 0
C
                DO 50 N = 1, 2
                     B(1,M) = B(1,M) + XJINV(1,N) * XM1(N,M)
                     B(2,M) = B(2,M) + XJINV(2,N) * XM2(N,M)
                     B(3,M) = B(3,M) + XJINV(2,N) * XM1(N,M)
     &                               + XJINV(1,N) * XM2(N,M)
   50           CONTINUE
C
                BT(M,1) = B(1,M)
                BT(M,2) = B(2,M)
                BT(M,3) = B(3,M)
   40      CONTINUE
C
      CALL PROD(BT,E,DUMM,8,3,3)
C
      CALL PROD(DUMM,B,SQ4,8,3,8)
C
           DO 60 M = 1, 8
                DO 70 N = 1, 8
                     ELSTIF(M,N) = ELSTIF(M,N)+SQ4(M,N) * DETJAC(IPOINT)
   70           CONTINUE
   60      CONTINUE
C
      IF (IPOINT  EQ  4 ) GO TO 1000
C
           IPOINT = IPOINT + 1
           GO TO 15
C
 1000 RETURN
      END
C-----------------------------
C
C          SUBPROGRAM "ASSEM"
C
C-----------------------------
C
      SUBROUTINE ASSEM(ELSTIF,I,J,K,L,A)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ELSTIF(8,8), JR(4), NR(8), A(420,420)
C
           JR(1) = I
           JR(2) = J
           JR(3) = K
           JR(4) = L
C
           DO 10 M = 1, 4
                IX = 2 * M - 1
                IY = IX + 1
                NR(IX) = 2 * JR(M) -1
                NR(IY) = 2 * JR(M)
```

```
   10    CONTINUE
C
         DO 20 M = 1, 8
              DO 30 N = 1,8
                   A(NR(M),NR(N)) = A(NR(M),NR(N)) + ELSTIF(M,N)
   30         CONTINUE
   20    CONTINUE
C
      RETURN
      END
C-----------------------------
C
C        SUBPROGRAM "SOLVE"
C
C-----------------------------
C
      SUBROUTINE SOLVE(I,J,K,L,E,STR,DISP,BODY)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION E(3,3), STR(3,1), STRAIN(3,1), BODY(420)
      DIMENSION B(3,8), DISP(2,1), XJINV(2,2), XM1(2,8), XM2(2,8)
      DIMENSION X(250), Y(250), DD(8,1)
C
      COMMON /CO/ X, Y
C
         XI = X(I)
         XJ = X(J)
         XK = X(K)
         XL = X(L)
C
         YI = Y(I)
         YJ = Y(J)
         YK = Y(K)
         YL = Y(L)
C
         DD(1,1) = BODY(2*I-1)
         DD(2,1) = BODY(2*I)
         DD(3,1) = BODY(2*J-1)
         DD(4,1) = BODY(2*J)
         DD(5,1) = BODY(2*K-1)
         DD(6,1) = BODY(2*K)
         DD(7,1) = BODY(2*L-1)
         DD(8,1) = BODY(2*L)
C
         DISP(1,1) = 0
         DISP(2,1) = 0
C
         DO 15 K = 1, 4
              DISP(1,1) = DISP(1,1) +  25 * DD(2*K-1,1)
              DISP(2,1) = DISP(2,1) +  25 * DD(2*K,1)
   15    CONTINUE
C
         DXDA = -  25 * XI +  25 * XJ +  25 * XK -  25 * XL
C
         DYDA = -  25 * YI +  25 * YJ +  25 * YK -  25 * YL
C
         DXDB = -  25 * XI -  25 * XJ +  25 * XK +  25 * XL
C
         DYDB = -  25 * YI -  25 * YJ + .25 * YK +  25 * YL
C
C----    DXDA, DYDA, DXDB AND DYDB ARE TERMS IN JACOBIAN
C
         XJDET = DXDA * DYDB - DXDB * DYDA
         XJINV(1,1) = DYDB / XJDET
         XJINV(1,2) = - DYDA / XJDET
         XJINV(2,1) = - DXDB / XJDET
```

```
         XJINV(2,2) = DXDA / XJDET
C
         DO 30 M = 1, 4
              XM1(1,2*M) = 0
              XM1(2,2*M) = 0.
              XM2(1,2*M-1) = 0
              XM2(2,2*M-1) = 0
   30    CONTINUE
C
         XM1(1,1) = -  25
         XM2(1,2) = XM1(1,1)
         XM1(1,3) = +  25
         XM2(1,4) = XM1(1,3)
         XM1(1,5) = +  25
         XM2(1,6) = XM1(1,5)
         XM1(1,7) = -  25
         XM2(1,8) = XM1(1,7)
C
         XM1(2,1) = -  25
         XM2(2,2) = XM1(2,1)
         XM1(2,3) = -  25
         XM2(2,4) = XM1(2,3)
         XM1(2,5) = +  25
         XM2(2,6) = XM1(2,5)
         XM1(2,7) = +  25
         XM2(2,8) = XM1(2,7)
C
         DO  40 M = 1, 8
              B(1,M) = 0
              B(2,M) = 0
              B(3,M) = 0
C
              DO 50 N = 1, 2
                   B(1,M) = B(1,M) + XJINV(1,N) * XM1(N,M)
                   B(2,M) = B(2,M) + XJINV(2,N) * XM2(N,M)
                   B(3,M) = B(3,M) + XJINV(2,N) * XM1(N,M)
     &                             + XJINV(1,N) * XM2(N,M)
   50         CONTINUE
C
   40    CONTINUE
C
      CALL PROD(B,DD,STRAIN,3,8,1)
C
      CALL PROD(E,STRAIN,STR,3,3,1)
C
      RETURN
      END
C-----------------------------
C
C        SUBPROGRAM "CLEAN"
C
C-----------------------------
C
      SUBROUTINE CLEAN(J,V,A,BODY,IUNKNO)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(420,420),BODY(420)
C
         DO 10 K = 1, IUNKNO
              BODY(K) = BODY(K) - A(K,J) * V
              A(J,K) = 0
              A(K,J) = 0
   10    CONTINUE
C
         A(J,J) = 1
      RETURN
```

```
      END
C-----------------------------
C
C           SUBPROGRAM "INPUT"
C
C-----------------------------
C
      SUBROUTINE INPUT(ICONNE.IDCOLU.COLUMN.IUNKNO.IER)
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 CARD(80), WORD(6), TIT(3), STR(3), SOI(3), PRO(3)
      CHARACTER*1 INTE(3), NODCOO(6), PRELOA(6), PREDIS(6), ELECON(6)
C
      DIMENSION VAR(5), ICONNE(250,4), IDCOLU(420)
      DIMENSION COLUMN(420), X(250), Y(250), AALPHA(4), ABETA(4)
C
      COMMON /CO/ X, Y
      COMMON /PROP/ESTR,PSTR,WSTR,EKN,EKS,ESOL,PSOL,WSOL
      COMMON /NNN/ NNSTR2,NNINT2,NNSOL2,NESTR,NEJON,NEINT, NESOL
      COMMON /IO/IRE,IWR
      COMMON /ALPH/ AALPHA, ABETA
C
      DATA TIT/1HT,1HI,1HT/, STR/1HS,1HT,1HR/, SOI/1HS,1HO,1HI/
      DATA PRO/1HP,1HR,1HO/, INTE/1HI,1HN,1HT/
      DATA NODCOO/1HN,1HO,1HD,1HC,1HO,1HO/
      DATA PREDIS/1HP,1HR,1HE,1HD,1HI,1HS/
      DATA PRELOA/1HP,1HR,1HE,1HL,1HO,1HA/
      DATA ELECON/1HE,1HL,1HE,1HC,1HO,1HN/
C
C----         INITIALIZATION OF GAUSS INTEGRATING POINTS GI AND WEIGHTS OME
C
              AALPHA(1) = - 577350269189626
              AALPHA(2) = + 577350269189626
              AALPHA(3) = + 577350269189626
              AALPHA(4) = - 577350269189626
C
              ABETA(1)  = - 577350269189626
              ABETA(2)  = - 577350269189626
              ABETA(3)  = + 577350269189626
              ABETA(4)  = + 577350269189626
C
              IUNKNO = 0
              NETO = 0
C
C----         STRUCTURE  ICHECK = -1
C----         INTERFACE  ICHECK =  0
C----         SOIL       ICHECK =  1
C
              ICHECK = -1
C
C----         ID  IDENTIFIER
C----         NW  NUMBER OF WORDS, 3 CHARACTERS PER WORD
C----         NV  NUMBER OF VARIABLES
C----         SET WORDS FOR TITLE CARD
C
              ID = 1
              NW = 1
              NV = 1
C
   17         DO 10 J = 1, 3
                   WORD(J) = TIT(J)
   10         CONTINUE
C
C----         READ TITLE CARD
C
      READ (IRE 4) (CARD(K)  K= 1 80)
```

```
      CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
      IF (IER  EQ  0) GO TO 20
C
      WRITE (IWR, 9)
            GO TO 10000
C
   20 WRITE (IWR, 93)
            JJ = VAR(1)
C
C----        READ INPUT CARDS OF TITLE
C
             DO 30 J = 1, JJ
             READ (IRE,4) (CARD(K), K=1,80)
             WRITE (IWR,96) (CARD(K), K=1,80)
   30        CONTINUE
C
C----        SET WORDS FOR PROPERTY CARDS
C
 5000        ID = 1
             NW = 2
      IF (ICHECK) 100, 200, 300
  100        DO 110 J = 1, 3
                   WORD(J) = STR(J)
  110        CONTINUE
             NV = 3
             GO TO 400
C
  200        DO 210 J = 1, 3
                   WORD(J) = INTE(J)
  210        CONTINUE
             NV = 2
             GO TO 400
C
  300        DO 310 J = 1, 3
                   WORD(J) = SOI(J)
  310        CONTINUE
             NV = 3
C
  400        DO 410 J = 1, 3
                   WORD(J+3) = PRO(J)
  410        CONTINUE
C
C----        READ PROPERTY CARD
C
      READ (IRE,4) (CARD(K), K= 1,80)
            NW = 2
      CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
      IF (IER  EQ  0) GO TO 415
C
      WRITE (IWR, 99)
            GO TO 10000
C
  415 IF (ICHECK) 420, 430, 440
  420       ESTR = VAR(1)
            PSTR = VAR(2)
            WSTR = VAR(3)
      WRITE (IWR,903) ESTR, PSTR, WSTR
            GO TO 450
C
  430       EKN  = VAR(1)
            EKS  = VAR(2)
      WRITE (IWR,906) EKN, EKS
            GO TO 450
C
  440       ESOL = VAR(1)
            PSOL = VAR(2)
```

```
            WSOL = VAR(3)
      WRITE (IWR,909) ESOL, PSOL, WSOL
C
C----       SET WORDS FOR COORDINATE CARDS
C
  450       DO 460 J = 1, 6
                  WORD(J) = NODCOO(J)
  460       CONTINUE
C
C----       READ COORDINATE CARD
C
      READ (IRE,4) (CARD(K), K= 1,80)
            NV = 1
      CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
      IF (IER  EQ  0) GO TO 480
C
  470 WRITE (IWR, 913)
            GO TO 10000
C
  480       NN = VAR(1)
      IF (ICHECK) 490, 500, 510
C
C----       LOADS ARE ASSUMED TO BE 0 AT THE BEGINNING
C----       IDCOLU = 0 LOADS ARE PRESCRIBED
C----       IDCOLU = 1 DISPLACEMENTS ARE PRESCRIBED
C
  490       NNSTR2 = 2 * NN
            IUNKNO = IUNKNO + NNSTR2
      WRITE (IWR,916) NN
            LL = 1
            GO TO 520
C
  500       NNINT2 = 2 * NN
            IUNKNO = IUNKNO + NNINT2
      WRITE (IWR,919) NN
            LL = IUNKNO - NNINT2 + 1
            GO TO 520
C
  510       NNSOL2 = 2 * NN
            IUNKNO = IUNKNO + NNSOL2
      WRITE (IWR,923) NN
            LL = IUNKNO - NNSOL2 + 1
C
  520       DO 525 J = LL, IUNKNO
                  IDCOLU(J) = 0
                  COLUMN(J) = 0
  525       CONTINUE
C
      WRITE (IWR,926)
            ID = 2
            NV = 3
C
C----       READ INPUT CARDS OF COORDINATE
C
            DO 530 J = 1,NN
            READ (IRE,4) (CARD(K),K=1,80)
            CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
            IF (IER  NE  0) GO TO 470
C
                  KK = VAR(1)
                  X(KK) = VAR(2)
                  Y(KK) = VAR(3)
            WRITE (IWR,929) KK, X(KK), Y(KK)
  530       CONTINUE
C
C----       SET WORDS FOR CONNECTIVITY CARDS
```

```
C
            DO 540 J = 1, 6
                  WORD(J) = ELECON(J)
  540       CONTINUE
C
C----       READ CONNECTIVITY CARD
C
      READ (IRE,4) (CARD(K), K= 1,80)
            ID =1
            NW = 2
            NV = 1
      IF (ICHECK  NE  - 1) GO TO 542
            NV = 2
  542 CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
      IF (IER  EQ  0) GO TO 545
C
  550 WRITE (IWR, 933)
            GO TO 10000
C
C----       READ INPUT CARDS OF CONNECTIVITY
C
  545 IF (ICHECK) 560, 580, 600
C
  560       KK = VAR(1)
            NESTR = KK
            NEJON = VAR(2)
            NETO = NETO + NESTR + NEJON
      WRITE (IWR,936) KK, NEJON
            NN = KK + NEJON
            ID = 3
  562       NV = 5
C
            DO 565 J = 1, NN
            READ (IRE,4) (CARD(K),K=1,80)
            CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
            IF (IER  NE  0) GO TO 550
C
                  M = VAR(1)
C
                  DO 570 K = 1, 4
                        ICONNE(M,K) = VAR(K+1)
  570             CONTINUE
C           IF (ICHECK  EQ  1) GO TO 575
C
            IF (J  LE  KK) GO TO 575
C
            WRITE (IWR,937) M, (ICONNE(M,K), K=1,4)
                  GO TO 565
  575       WRITE (IWR,938) M, (ICONNE(M,K), K=1,4)
  565       CONTINUE
C
            GO TO 620
C
  580       NM = VAR(1)
            NEINT = NM
            NETO = NETO + NEINT
      WRITE (IWR,939) NM
            ID = 4
  582       NV = 3
C
            DO 585 J = 1, NM
            READ (IRE,4) (CARD(K),K=1,80)
            CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
            IF (IER  NE  0) GO TO 550
C
            M = VAR(1)
```

```
                DO 590 K = 1, 2
                     ICONNE(M,K) = VAR(K+1)
  590           CONTINUE
C
           WRITE (IWR,941) M, (ICONNE(M,K), K=1,2)
  585      CONTINUE
C
           GO TO 620
C
  600      NN = VAR(1)
           NESOL = NN
           NETO = NETO + NESOL
      WRITE (IWR,943) NN
           ID =3
           GO TO 562
C
C----      SET WORDS FOR PRESCRIBED DISPLACEMENTS
C
  620      DO 640 J = 1, 6
                WORD(J) = PREDIS(J)
  640      CONTINUE
C
C----      READ PRESCRIBED DISPLACEMENT CARD
C
      READ (IRE,4) (CARD(K), K= 1,80)
           ID = 1
           NW = 2
           NV = 1
      CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
      IF (IER  EQ  O) GO TO 645
C
  650 WRITE (IWR,946)
           GO TO 10000
C
  645      JJ = VAR(1)
      IF (ICHECK) 655, 658, 660
  655 WRITE (IWR,949) JJ
           GO TO 670
C
  658 WRITE (IWR,951) JJ
           GO TO 670
C
  660 WRITE (IWR,953) JJ
  670 IF (JJ  EQ  O) GO TO 800
           ID = 2
C
C----      READ INPUT CARDS OF PRESCRIBED DISPLACEMENTS
C
           DO 700 K = 1, JJ
           READ (IRE,4) (CARD(J), J=1, 80)
           CALL CONV2(IER,CARD,WORD,VAR,ID)
           IF (IER  NE  O) GO TO 650
C
                KK = VAR(1)
                IP = 1
                MM = 2 * KK - 1
  710      IF (WORD(IP)  EQ  ' ') GO TO 740
C
           IF (WORD(IP)  NE  'X') GO TO 720
C
                IDCOLU(MM) = 1
                COLUMN(MM) = VAR(IP+1)
                GO TO 730
C
  720           IDCOLU(MM+1) = 1
                COLUMN(MM+1) = VAR(IP+1)
```

```
  730           IP = IP + 1
                GO TO 710
C
  740      IF (IP  EQ  3) GO TO 760
C
           WRITE (IWR,956) KK, WORD(1), VAR(2)
                GO TO 700
C
  760      WRITE (IWR,959) KK, WORD(1), VAR(2), WORD(2), VAR(3)
  700      CONTINUE
C
C----      SET WORD FOR PRESCRIBED NON-ZERO LOADS
C
  800      DO 840 J = 1, 6
                WORD(J) = PRELOA(J)
  840      CONTINUE
C
C
C----      READ PRESCRIBED LOAD CARDS
C
      READ (IRE,4) (CARD(K), K= 1,80)
           ID = 1
           NW = 2
           NV = 1
      CALL CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
      IF (IER .EQ  O) GO TO 845
C
  855 WRITE (IWR,963)
           GO TO 10000
C
  845      JJ = VAR(1)
      IF (ICHECK) 860, 865, 870
C
  860 WRITE (IWR,966) JJ
           GO TO 880
C
  865 WRITE (IWR,968) JJ
           GO TO 880
C
  870 WRITE (IWR,969) JJ
  880 IF (JJ  EQ O) GO TO 1200
           ID = 2
C
C----      READ INPUT CARDS OF PRESCRIBED LOADS
C
           DO 900 K = 1, JJ
           READ (IRE,4) (CARD(J), J=1, 80)
           CALL CONV2(IER,CARD,WORD,VAR,ID)
           IF (IER  NE. O) GO TO 855
C
                KK = VAR(1)
                IP = 1
                MM = 2 * KK - 1
  910      IF (WORD(IP)  EQ  ' ') GO TO 940
C
           IF (WORD(IP)  NE  'X') GO TO 920
C
                COLUMN(MM) = VAR(IP+1)
                GO TO 930
C
  920           COLUMN(MM+1) = VAR(IP+1)
  930           IP = IP + 1
                GO TO 910
C
  940      IF (ICHECK) 925, 1200, 935
C
```

```
  925       IF (IP  EQ  3) GO TO 960
C
            WRITE (IWR,973) KK, WORD(1), VAR(2)
                  GO TO 900
C
  960       WRITE (IWR,976) KK, WORD(1), VAR(2), WORD(2), VAR(3)
                  GO TO 900
C
  935       IF (IP  EQ  3) GO TO 965
C
            WRITE (IWR,979) KK, WORD(1), VAR(2)
                  GO TO 900
C
  965       WRITE (IWR,983) KK, WORD(1), VAR(2), WORD(2), VAR(3)
  900       CONTINUE
C
 1200 IF (ICHECK) 1300, 1400, 10000
C
 1300       ICHECK = 0
            GO TO 5000
C
 1400       ICHECK = 1
            GO TO 5000
C
    4 FORMAT(80A1)
C
    9 FORMAT(///,26X,'******   TITLE CARDS ARE LOST OR WRONG    ******')
C
   93 FORMAT(/////,26X,'PROBLEM IDENTIFICATION ',
     &             /,26X,'------------------------',/)
C
   96 FORMAT(36X,80A1)
C
   99 FORMAT(//,26X,'******  PROPERTY CARDS ARE LOST OR WRONG  ******',
     &          /,26X,'                  OR                       ',
     &          /,26X,'**** NUMBER OF TITLE CARDS IS INCONSISTENT *****',
     &          /,26X,'                  OR                       ',
     &          /,26X,'*NUMBER OF STRUCTURE LOAD CARDS IS INCONSISTENT*',
     &          /,26X,'                  OR                       ',
     &          /,26X,'****   NUMBER OF INTERFACE CONNECTIVITY    *****',
     &          /,26X,'*****        CARDS IS INCONSISTENT         *****')
C
  903 FORMAT(///,26X,'STRUCTURE INPUT ',
     &           /,26X,'-----------------',
     &          //,36X,'ELASTIC MODULUS = ',E11 4,'   KSI',
     &           /,36X,'POISSON RATIO   = ',E11 4,
     &           /,36X,'UNIT SELFWEIGHT = ',E11 4,'   KCI')
C
  906 FORMAT(///,26X,'INTERFACE INPUT·',
     &           /,26X,'-----------------',
     &          //,36X,'UNIT NORMAL STIFFNESS         = ',E11 4,'   KPI',
     &           /,36X,'UNIT TANGENTIAL STIFFNESS     = ',E11 4,'   KPI')
C
  909 FORMAT(///,26X,'SOIL INPUT ',
     &           /,26X,'------------',
     &          //,36X,'ELASTIC MODULUS = ', E11 4,'   KSI',
     &           /,36X,'POISSON RATIO   = ', E11 4,
     &           /,36X,'UNIT SELFWEIGHT = ', E11 4,'   KCI')
C
  913 FORMAT(//,26X,'****** COORDINATE CARDS ARE LOST OR WRONG ******')
C
  916 FORMAT(//,36X,'NUMBER OF NODES IN STRUCTURE = ',I3)
C
  919 FORMAT(//,36X,'NUMBER OF NODES IN INTERFACE = ',I3)
C
  923 FORMAT(// 36X,'NUMBER OF NODES IN SOIL       = ' I3)
```

```
C
  926 FORMAT(/,36X,'NODE',10X,'X COORDINATE',10X,'Y COORDINATE')
C
  929 FORMAT(36X,I3,11X,E11 4,11X,E11 4)
C
  933 FORMAT(//,26X,'***** CONNECTIVITY CARDS ARE LOST OR WRONG *****',
     &          /,26X,'                  OR                       ',
     &          /,26X,'** NUMBER OF COORDINATE CARDS IS INCONSISTENT **')
C
  936 FORMAT(//,36X,'NUMBER OF ELEMENTS IN STRUCTURE = ',I3,
     &          /,36X,'NUMBER OF JOINT ELEMENTS        = ',I3,
     &     //,36X,'ELEMENT NO',4X,'NODE 1',6X,'NODE 2',6X,'NODE 3',
     &          6X,'NODE 4')
C
  937 FORMAT(39X,I3,4(9X,I3),'    JOINT ELEMENT')
C
  938 FORMAT(39X,I3,4(9X,I3))
C
  939 FORMAT(//,36X,'NUMBER OF ELEMENTS IN INTERFACE = ',I3,
     &     //,36X,'ELEMENT NO',4X,'NODE 1',6X,'NODE 2')
C
  941 FORMAT(39X,I3,2(9X,I3))
C
  943 FORMAT(//,36X,'NUMBER OF ELEMENTS IN SOIL      = ',I3,
     &     //,36X,'ELEMENT NO',4X,'NODE 1',6X,'NODE 2',6X,'NODE 3',
     &          6X,'NODE 4')
C
  946 FORMAT(//,26X,'***** DISPLACEMENT CARDS ARE LOST OR WRONG *****',
     &          /,26X,'                  OR                       ',
     &          /,26X,'* NUMBER OF CONNECTIVITY CARDS IS INCONSISTENT *')
C
  949 FORMAT(//,36X,'NUMBER OF PRESCRIBED DISPLACEMENT',
     &                ' IN STRUCTURE = ',I3,/)
C
  951 FORMAT(//,36X,'NUMBER OF PRESCRIBED DISPLACEMENT',
     &                ' IN INTERFACE = ',I3,/)
C
  953 FORMAT(//,36X,'NUMBER OF PRESCRIBED DISPLACEMENT',
     &                ' IN SOIL = ',I3,/)
C
  956 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,'INCH')
C
  959 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,A1,' = ',E11 4,
     &         '    INCH')
C
  963 FORMAT(//,26X,'**** LOAD OR STRESS CARDS ARE LOST OR WRONG ****',
     &          /,26X,'                  OR                       ',
     &          /,26X,'* NUMBER OF DISPLACEMENT CARDS IS INCONSISTENT *')
C
  966 FORMAT(//,36X,'NUMBER OF PRESCRIBED NON-ZERO LOAD IN STRUCTURE',
     &                 ' = ',I3,/)
C
  968 FORMAT(//,36X,'NUMBER OF PRESCRIBED NON-ZERO LOAD IN INTERFACE',
     &                 ' = ',I3,/)
C
  969 FORMAT(//,36X,'NUMBER OF PRESCRIBED NON-ZERO LOAD IN SOIL',
     &                 ' = ',I3,/)
C
  973 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,'KIPS')
C
  976 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,A1,' = ',E11 4,
     &         '    KIPS')
C
  979 FORMAT(36X,'NODE ',I3,8X,A1,' = ',E11 4,8X,'KIPS')
C
  983 FORMAT(36X 'NODE ',I3,8X A1 ' = ',F11 4 8X A1 ' = ' F11 4
```

```
     &                KIPS')
C
10000 RETURN
      END
C-----------------------------
C
C          SUBPROGRAM "PROD1"
C
C-----------------------------
C
      SUBROUTINE PROD1(A,B,C,M,IN)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(M,M), B(M), C(M)
C
           DO 10 I = 1, IN
                C(I) = O
                DO 20 K = 1, IN
                     C(I) = C(I) + A(I,K) * B(K)
   20           CONTINUE
   10      CONTINUE
C
      RETURN
      END
C-----------------------------
C
C          SUBPROGRAM "CONV1"
C
C-----------------------------
C
      SUBROUTINE CONV1(IER,NW,NV,CARD,WORD,VAR,ID)
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 CARD(80), WORD(6)
      DIMENSION VAR(5)
C
C----      FOR MAJOR INPUT CARD (START WITH CHARACTER), ID = 1
C----      FOR NODE COORDINATE, ID = 2
C----      FOR STRUCTURE CONNECTIVITY, ID = 3
C----      FOR SOIL CONNECTIVITY, ID = 4
C
           J = 1
      IF (ID  NE 1) GO TO 65
C
C----      FIND CHARACTER
C
           DO 100 K = 1, NW
   20      IF (J  GE  77) GO TO 500
C
           IF ((CARD(J)  NE  ' ')  AND  (CARD(J)  NE  ',')) GO TO 30
C
                J = J + 1
                GO TO 20
C
   30      IF ((CARD(J)  NE  WORD(3*K-2))  OR  (CARD(J+1)  NE
     &     WORD(3*K-1))  OR  (CARD(J+2)  NE  WORD(3*K))) GO TO 500
C
                J = J + 3
   60      IF (J  GE  80) GO TO 500
C
           IF ((CARD(J)  EQ  ' ')  OR  (CARD(J)  EQ  ',')) GO TO 100
C
                J = J + 1
                GO TO 60
C
  100      CONTINUE
```

```
C
C----      FIND VARIABLE
C
   65      DO 200 K = 1, NV
                VAR(K) = O
   70      IF (J GE 80) GO TO 500
C
           IF ((CARD(J)  NE  ' ')  AND  (CARD(J)  NE  ',')) GO TO 80
C
                J = J + 1
                GO TO 70
C
   80      CALL VALUE(IER,CARD,VAR(K),ID,J)
           IF (VAR(1)  LT  O ) GO TO 500
C
  200      CONTINUE
C
           GO TO 1000
C
  500      IER = 1
 1000 RETURN
      END
C-----------------------------
C
C          SUBPROGRAM "CONV2"
C
C-----------------------------
C
      SUBROUTINE CONV2(IER,CARD,WORD,VAR,ID)
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 CARD(80), WORD(6)
      DIMENSION VAR(5)
C
C----      INITIALIZATION
C
           WORD(1) = ' '
           WORD(2) = ' '
           VAR(1) = O
           VAR(2) = O
           VAR(3) = O
           J = 1
           IC = 1
   10 IF (J  GE  80) GO TO 500
C
      IF ((CARD(J)  NE  ' ')  AND  (CARD(J)  NE  ',')) GO TO 100
C
           J = J + 1
           GO TO 10
C
  100 CALL VALUE(IER,CARD,VAR(IC),ID,J)
      IF ((IER  NE  O)  OR  (VAR(1)  LT  1 )) GO TO 500
C
   20 IF ((J  GE  80) .AND  (IC  NE  1)) GO TO  900
C
      IF ((J  GE  80)  AND  (IC  EQ  1)) GO TO 500
C
      IF ((CARD(J) .NE  ' ')  AND. (CARD(J) .NE  ',')) GO TO 200
C
           J = J + 1
           GO TO 20
C
  200 IF ((CARD(J)  NE  'X')  AND  (CARD(J)  NE  'Y')) GO TO 500
C
           WORD(IC) = CARD(J)
           IC = IC + 1
```

```
          J = J + 1
          GO TO 10
C
  900 IF (WORD(1)  EQ  WORD(2)) GO TO 500
          WORD(3) = ' '
          GO TO 1000
  500     IER = 1
 1000 RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "VALUE"
C
C-----------------------------
C
      SUBROUTINE VALUE(IER,CARD,VAL,ID,J)
C
      IMPLICIT REAL*8(A-H,O-Z)
      CHARACTER*1 CARD(80)
      DIMENSION IDIGIT(16)
C
C----     ND  NUMBER  OF DIGIT
C----     IP  LOCATION OF " "
C----     INITIALIZATION
C
          SIGN = 1 0
          ND = 1
          IP = 0
          VAL = 0
      IF ((CARD(J)  NE  '-')  AND  (CARD(J)  NE  '+')) GO TO 10
C
      IF (CARD(J)  EQ  '+') GO TO 15
C
      IF (ID  NE  2) GO TO 500
C
          SIGN = -1
   15     J = J + 1
      IF (J  GT  80) GO TO 500
      IF (CARD(J)  EQ  ' ') GO TO 15
   10     KK = J
C
          DO 1000 K = KK, 80
          IF (CARD(K)  EQ. '0') GO TO 20
C
          IF (CARD(K)  EQ  '1') GO TO 30
C
          IF (CARD(K)  EQ  '2') GO TO 40
C
          IF (CARD(K)  EQ  '3') GO TO 50
C
          IF (CARD(K)  EQ  '4') GO TO 60
C
          IF (CARD(K)  EQ  '5') GO TO 70
C
          IF (CARD(K)  EQ  '6') GO TO 80
C
          IF (CARD(K)  EQ  '7') GO TO 90
C
          IF (CARD(K)  EQ  '8') GO TO 100
C
          IF (CARD(K)  EQ  '9') GO TO 110
C
          IF (CARD(K)  EQ  ' ') GO TO 120
C
          IF ((CARD(K)  NE  ' ')  AND  (CARD(K)  NE  ',')) GO TO 500
C
```

```
                J = K + 1
                GO TO 125
C
   20           IDIGIT(ND) = 0
                GO TO 115
C
   30           IDIGIT(ND) = 1
                GO TO 115
C
   40           IDIGIT(ND) = 2
                GO TO 115
C
   50           IDIGIT(ND) = 3
                GO TO 115
C
   60           IDIGIT(ND) = 4
                GO TO 115
C
   70           IDIGIT(ND) = 5
                GO TO 115
C
   80           IDIGIT(ND) = 6
                GO TO 115
C
   90           IDIGIT(ND) = 7
                GO TO 115
C
  100           IDIGIT(ND) = 8
                GO TO 115
C
  110           IDIGIT(ND) = 9
  115           ND = ND + 1
                GO TO 1000
C
  120           IP = ND
 1000     CONTINUE
C
  125 IF (IP  NE  0) GO TO 130
C
          IP = ND
C
  130     NM = ND - 1
C
          DO 200 K = 1, NM
              VAL = VAL + SIGN * IDIGIT(K) * 10  ** (IP-K-1)
  200     CONTINUE
C
          GO TO 2000
C
  500     IER = 1
 2000 RETURN
      END
C-----------------------------
C
C         SUBPROGRAM "PROD"
C
C-----------------------------
C
      SUBROUTINE PROD(A,B,C,M,L,N)
C
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION A(M,L), B(L,N), C(M,N)
C
          DO 10 I = 1, M
              DO 20 J = 1, N
                  C(I,J) = 0
```

```
                     DO 30 K = 1, L
                          C(I,J) = C(I,J) + A(I,K) * B(K,J)
   30                CONTINUE
   20           CONTINUE
   10      CONTINUE
C
      RETURN
      END
$ENTRY
  TITLE 4
THIS PROBLEM IS TO TEST A U-FRAME STRUCTURE BY THE FINITE ELEMENT METHOD
ALONE  IT INCLUDES 204 NODES AND 185 ELEMENTS   THE RESULTS ARE COMPARED
WITH THOSE OBTAINED FROM THE COUPLING OF BOUNDARY AND FINITE ELEMENT
METHODS
    STR PROP 3000  25 0 00008681
NODE COOR 90
 1 0  576
  2 0,  540
3  0          504
  4  0   468
 5   0   432
 6 144  576
  7 144,  540
8  144          504
  9  144    468.
 10   144  432
11 264  576
 12 264,  540
13 264          504
  14 264    468
 15   264  432
16 360  576
 17 360,  540
18 360          504
  19 360    468
 20   360  432
    21     432    1392
22    432    1272
 23            432    +1128
  24    + 432,,,     984
    25    432   840
  26   432   768
    27    432    696
   28     432      576
  29    432   540
30      432   504
31     432    468
  32   432,,   432
33     456     1392
34    456    1272
  35   456   1128
 36      462    984
   37   468      840
 38     468    768
39      480     696
 40   , 480                         624
 41    480      576
42      480      528
 43     504       480
 44      516,    450
   45     480     1392
  46    480,, 1272
 47   480   1128
48      492,,,   984
  49   504   840
  50      504  768
```

```
    51        528     696
    51       528   648
    53       528              600
      54      528    552
        55     576      504
        56     600   468
        57     504     1392
      58     504   1272
        59     504    1128
      60     522  984
      61      540     840
     62      540    768
    63    564   696
    64   600   552
    65       648     504
      66     684     486
      67          552        1392
        68      552,   1344
          69     528     1320
       70     528  1272
     71    528      1128
    72     552      984
      73    576    840
        74    600     768
          75    600    696,
        76    672  816
        77     672     768
      78     636   696
     79    672     696
     80      672     648
    81     672   600
   82             672      552
  83          720                744
    84    720   672
    84   720  600
    86   720     528
     87   768    792
       88     768    696
       89     768         600
          90     768      504
  ELE CONNE  65   16
 1  1   2  7  6
 1   2   3    8    7
 1   3   4   9   8
 1  4   5   10   9
 1    6   7    12   11
   6   7   8   13      12
  7   8   9   14     13
   8    9   10   15     14
     9   11   12   17     16
    10       12   13   18   17
   11      13   14   19   18
  12  14  15     20   19
13      16   17   29   28
    14     17   18   30  29
  15   18  19   31     30
 16   19    20  32   31
 17      21  22  34   33
    18      22   23   35   34
   19   23   24    36    35
 20          24   25   37   36
   21  25   26   38   37
  22  26   27  39   38
    23   27  28  40   39
  24   28   29  41   40
     25  29  30    42    41
```

```
   26  30   31    43    42
 27  31  32   44   43
      28        33   34   46   45
      29   34   35   47   46
      30   35   36         48   47
      31    36   37  49    48
        32   37   38    50    49
          33   38   39   51   50
          34   39  40   52  51
  35   40  41   53   52
  36  41     42     54   53
  37     42   43   55    54
  38    43   44   56   55
  39    45   46   58   57
      40    46   47   59    58
        41   47    48   60   59
        42    48  49   61  60
        43    49      50    62     61
   44   50   51   63   62
   45    54   55   65   64
     46   55    56  66   65
        47   57   69   68   67
          48  57   58   70    69
     49   58  59   71   70
  50   59    60       72   71
   51  60   61   73   72
   52    61    62   74   73
     53   62   63    75    74
      54     64  65   86   82
      55     65   66   90   86
      56   73   74   77    76
  57   74   75   78    77
    58   76    77    83   87
  59    77   78   79   83
    60    79  80   84    83
    61   83   84    88    87
  62   80  81   85    84
    63   84  85   89    88
  64   81   82   86     85
    65    85   86     90     89
    66   92   91    70   71
    67   93   92    71   72
    68   94   93    72   73
    69   95   94    73   76
    70   96   95    76   87
    71   97   96    87   88
    72   98   97    88   89
    73   99   98    89   90
    74  100   99    90   66
    75  101   100   66   56
    76  102   101   56   44
    77  103   102   44   32
    78  104   103   32   20
    79  105   104   20   15
    80  106   105   15   10
    81  107   106   10   5
PRE  DIS  5
 1 X O
   2   X O
   3   X O
   4  X O
   5   X O
   PRE  LOAD      11
1  Y  -1 810
  6   Y  -3 318
 11 Y -2 714
```

```
    16     Y    - 2 111
       22   X  + O 0875
   23     X  O.7492
          24    X     1 498
          25    X  1.592
          26    X  1.310
          27    X  2 053
     28    X    1 422   Y -O.905
 INT PROP  3000     5 56
  NODE COOR      17
91          528  1272
   92     528   1128
    93          552   984
94     576        840
        95         672   .816
     96        768    792
         97     768   696
              98     768   600
                99    768      504
           100  684         486
      101   600     468.
      102   516     450.
      103   432      432
      104  360.   +    432.
      105  264    432.
      106   144        432
      107   O      432
   ELEM CONN     16
82     91       92
   83     92    93
84    93    94
   85     94    95
      86    95    96
      87    96      97
   88  97      98
 89  98     99
   90   99   100
     91   100     101
     92   101    102
     92   102   103
     94     103    104
     95   104    105
     96  105    106
     97     106          107
   PRES  DISP  1
   107   X  O.
      PRES  LOAD   O
   SOI PROP 15   35  O.00007407
NODE COORDINATES 97
   108   O    360
109   O    264
      110  O    144
111  O    O
   112  144    360
113   144    264
114  144    144.
115              144.   O
   116       264    360
117      264    264
    118     264   144
119     264      O
 120  360     360
  121 360      264
    122     360   144.
  123  360    O
   124     432   360
```

```
 125     432    264
  126      432          144
  127   432     0
128           516         360
    129   516     264
  130   516    144
   131     516            0
 132     600        360
  133         600     264
  134   600      144
135           600      0
    136          672        1272
  137     672         1128
   138     672    984
 139     684        360
  140      684          264
  141   684        144
142            684      0
    143    768         1272
  144   768      1128
   145     768         984
 146     768        408
  147      768           264
  148   768     144
149             768                   0
    150     864        1272
  151     864    1128
  151     864    984
  153     864     792
  154     864     696
  155     864    600
  156     864        504
  157     864    408
  158     864      264
  159     864    144
  160     864    0
    161     1008       1272
  162    1008    1128
  163    1008    984
  164    1008     792
  165    1008     696
  166    1008    600
  167    1008        504
  168    1008    408
  169    1008      264
  170    1008    144
  171    1008    0
    172     1200      1272
  173    1200    1128
  174         1200    984
  175    1200      792
  176     1200         696
  177    1200    600
  178    1200        504
  179    1200    408
  180    1200      264
  181    1200    144
  182    1200    0
    183     1440       1272
  184    1440    1128
  185         1440    984
  186    1440      792
  187     1440         696
  188    1440    600
  189    1440        504
  190    1440    408
```

```
  191    1440      264
  192    1440    144
  193    1440    0
    194     1680     1272
  195    1680    1128
  195         1680    984
  197    1680     792
  198     1680         696
  199    1680    600
  200    1680        504
  201    1680    408
  202    1680      264
  203    1680    144
  204    1680    0
ELEMENT CONNECTIVITY        88
98       107    108    112    106
    99    108    109   113    112
100  109   110   114    113
  101 110   111    115     114
 102  106    112    116     105
  103     112    113       117     116
   104   113     114       118      117
  105    114     115     119    118
      106 105     116    120     104
107  116    117    121    120
  108 117    118    122    121
     109    118    119    123     122
110    104    120      124      103
      111    120    121     125    124
  112     121    122    126     125
      113    122    123    127     126
 114   103      124    128    102
115    124    125     129     128
     116   125    126    130,    129
          117     126    127    131      130
          118      102    128     132     101
          119    128    129     133     132
          120       129    130    134     133
          121    130     131     135    134
          122   91     92     137    136
  123   92     93     138    137
  124   93     94     95        138
  125   101     132      139     100
  126   132     133     140     139
  127    133    134   141          140
  128   134    135     142     141
        129   136    137    144    143
        130   137   138    145   144
        131    138     95     96     145
        132   100    139      146     99
        133    139      140      147     146
        134   140    141    148   147
        135    141   142   149   148
      136   143   144    151    150
  137    144   145    152    151
 138    145     96    153        152
        139      96    97     154      153
      140   97    98     155     154
     141  98    99     156    155
 142       99    146    157    156
 143     146    147    158     157
     144    147    148     159     158
     145     148     149      160       159
146     150      151     162     161
147     151      152     163     162
148     152      153     164     163
```

```
149    153     154    165    164
150    154     155    166    165
151    155     156    167    166
152    156     157    168    167
153    157     158    169    168
154    158     159    170    169
155    159     160    171    170
   156       161    162    173    172
   157       162    163    174    173
   158       163    164    175    174
   159       164    165    176    175
   160       165    166    177    176
   161       166    167    178    177
   162       167    168    179    178
   163       168    169    180    179
   164       169    170    181    180
   165       170    171    182    181
     166     172      173      184      183
     167     173      174      185      184
     168     174      175      186      185
     169     175      176      187      186
     170     176      177      188      187
     171     177      178      189      188
     172     178      179      190      189
     173     179      180      191      190
     174     180      181      192      191
     175     181      182      193      192
176    183      184    195   194
177    184      185    196   195
178    185      186    197   196
179    186      187    198   197
180    187      188    199   198
181    188      189    200   199
182    189      190    201   200
183    190      191    202   201
184    191      192    203   202
185    192      193    204   203
PRESCRIBED DISPLACEMENT 27
108    X O
 109   X   O
   110      X  O.
  111    X O  Y O
     115      Y O
119  Y O
123  Y O
127  Y O
   131  Y O
135  Y O
 142   Y O
    149  Y O
  160    Y O
171      Y O
182    Y O
193    Y O
194    X O
195    X O
196    X O
197    X O
198    X O
199    X O
200    X O
201    X O
202    X O
203    X O
204    X O      Y  O
PRES LOAD       3
```

```
150    Y  -0 9
161    Y   -4 2
172  Y    -1 2
$IBSYS
//
```

# APPENDIX C

# RESULTS OF LOAD TRANSFER TESTS

```
                         FINITE  ELEMENT  METHOD

                               APPLIED  TO

                    SOIL-STRUCTURE  INTERACTION  PROBLEMS



PROBLEM IDENTIFICATION
----------------------

          THIS PROBLEM IS TO TEST THE INTERFACE BY THE FINITE ELEMENT METHOD
          ALONE  THE NORMAL AND TANGENTIAL STRESSES ARE PRINTED FOR EACH ITERATION


STRUCTURE INPUT
---------------

          ELASTIC MODULUS =  0 3000D 04    KSI
          POISSON RATIO   =  0 2500D 00
          UNIT SELFWEIGHT =  0 8681D-04    KCI


          NUMBER OF NODES IN STRUCTURE =   8

          NODE          X COORDINATE          Y COORDINATE
            1            0 0000D 00            0 1000D 03
            2            0 0000D 00            0 5000D 02
            3            0 5000D 02            0 1500D 03
            4            0 5000D 02            0 1000D 03
            5            0 5000D 02            0 5000D 02
            6            0 1000D 03            0 1500D 03
            7            0 1000D 03            0 1000D 03
            8            0 1000D 03            0 5000D 02


          NUMBER OF ELEMENTS IN STRUCTURE =   3
          NUMBER OF JOINT ELEMENTS        =   4

          ELEMENT NO    NODE 1      NODE 2      NODE 3      NODE 4
               1           1           2           5           4
               2           3           4           7           6
               3           4           5           8           7
               4          10           9           6           7    JOINT ELEMENT
               5          11          10           7           8    JOINT ELEMENT
               6          12          11           8           5    JOINT ELEMENT
               7          13          12           5           2    JOINT ELEMENT


          NUMBER OF PRESCRIBED DISPLACEMENT IN STRUCTURE =   2

          NODE   1        X =  0 0000D 00         INCH
          NODE   2        X =  0 0000D 00         INCH
```

```
          NUMBER OF PRESCRIBED NON-ZERO LOAD IN STRUCTURE =   1

          NODE   3        X = -0 1500D 02         KIPS


INTERFACE INPUT·
----------------

          UNIT NORMAL STIFFNESS          =  0 3000D 04   KPI
          UNIT TANGENTIAL STIFFNESS      =  0 1500D 02   KPI


          NUMBER OF NODES IN INTERFACE =   5

          NODE          X COORDINATE          Y COORDINATE
            9            0 1000D 03            0 1500D 03
           10            0 1000D 03            0 1000D 03
           11            0 1000D 03            0 5000D 02
           12            0 5000D 02            0 5000D 02
           13            0 0000D 00            0 5000D 02


          NUMBER OF ELEMENTS IN INTERFACE =   4

          ELEMENT NO    NODE 1      NODE 2
               8           9          10
               9          10          11
              10          11          12
              11          12          13


          NUMBER OF PRESCRIBED DISPLACEMENT IN INTERFACE =   1

          NODE  13        X =  0 0000D 00         INCH


          NUMBER OF PRESCRIBED NON-ZERO LOAD IN INTERFACE =   0



SOIL INPUT
----------

          ELASTIC MODULUS =  0 1500D 02   KSI
          POISSON RATIO   =  0 3500D 00
          UNIT SELFWEIGHT =  0 7407D-04   KCI


          NUMBER OF NODES IN SOIL          =  11

          NODE          X COORDINATE          Y COORDINATE
           14            0 0000D 00            0 0000D 00
           15            0 5000D 02            0 0000D 00
           16            0 1000D 03            0 0000D 00
           17            0 1500D 03            0 1500D 03
           18            0 1500D 03            0 1000D 03
           19            0 1500D 03            0 5000D 02
```

```
 20            0  1500D  03            0  0000D  00
 21            0  2000D  03            0  1500D  03
 22            0  2000D  03            0  1000D  03
 23            0  2000D  03            0  5000D  02
 24            0  2000D  03            0  0000D  00

NUMBER OF ELEMENTS IN SOIL      =   8

ELEMENT NO    NODE 1      NODE 2      NODE 3      NODE 4
    12          13          14          15          12
    13          12          15          16          11
    14           9          10          18          17
    15          10          11          19          18
    16          11          16          20          19
    17          17          18          22          21
    18          18          19          23          22
    19          19          20          24          23

NUMBER OF PRESCRIBED DISPLACEMENT IN SOIL =   8

NODE  14        X =  0 0000D 00        Y =  0 0000D 00    INCH
NODE  15        Y =  0 0000D 00        INCH
NODE  16        Y =  0 0000D 00        INCH
NODE  20        Y =  0 0000D 00        INCH
NODE  21        X =  0 0000D 00        INCH
NODE  22        X =  0 0000D 00        INCH
NODE  23        X =  0 0000D 00        INCH
NODE  24        X =  0 0000D 00        Y =  0 0000D 00    INCH

NUMBER OF PRESCRIBED NON-ZERO LOAD IN SOIL =   3

NODE   9        Y = -0 1000D 01        KIPS
NODE  17        Y = -0 6000D 01        KIPS
NODE  21        Y = -0 5000D 01        KIPS


    ITERATION =   0

    ELEMENT =  4  SIGMA =  0 016047320  TAU =  0 056051976

    ELEMENT =  5  SIGMA = -0 117704953  TAU =  0 017042793

    ELEMENT =  6  SIGMA = -0 025031025  TAU =  0 004648125

    ELEMENT =  7  SIGMA = -0 061085243  TAU =  0 004266631


    ITERATION =   1

    ELEMENT =  4  SIGMA =  0 002262905  TAU =  0 008678506

    ELEMENT =  5  SIGMA = -0 119664106  TAU =  0 004750739
```

```
ELEMENT =  6  SIGMA = -0 020380114  TAU =  0 004624049

ELEMENT =  7  SIGMA = -0 062122606  TAU =  0 004253621


ITERATION =   2

ELEMENT =  4  SIGMA =  0 000322616  TAU =  0 001343685

ELEMENT =  5  SIGMA = -0 119939620  TAU =  0 002847566

ELEMENT =  6  SIGMA = -0 019660013  TAU =  0 004620428

ELEMENT =  7  SIGMA = -0 062283220  TAU =  0 004251575


ITERATION =   3

ELEMENT =  4  SIGMA =  0 000046477  TAU =  0 000208041

ELEMENT =  5  SIGMA = -0 119978796  TAU =  0 002552900

ELEMENT =  6  SIGMA = -0 019548520  TAU =  0 004619881

ELEMENT =  7  SIGMA = -0 062308087  TAU =  0 004251255


ITERATION =   4

ELEMENT =  4  SIGMA =  0 000006761  TAU =  0 000032211

ELEMENT =  5  SIGMA = -0 119984426  TAU =  0 002507277

ELEMENT =  6  SIGMA = -0 019531257  TAU =  0 004619798

ELEMENT =  7  SIGMA = -0 062311937  TAU =  0 004251204


ITERATION =   5

ELEMENT =  4  SIGMA =  0 000000992  TAU =  0 000004987

ELEMENT =  5  SIGMA = -0 119985243  TAU =  0 002500213

ELEMENT =  6  SIGMA = -0 019528585  TAU =  0 004619785

ELEMENT =  7  SIGMA = -0 062312533  TAU =  0 004251197


ITERATION =   6
```

```
ELEMENT =  4  SIGMA =  0 000000147  TAU =  0 000000772

ELEMENT =  5  SIGMA = -0 119985363  TAU =  0 002499120

ELEMENT =  6  SIGMA = -0 019528171  TAU =  0 004619783

ELEMENT =  7  SIGMA = -0 062312626  TAU =  0 004251195


ITERATION =   7

ELEMENT =  4  SIGMA =  0 000000022  TAU =  0 000000120

ELEMENT =  5  SIGMA = -0 119985380  TAU =  0 002498950

ELEMENT =  6  SIGMA = -0 019528107  TAU =  0 004619783

ELEMENT =  7  SIGMA = -0 062312640  TAU =  0 004251195


ITERATION =   8




**********************
*  STRUCTURE OUTPUT  *
**********************


      NODAL DISPLACEMENT

NODE        U              V
  1     0 0000D 00    -0 1251D 00
  2     0 0000D 00    -0 1284D 00
  3    -0 1137D 00    -0 1176D 00
  4    -0 2579D-01    -0 1093D 00
  5     0 1439D-01    -0 1062D 00
  6    -0 1038D 00    -0 4282D-01
  7    -0 3442D-01    -0 4972D-01
  8     0 1833D-01    -0 5597D-01


                    ELEMENT DISPLACEMENT AND STRESS

ELEMENT     U               V             SXX            SYY            SXY
  1     -0 2850D-02    -0 1172D 00    -0 4088D 00    -0 1319D 00    -0 2659D-01
  2     -0 6943D-01    -0 7986D-01     0 2850D-01    -0 3453D-01    -0 2752D 00
  3     -0 6871D-02    -0 8029D-01    -0 1315D 00     0 5525D-01     0 2023D 00
```

```
               RELATIVE DISPLACEMENT AND LOCAL STRESS AT JOINT ELEMENT

ELEMENT      LOCAL X AXIS          TAN  DISP      NOR  DISP        TAU          SIGMA
   4      NODE 10 TO NODE  9     0 1234D-08     0 1094D-11    0 1851D-07    0 3282D-08
   5      NODE 11 TO NODE 10     0 1666D-03    -0 4000D-04    0 2499D-02   -0 1200D 00
   6      NODE 12 TO NODE 11     0 3080D-03    -0 6509D-05    0 4620D-02   -0 1953D-01
   7      NODE 13 TO NODE 12     0 2834D-03    -0 2077D-04    0 4251D-02   -0 6231D-01



   *********************
   *  INTERFACE OUTPUT *
   *********************


      NODAL DISPLACEMENT

NODE        U              V
  9    -0 1038D 00    -0 4250D-01
 10    -0 3445D-01    -0 5004D-01
 11     0 1829D-01    -0 5599D-01
 12     0 1382D-01    -0 1061D 00
 13     0 0000D 00    -0 1283D 00



   ******************
   *  SOIL  OUTPUT  *
   ******************


      NODAL DISPLACEMENT

NODE        U              V
 14     0 0000D 00     0 0000D 00
 15    -0 2768D-02     0 0000D 00
 16    -0 5920D-01     0 0000D 00
 17     0 2025D-01    -0 6728D 00
 18    -0 1146D 00    -0 4128D 00
 19    -0 5556D-01    -0 2056D 00
 20    -0 5696D-01     0 0000D 00
 21     0 0000D 00    -0 1080D 01
 22     0 0000D 00    -0 6405D 00
 23     0 0000D 00    -0 2796D 00
 24     0 0000D 00     0 0000D 00


                    ELEMENT DISPLACEMENT AND STRESS

ELEMENT     U               V             SXX            SYY            SXY
```

```
12     0 2763D-02    -0 5862D-01    -0 2774D-01    -0 5502D-01     0 2155D-02
13    -0 7466D-02    -0 4053D-01    -0 3353D-01    -0 4577D-01     0 8013D-02
14    -0 5816D-01    -0 2945D 00    -0 2217D-01    -0 5508D-01    -0 5153D-01
15    -0 4659D-01    -0 1811D 00    -0 6318D-01    -0 6843D-01    -0 3468D-01
16    -0 3836D-01    -0 6539D-01    -0 5114D-01    -0 7225D-01    -0 3927D-02
17    -0 2360D-01    -0 7017D 00    -0 6801D-01    -0 1563D 00    -0 2781D-01
18    -0 4255D-01    -0 3846D 00    -0 3268D-01    -0.1147D 00    -0 2004D-01
19    -0 2813D-01    -0 1213D 00    -0 3580D-01    -0 1022D 00    -0 4035D-02
```

# APPENDIX D

# SELECTED PRINTOUT SHEETS FOR PROGRAM BOUFIN

```
              COUPLING  OF  FINITE  AND  BOUNDARY  ELEMENT  METHODS

                                  APPLIED  TO

                     SOIL-STRUCTURE  INTERACTION  PROBLEMS



                ********  SYMMETRICAL  ABOUT  Y  AXIS  ********




PROBLEM IDENTIFICATION
----------------------

          THIS PROBLEM IS TO TEST A U-FRAME STRUCTURE BY THE COUPLING OF BOUNDARY
          AND FINITE ELEMENT METHODS.  IT INCLUDES 209 NODES AND 221 ELEMENTS.
          THE RESULTS ARE COMPARED WITH THOSE OBTAINED FROM THE FINITE ELEMENT
          METHOD ALONE   NOTE THAT TWO SMALL ELEMENTS ARE USED AT EACH END OF
          INTERFACE TO AVOID DISPLACEMENT DISCONTINUITIES



STRUCTURE INPUT.
----------------

          ELASTIC MODULUS =  O 3000D 04   KSI
          POISSON RATIO   =  O 2500D 00
          UNIT SELFWEIGHT =  O 8681D-04   KCI


          NUMBER OF NODES IN STRUCTURE =  90

          NODE          X COORDINATE          Y COORDINATE
            1            O 0000D 00            O 5760D 03
            2            O 0000D 00            O 5400D 03
            3            O 0000D 00            O 5040D 03
            4            O 0000D 00            O 4680D 03
            5            O 0000D 00            O 4320D 03
            6            O 1440D 03            O 5760D 03
            7            O 1440D 03            O 5400D 03
            8            O 1440D 03            O.5040D 03
            9            O 1440D 03            O 4680D 03
           10            O 1440D 03            O 4320D 03
           11            O 2640D 03            O 5760D 03
           12            O 2640D 03            O 5400D 03
           13            O 2640D 03            O 5040D 03
           14            O 2640D 03            O 4680D 03
           15            O 2640D 03            O 4320D 03
           16            O 3600D 03            O 5760D 03
           17            O 3600D 03            O 5400D 03
           18            O 3600D 03            O 5040D 03
           19            O 3600D 03            O 4680D 03
           20            O 3600D 03            O 4320D 03
           21            O 4320D 03            O 1392D 04
           22            O 4320D 03            O 1272D 04
           23            O 4320D 03            O 1128D 04
           24            O 4320D 03            O 9840D 03
           25            O 4320D 03            O 8400D 03
           26            O 4320D 03            O 7680D 03
           27            O 4320D 03            O 6960D 03
           28            O 4320D 03            O 5760D 03
           29            O 4320D 03            O 5400D 03
           30            O 4320D 03            O.5040D 03
           31            O 4320D 03            O 4680D 03
           32            O 4320D 03            O 4320D 03
           33            O 4560D 03            O 1392D 04
           34            O 4560D 03            O 1272D 04
           35            O 4560D 03            O.1128D 04
           36            O 4620D 03            O 9840D 03
           37            O 4680D 03            O 8400D 03
           38            O 4680D 03            O 7680D 03
           39            O 4800D 03            O.6960D 03
           40            O 4800D 03            O 6240D 03
           41            O 4800D 03            O 5760D 03
           42            O 4800D 03            O 5280D 03
           43            O 5040D 03            O 4800D 03
           44            O 5160D 03            O 4500D 03
           45            O 4800D 03            O 1392D 04
           46            O 4800D 03            O 1272D 04
           47            O 4800D 03            O 1128D 04
           48            O 4920D 03            O 9840D 03
           49            O 5040D 03            O 8400D 03
           50            O 5040D 03            O 7680D 03
           51            O 5280D 03            O 6960D 03
           52            O 5280D 03            O 6480D 03
           53            O 5280D 03            O 6000D 03
           54            O 5280D 03            O 5520D 03
           55            O 5760D 03            O 5040D 03
           56            O 6000D 03            O 4680D 03
           57            O 5040D 03            O 1392D 04
           58            O 5040D 03            O 1272D 04
           59            O 5040D 03            O.1128D 04
           60            O 5220D 03            O 9840D 03
           61            O 5400D 03            O.8400D 03
           62            O 5400D 03            O 7680D 03
           63            O 5640D 03            O 6960D 03
           64            O 6000D 03            O 5520D 03
           65            O 6480D 03            O 5040D 03
           66            O 6840D 03            O 4860D 03
           67            O.5520D 03            O 1392D 04
           68            O 5520D 03            O 1344D 04
           69            O 5280D 03            O 1320D 04
           70            O 5280D 03            O 1272D 04
           71            O 5280D 03            O 1128D 04
           72            O 5520D 03            O 9840D 03
           73            O 5760D 03            O 8400D 03
           74            O 6000D 03            O 7680D 03
           75            O 6000D 03            O 6960D 03
           76            O 6720D 03            O 8160D 03
           77            O 6720D 03            O 7680D 03
           78            O 6360D 03            O 6960D 03
           79            O 6720D 03            O 6960D 03
           80            O 6720D 03            O 6480D 03
           81            O 6720D 03            O 6000D 03
```

```
 82            0 6720D 03           0 5520D 03
 83            0 7200D 03           0 7440D 03
 84            0 7200D 03           0 6720D 03
 85            0.7200D 03           0 6000D 03
 86            0 7200D 03           0 5280D 03
 87            0 7680D 03           0 7920D 03
 88            0 7680D 03           0 6960D 03
 89            0 7680D 03           0.6000D 03
 90            0 7680D 03           0 5040D 03

NUMBER OF ELEMENTS IN STRUCTURE =  65
NUMBER OF JOINT ELEMENTS        =  16

ELEMENT NO    NODE 1      NODE 2      NODE 3      NODE 4
     1           1           2           7           6
     2           2           3           8           7
     3           3           4           9           8
     4           4           5          10           9
     5           6           7          12          11
     6           7           8          13          12
     7           8           9          14          13
     8           9          10          15          14
     9          11          12          17          16
    10          12          13          18          17
    11          13          14          19          18
    12          14          15          20          19
    13          16          17          29          28
    14          17          18          30          29
    15          18          19          31          30
    16          19          20          32          31
    17          21          22          34          33
    18          22          23          35          34
    19          23          24          36          35
    20          24          25          37          36
    21          25          26          38          37
    22          26          27          39          38
    23          27          28          40          39
    24          28          29          41          40
    25          29          30          42          41
    26          30          31          43          42
    27          31          32          44          43
    28          33          34          46          45
    29          34          35          47          46
    30          35          36          48          47
    31          36          37          49          48
    32          37          38          50          49
    33          38          39          51          50
    34          39          40          52          51
    35          40          41          53          52
    36          41          42          54          53
    37          42          43          55          54
    38          43          44          56          55
    39          45          46          58          57
    40          46          47          59          58
    41          47          48          60          59
    42          48          49          61          60
    43          49          50          62          61
    44          50          51          63          62
    45          54          55          65          64
```

```
   46          55          56          66          65
   47          57          69          68          67
   48          57          58          70          69
   49          58          59          71          70
   50          59          60          72          71
   51          60          61          73          72
   52          61          62          74          73
   53          62          63          75          74
   54          64          65          86          82
   55          65          66          90          86
   56          73          74          77          76
   57          74          75          78          77
   58          76          77          83          87
   59          77          78          79          83
   60          79          80          84          83
   61          83          84          88          87
   62          80          81          85          84
   63          84          85          89          88
   64          81          82          86          85
   65          85          86          90          89
   66          92          91          70          71     JOINT ELEMENT
   67          93          92          71          72     JOINT ELEMENT
   68          94          93          72          73     JOINT ELEMENT
   69          95          94          73          76     JOINT ELEMENT
   70          96          95          76          87     JOINT ELEMENT
   71          97          96          87          88     JOINT ELEMENT
   72          98          97          88          89     JOINT ELEMENT
   73          99          98          89          90     JOINT ELEMENT
   74         100          99          90          66     JOINT ELEMENT
   75         101         100          66          56     JOINT ELEMENT
   76         102         101          56          44     JOINT ELEMENT
   77         103         102          44          32     JOINT ELEMENT
   78         104         103          32          20     JOINT ELEMENT
   79         105         104          20          15     JOINT ELEMENT
   80         106         105          15          10     JOINT ELEMENT
   81         107         106          10           5     JOINT ELEMENT

NUMBER OF PRESCRIBED DISPLACEMENT IN STRUCTURE =   5

NODE   1        X =  0 0000D 00       INCH
NODE   2        X =  0 0000D 00       INCH
NODE   3        X =  0 0000D 00       INCH
NODE   4        X =  0 0000D 00       INCH
NODE   5        X =  0 0000D 00       INCH

NUMBER OF PRESCRIBED NON-ZERO LOAD IN STRUCTURE =  11

NODE   1        Y = -0 1810D 01       KIPS
NODE   6        Y = -0 3318D 01       KIPS
NODE  11        Y = -0 2714D 01       KIPS
NODE  16        Y = -0 2111D 01       KIPS
NODE  22        X =  0 8750D-01       KIPS
NODE  23        X =  0.7492D 00       KIPS
NODE  24        X =  0 1498D 01       KIPS
NODE  25        X =  0 1592D 01       KIPS
NODE  26        X =  0 1310D 01       KIPS
NODE  27        X =  0 2053D 01       KIPS
NODE  28        X =  0 1422D 01       Y = -0 9050D 00    KIPS
```

```
INTERFACE INPUT
----------------

         UNIT NORMAL STIFFNESS          =  O 3000D 04    KPI
         UNIT TANGENTIAL STIFFNESS      =  O 5560D O1    KPI


         NUMBER OF NODES IN INTERFACE =  17

         NODE            X COORDINATE           Y COORDINATE
          91              O 5280D 03             O 1271D 04
          92              O 5280D 03             O 1128D 04
          93              O 5520D 03             O 9840D 03
          94              O 5760D 03             O 8400D 03
          95              O 6720D 03             O 8160D 03
          96              O 7680D 03             O 7920D 03
          97              O 7680D 03             O 6960D 03
          98              O 7680D 03             O 6000D 03
          99              O 7680D 03             O 5040D 03
         100              O 6840D 03             O 4860D 03
         101              O 6000D 03             O 4680D 03
         102              O 5160D 03             O 4500D 03
         103              O 4320D 03             O 4320D 03
         104              O 3600D 03             O 4320D 03
         105              O 2640D 03             O 4320D 03
         106              O 1440D 03             O 4320D 03
         107              O 1000D 01             O 4320D 03


         NUMBER OF ELEMENTS IN INTERFACE =  16

         ELEMENT NO     NODE 1       NODE 2
             82            91           92
             83            92           93
             84            93           94
             85            94           95
             86            95           96
             87            96           97
             88            97           98
             89            98           99
             90            99          100
             91           100          101
             92           101          102
             93           102          103
             94           103          104
             95           104          105
             96           105          106
             97           106          107


         NUMBER OF PRESCRIBED DISPLACEMENT IN INTERFACE =   O


         NUMBER OF PRESCRIBED NON-ZERO LOAD IN INTERFACE =   O
```

```
SOIL INPUT:
-----------

         ELASTIC MODULUS =  O 1500D 02   KSI
         POISSON RATIO   =  O 3500D 00
         UNIT SELFWEIGHT =  O 7407D-04   KCI


         NUMBER OF NODES IN SOIL          =  38
         NUMBER OF NODES FOR INTEGRATION =  64

         NODE            X COORDINATE           Y COORDINATE
         108              O 0000D 00             O 4310D 03
         109              O 0000D 00             O 3600D 03
         110              O 0000D 00             O 2640D 03
         111              O 0000D 00             O 1440D 03
         112              O 0000D 00             O 0000D 00
         113              O 0000D 00             O 0000D 00
         114              O 1440D 03             O 0000D 00
         115              O 2640D 03             O 0000D 00
         116              O 3600D 03             O 0000D 00
         117              O 4320D 03             O 0000D 00
         118              O 5160D 03             O 0000D 00
         119              O.6000D 03             O 0000D 00
         120              O 6840D 03             O 0000D 00
         121              O 7680D 03             O 0000D 00
         122              O 8640D 03             O 0000D 00
         123              O 1008D 04             O 0000D 00
         124              O 1200D 04             O 0000D 00
         125              O 1440D 04             O 0000D 00
         126              O 1680D 04             O 0000D 00
         127              O 1680D 04             O 0000D 00
         128              O 1680D 04             O 1440D 03
         129              O 1680D 04             O 2640D 03
         130              O 1680D 04             O 4080D 03
         131              O 1680D 04             O.5040D 03
         132              O 1680D 04             O 6000D 03
         133              O 1680D 04             O 6960D 03
         134              O. 1680D 04            O 7920D 03
         135              O 1680D 04             O 9840D 03
         136              O. 1680D 04            O 1128D 04
         137              O 1680D 04             O 1272D 04
         138              O 1680D 04             O 1272D 04
         139              O 1440D 04             O. 1272D 04
         140              O. 1200D 04            O 1272D 04
         141              O 1008D 04             O 1272D 04
         142              O 8640D 03             O 1272D 04
         143              O 7680D 03             O 1272D 04
         144              O 6720D 03             O 1272D 04
         145              O 5290D 03             O. 1272D 04
         146              O 1440D 03             O 3600D 03
         147              O 1440D 03             O 2640D 03
         148              O 1440D 03             O 1440D 03
         149              O 2640D 03             O 3600D 03
         150              O 2640D 03             O 2640D 03
         151              O 2640D 03             O 1440D 03
         152              O 3600D 03             O 3600D 03
         153              O 3600D 03             O 2640D 03
```

```
154              O 3600D 03           O 1440D 03
155              O 4320D 03           O 3600D 03
156              O 4320D 03           O 2640D 03
157              O 4320D 03           O 1440D 03
158              O 5160D 03           O 3600D 03
159              O 5160D 03           O 2640D 03
160              O 5160D 03           O 1440D 03
161              O 6000D 03           O 3600D 03
162              O 6000D 03           O 2640D 03
163              O 6000D 03           O 1440D 03
164              O 6720D 03           O 1128D 04
165              O 6720D 03           O 9840D 03
166              O 6840D 03           O 3600D 03
167              O 6840D 03           O 2640D 03
168              O 6840D 03           O 1440D 03
169              O 7680D 03           O 1128D 04
170              O 7680D 03           O.9840D 03
171              O 7680D 03           O 4080D 03
172              O 7680D 03           O 2640D 03
173              O 7680D 03           O 1440D 03
174              O 8640D 03           O 1128D 04
175              O 8640D 03           O 9840D 03
176              O 8640D 03           O 7920D 03
177              O 8640D 03           O 6960D 03
178              O 8640D 03           O 6000D 03
179              O 8640D 03           O 5040D 03
180              O 8640D 03           O 4080D 03
181              O 8640D 03           O 2640D 03
182              O 8640D 03           O 1440D 03
183              O 1008D 04           O 1128D 04
184              O 1008D 04           O 9840D 03
185              O 1008D 04           O 7920D 03
186              O 1008D 04           O 6960D 03
187              O 1008D 04           O 6000D 03
188              O 1008D 04           O 5040D 03
189              O 1008D 04           O 4080D 03
190              O.1008D 04           O 2640D 03
191              O 1008D 04           O 1440D 03
192              O 1200D 04           O 1128D 04
193              O 1200D 04           O 9840D 03
194              O.1200D 04           O 7920D 03
195              O 1200D 04           O 6960D 03
196              O 1200D 04           O 6000D 03
197              O 1200D 04           O 5040D 03
198              O 1200D 04           O 4080D 03
199              O 1200D 04           O 2640D 03
200              O 1200D 04           O 1440D 03
201              O.1440D 04           O 1128D 04
202              O.1440D 04           O 9840D 03
203              O 1440D 04           O 7920D 03
204              O 1440D 04           O.6960D 03
205              O 1440D 04           O 6000D 03
206              O 1440D 04           O 5040D 03
207              O 1440D 04           O.4080D 03
208              O 1440D 04           O 2640D 03
209              O 1440D 04           O 1440D 03

NUMBER OF ELEMENTS IN SOIL      =   36
```

```
ELEMENT NO     NODE 1      NODE 2
    98          107         108
    99          108         109
   100          109         110
   101          110         111
   102          111         112
   103          113         114
   104          114         115
   105          115         116
   106          116         117
   107          117         118
   108          118         119
   109          119         120
   110          120         121
   111          121         122
   112          122         123
   113          123         124
   114          124         125
   115          125         126
   116          127         128
   117          128         129
   118          129         130
   119          130         131
   120          131         132
   121          132         133
   122          133         134
   123          134         135
   124          135         136
   125          136         137
   126          138         139
   127          139         140
   128          140         141
   129          141         142
   130          142         143
   131          143         144
   132          144         145
   133          145          91

NUMBER OF ELEMENTS FOR INTEGRATION =  88

ELEMENT NO     NODE 1      NODE 2      NODE 3      NODE 4
   134          108         109         146         106
   135          109         110         147         146
   136          110         111         148         147
   137          111         112         114         148
   138          106         146         149         105
   139          146         147         150         149
   140          147         148         151         150
   141          148         114         115         151
   142          105         149         152         104
   143          149         150         153         152
   144          150         151         154         153
   145          151         115         116         154
   146          104         152         155         103
   147          152         153         156         155
   148          153         154         157         156
   149          154         116         117         157
   150          103         155         158         102
   151          155         156         159         158
```

```
 152      156      157      160      159
 153      157      117      118      160
 154      102      158      161      101
 155      158      159      162      161
 156      159      160      163      162
 157      160      118      119      163
 158       91       92      164      144
 159       92       93      165      164
 160       93       94       95      165
 161      101      161      166      100
 162      161      162      167      166
 163      162      163      168      167
 164      163      119      120      168
 165      144      164      169      143
 166      164      165      170      169
 167      165       95       96      170
 168      100      166      171       99
 169      166      167      172      171
 170      167      168      173      172
 171      168      120      121      173
 172      143      169      174      142
 173      169      170      175      174
 174      170       96      176      175
 175       96       97      177      176
 176       97       98      178      177
 177       98       99      179      178
 178       99      171      180      179
 179      171      172      181      180
 180      172      173      182      181
 181      173      121      122      182
 182      142      174      183      141
 183      174      175      184      183
 184      175      176      185      184
 185      176      177      186      185
 186      177      178      187      186
 187      178      179      188      187
 188      179      180      189      188
 189      180      181      190      189
 190      181      182      191      190
 191      182      122      123      191
 192      141      183      192      140
 193      183      184      193      192
 194      184      185      194      193
 195      185      186      195      194
 196      186      187      196      195
 197      187      188      197      196
 198      188      189      198      197
 199      189      190      199      198
 200      190      191      200      199
 201      191      123      124      200
 202      140      192      201      139
 203      192      193      202      201
 204      193      194      203      202
 205      194      195      204      203
 206      195      196      205      204
 207      196      197      206      205
 208      197      198      207      206
 209      198      199      208      207
 210      199      200      209      208
 211      200      124      125      209
```

```
    212          139          201          136          137
    213          201          202          135          136
    214          202          203          134          135
    215          203          204          133          134
    216          204          205          132          133
    217          205          206          131          132
    218          206          207          130          131
    219          207          208          129          130
    220          208          209          128          129
    221          209          125          126          128

 NUMBER OF PRESCRIBED DISPLACEMENT IN SOIL =  25

 NODE 113         Y =  0 0000D 00         INCH
 NODE 114         Y =  0 0000D 00         INCH
 NODE 115         Y =  0 0000D 00         INCH
 NODE 116         Y =  0 0000D 00         INCH
 NODE 117         Y =  0 0000D 00         INCH
 NODE 118         Y =  0 0000D 00         INCH
 NODE 119         Y =  0 0000D 00         INCH
 NODE 120         Y =  0 0000D 00         INCH
 NODE 121         Y =  0 0000D 00         INCH
 NODE 122         Y =  0 0000D 00         INCH
 NODE 123         Y =  0 0000D 00         INCH
 NODE 124         Y =  0 0000D 00         INCH
 NODE 125         Y =  0 0000D 00         INCH
 NODE 126         Y =  0 0000D 00         INCH
 NODE 127         X =  0 0000D 00         INCH
 NODE 128         X =  0 0000D 00         INCH
 NODE 129         X =  0 0000D 00         INCH
 NODE 130         X =  0 0000D 00         INCH
 NODE 131         X =  0 0000D 00         INCH
 NODE 132         X =  0 0000D 00         INCH
 NODE 133         X =  0 0000D 00         INCH
 NODE 134         X =  0 0000D 00         INCH
 NODE 135         X =  0 0000D 00         INCH
 NODE 136         X =  0 0000D 00         INCH
 NODE 137         X =  0 0000D 00         INCH

 NUMBER OF PRESCRIBED NON-ZERO TRACTION IN SOIL =    1

 NODE 141         Y = -0 3750D-01        KPI



 **********************
 *  STRUCTURE OUTPUT  *
 **********************


      NODAL DISPLACEMENT

NODE       U              V
  1     0 0000D 00    -0 1200D 01
  2     0 0000D 00    -0 1199D 01
```

```
  3     0 0000D 00    -0 1199D 01
  4     0 0000D 00    -0 1199D 01
  5     0 0000D 00    -0 1200D 01
  6     0 1419D-01    -0 1218D 01
  7     0 5891D-02    -0 1217D 01
  8    -0 2319D-02    -0 1216D 01
  9    -0 1053D-01    -0 1216D 01
 10    -0 1882D-01    -0 1217D 01
 11     0 2497D-01    -0 1258D 01
 12     0 1019D-01    -0 1257D 01
 13    -0 4271D-02    -0 1257D 01
 14    -0 1871D-01    -0 1257D 01
 15    -0 3350D-01    -0 1257D 01
 16     0 3140D-01    -0 1305D 01
 17     0 1230D-01    -0 1304D 01
 18    -0 5925D-02    -0 1304D 01
 19    -0 2405D-01    -0 1304D 01
 20    -0 4277D-01    -0 1304D 01
 21     0 5938D 00    -0 1350D 01
 22     0 4998D 00    -0 1350D 01
 23     0 3888D 00    -0 1351D 01
 24     0 2856D 00    -0 1352D 01
 25     0 1914D 00    -0 1353D 01
 26     0 1465D 00    -0 1353D 01
 27     0 1030D 00    -0 1352D 01
 28     0 3377D-01    -0 1348D 01
 29     0 1308D-01    -0 1346D 01
 30    -0 7003D-02    -0 1345D 01
 31    -0 2687D-01    -0 1345D 01
 32    -0 4714D-01    -0 1345D 01
 33     0 5938D 00    -0 1369D 01
 34     0 4998D 00    -0 1369D 01
 35     0 3887D 00    -0 1369D 01
 36     0 2855D 00    -0 1373D 01
 37     0 1913D 00    -0 1376D 01
 38     0 1465D 00    -0 1375D 01
 39     0 1030D 00    -0 1380D 01
 40     0 6168D-01    -0 1378D 01
 41     0 3416D-01    -0 1376D 01
 42     0 6497D-02    -0 1375D 01
 43    -0 2191D-01    -0 1388D 01
 44    -0 3996D-01    -0 1395D 01
 45     0 5938D 00    -0 1388D 01
 46     0 4999D 00    -0 1388D 01
 47     0 3887D 00    -0 1387D 01
 48     0 2855D 00    -0 1393D 01
 49     0 1913D 00    -0 1399D 01
 50     0 1467D 00    -0 1397D 01
 51     0 1027D 00    -0 1410D 01
 52     0 7608D-01    -0 1407D 01
 53     0 4854D-01    -0 1405D 01
 54     0 2077D-01    -0 1403D 01
 55    -0 9258D-02    -0 1432D 01
 56    -0 3206D-01    -0 1447D 01
 57     0 5938D 00    -0 1407D 01
 58     0 5000D 00    -0 1406D 01
 59     0 3888D 00    -0 1405D 01
 60     0 2855D 00    -0 1414D 01
 61     0 1913D 00    -0 1422D 01
 62     0 1469D 00    -0 1420D 01
```

```
 63     0 1023D 00    -0 1435D 01
 64     0 1991D-01    -0 1447D 01
 65    -0 1084D-01    -0 1478D 01
 66    -0 2266D-01    -0 1501D 01
 67     0 5938D 00    -0 1444D 01
 68     0 5563D 00    -0 1444D 01
 69     0 5375D 00    -0 1425D 01
 70     0 5001D 00    -0 1425D 01
 71     0 3889D 00    -0 1422D 01
 72     0 2857D 00    -0 1434D 01
 73     0 1913D 00    -0 1444D 01
 74     0 1466D 00    -0 1458D 01
 75     0 1021D 00    -0 1458D 01
 76     0 1751D 00    -0 1502D 01
 77     0 1458D 00    -0 1502D 01
 78     0 1021D 00    -0 1480D 01
 79     0 1021D 00    -0 1501D 01
 80     0 7352D-01    -0 1500D 01
 81     0 4559D-01    -0 1498D 01
 82     0 1816D-01    -0 1494D 01
 83     0 1310D 00    -0 1531D 01
 84     0 8798D-01    -0 1529D 01
 85     0 4611D-01    -0 1526D 01
 86     0 3292D-02    -0 1525D 01
 87     0 1599D 00    -0 1561D 01
 88     0 1023D 00    -0 1558D 01
 89     0 4610D-01    -0 1555D 01
 90    -0 1144D-01    -0 1554D 01


                         ELEMENT DISPLACEMENT AND STRESS

ELEMENT    U                V                SXX              SYY              SXY
   1     0 5020D-02    -0 1209D 01     0 2138D 00    -0 2796D-01    -0 6956D-02
   2     0 8930D-03    -0 1208D 01     0 2810D-01    -0 3478D-01    -0 8641D-02
   3    -0 3213D-02    -0 1208D 01    -0 1572D 00    -0 4304D-01    -0 8469D-02
   4    -0 7340D-02    -0 1208D 01    -0 3431D 00    -0 5053D-01    -0 7027D-02
   5     0 1381D-01    -0 1237D 01     0 1920D 00    -0 2740D-01    -0 1822D-01
   6     0 2373D-02    -0 1237D 01     0 1998D-01    -0 3396D-01    -0 2607D-01
   7    -0 8958D-02    -0 1237D 01    -0 1495D 00    -0 4364D-01    -0 2657D-01
   8    -0 2039D-01    -0 1237D 01    -0 3220D 00    -0 5228D-01    -0 1831D-01
   9     0 1972D-01    -0 1281D 01     0 1344D 00    -0 2379D-01    -0 2193D-01
  10     0 3073D-02    -0 1281D 01    -0 3937D-02    -0 3449D-01    -0 4501D-01
  11    -0 1324D-01    -0 1280D 01    -0 1317D 00    -0 4528D-01    -0 4695D-01
  12    -0 2975D-01    -0 1281D 01    -0 2612D 00    -0 5311D-01    -0 2831D-01
  13     0 2264D-01    -0 1326D 01     0 4961D-01    -0 6113D-01    -0 4331D-01
  14     0 3113D-02    -0 1325D 01    -0 2790D-01    -0 6409D-01    -0 5834D-01
  15    -0 1596D-01    -0 1325D 01    -0 1066D 00    -0 5959D-01    -0 5561D-01
  16    -0 3521D-01    -0 1324D 01    -0 1800D 00    -0 6026D-01    -0 3182D-01
  17     0 5468D 00    -0 1360D 01     0 3459D-03    -0 4159D-02     0 1014D-02
  18     0 4443D 00    -0 1360D 01    -0 2412D-02     0 1733D-02     0 8546D-02
  19     0 3372D 00    -0 1361D 01    -0 7198D-02     0 1208D-01     0 4064D-02
  20     0 2385D 00    -0 1363D 01    -0 1439D-01    -0 8632D-02    -0 6528D-02
  21     0 1689D 00    -0 1364D 01    -0 1535D-01    -0 3354D-01    -0 6507D-02
  22     0 1247D 00    -0 1365D 01    -0 1980D-01    -0 6150D-01    -0 4141D-03
  23     0 7535D-01    -0 1364D 01    -0 3033D-01    -0 1206D 00    -0 2679D-01
  24     0 3567D-01    -0 1362D 01    -0 1452D-01    -0 1223D 00    -0 2560D-01
  25     0 1168D-01    -0 1361D 01    -0 1783D-01    -0 1008D 00    -0 3059D-01
```

```
 26   -0 1232D-01   -0 1363D 01   -0 8096D-01   -0 7861D-01   -0 3008D-01
 27   -0 3397D-01   -0 1368D 01   -0 1250D 00   -0 6167D-01   -0 2459D-01
 28    0 5468D 00   -0 1378D 01    0 1896D-02   -0 5877D-02    0 1332D-02
 29    0 4443D 00   -0 1378D 01   -0 1025D-02   -0 1540D-01    0 1143D-01
 30    0 3371D 00   -0 1380D 01   -0 1002D-01   -0 2558D-01    0 5934D-02
 31    0 2384D 00   -0 1385D 01   -0 1720D-01   -0 4235D-01   -0 7924D-02
 32    0 1689D 00   -0 1387D 01   -0 1096D-01   -0 6086D-01   -0 1546D-01
 33    0 1247D 00   -0 1391D 01   -0 3319D-01   -0 9748D-01   -0 2206D-01
 34    0 8586D-01   -0 1394D 01   -0 3697D-01   -0 1662D 00   -0 4525D-01
 35    0 5512D-01   -0 1392D 01   -0 3742D-02   -0 1348D 00   -0 9843D-02
 36    0 2749D-01   -0 1390D 01   -0 8573D-02   -0 1192D 00   -0 2356D-02
 37   -0 9746D-03   -0 1399D 01   -0 6386D-01   -0 5064D-01   -0 6473D-02
 38   -0 2580D-01   -0 1416D 01   -0 1161D 00   -0 5626D-01   -0 1795D-01
 39    0 5469D 00   -0 1397D 01    0 3723D-02   -0 7216D-02    0 3404D-03
 40    0 4443D 00   -0 1396D 01    0 7123D-03   -0 3314D-01    0 1087D-01
 41    0 3371D 00   -0 1400D 01   -0 1267D-01   -0 6271D-01    0 7310D-02
 42    0 2384D 00   -0 1407D 01   -0 2173D-01   -0 7608D-01   -0 4231D-02
 43    0 1691D 00   -0 1410D 01   -0 1624D-01   -0 7581D-01   -0 2282D-01
 44    0 1247D 00   -0 1416D 01   -0 4167D-01   -0 9705D-01   -0 5178D-01
 45    0 5147D-02   -0 1440D 01   -0 6514D-01   -0 3261D-01   -0 1353D-01
 46   -0 1870D-01   -0 1465D 01   -0 8976D-01   -0 6617D-01   -0 2565D-01
 47    0 5703D 00   -0 1430D 01    0 6537D-04   -0 4187D-02   -0 1358D-04
 48    0 5329D 00   -0 1416D 01    0 6161D-02   -0 5977D-02   -0 3417D-02
 49    0 4445D 00   -0 1415D 01    0 2845D-02   -0 5176D-01    0 7089D-02
 50    0 3372D 00   -0 1419D 01   -0 1400D-01   -0 1011D 00    0 8312D-02
 51    0 2385D 00   -0 1429D 01   -0 2768D-01   -0 1126D 00    0 5754D-02
 52    0 1690D 00   -0 1436D 01   -0 3544D-01   -0 7281D-01   -0 1643D-01
 53    0 1245D 00   -0 1443D 01   -0 2909D-01   -0 2980D-01   -0 2438D-01
 54    0 7630D-02   -0 1486D 01   -0 7724D-01   -0 7728D-01   -0 4038D-01
 55   -0 1041D-01   -0 1514D 01   -0 4330D-01   -0 8175D-01   -0 7829D-02
 56    0 1647D 00   -0 1477D 01   -0 5527D-01   -0 3816D-01    0 7495D-02
 57    0 1241D 00   -0 1475D 01   -0 2916D-01   -0 1895D-01    0 4557D-02
 58    0 1530D 00   -0 1524D 01   -0 3582D-01   -0 5726D-01   -0 2808D-02
 59    0 1202D 00   -0 1504D 01   -0 2589D-01   -0 5179D-01    0 1501D-01
 60    0 9865D-01   -0 1515D 01   -0 2069D-01   -0 9977D-01    0 8086D-02
 61    0 1203D 00   -0 1545D 01   -0 2708D-01   -0 9814D-01   -0 7099D-02
 62    0 6330D-01   -0 1513D 01   -0 7551D-02   -0 1249D 00   -0 1545D-01
 63    0 7063D-01   -0 1542D 01   -0 2950D-01   -0 1209D 00   -0 1845D-01
 64    0 2829D-01   -0 1511D 01   -0 6027D-01   -0 1519D 00   -0 4516D-01
 65    0 2102D-01   -0 1540D 01   -0 3896D-01   -0 7621D-01   -0 1286D-01
```

```
            RELATIVE DISPLACEMENT AND LOCAL STRESS AT JOINT ELEMENT

ELEMENT      LOCAL X AXIS         TAN  DISP      NOR  DISP       TAU           SIGMA
  66       NODE 92 TO NODE 91     0 2848D-03   -0 2476D-05    0 1583D-02   -0 7428D-02
  67       NODE 93 TO NODE 92     0 1095D-02   -0 7810D-05    0 6090D-02   -0 2343D-01
  68       NODE 94 TO NODE 93     0 1073D-02   -0 5391D-05    0 5964D-02   -0 1617D-01
  69       NODE 95 TO NODE 94    -0 7286D-03   -0 1176D-04   -0 4051D-02   -0 3527D-01
  70       NODE 96 TO NODE 95    -0 7254D-03   -0 2385D-04   -0 4033D-02   -0 7156D-01
  71       NODE 97 TO NODE 96     0 1964D-02   -0 8693D-05    0 1092D-01   -0 2608D-01
  72       NODE 98 TO NODE 97     0 1483D-02   -0 1198D-04    0 8246D-02   -0 3594D-01
  73       NODE 99 TO NODE 98    -0 4783D-03   -0 8959D-05   -0 2660D-02   -0 2688D-01
  74       NODE100 TO NODE 99    -0 2171D-03   -0 1788D-04   -0 1207D-02   -0 5364D-01
  75       NODE101 TO NODE100    -0 6284D-03   -0 9434D-05   -0 3494D-02   -0 2830D-01
  76       NODE102 TO NODE101    -0 1021D-02   -0 1697D-04   -0 5676D-02   -0 5091D-01
  77       NODE103 TO NODE102    -0 6734D-03   -0 1343D-04   -0 3744D-02   -0 4030D-01
  78       NODE104 TO NODE103    -0 1728D-03   -0 2351D-04   -0 9607D-03   -0 7054D-01
  79       NODE105 TO NODE104    -0 1951D-03   -0 1648D-04   -0 1085D-02   -0 4945D-01
  80       NODE106 TO NODE105    -0 1434D-03   -0 1720D-04   -0 7972D-03   -0 5161D-01
  81       NODE107 TO NODE106     0 6673D-04   -0 1345D-04    0 3710D-03   -0 4036D-01
```

```
  ***********************
  *  INTERFACE OUTPUT   *
  ***********************


          DISPLACEMENT AND TRACTION AT INTERFACE

NODE        U              V             TXX            TYY
  91     0 5001D 00    -0 1426D 01    -0 2247D-01     0 9875D-02
  92     0 3889D 00    -0 1423D 01     0 1042D-01     0 6970D-02
  93     0 2861D 00    -0 1436D 01     0 2072D-01     0 1185D-01
  94     0 1913D 00    -0 1444D 01     0 1410D-01     0 1205D-01
  95     0 1737D 00    -0 1502D 01     0 1310D-01     0 3708D-01
  96     0.1599D 00    -0 1561D 01     0 2053D-01     0 5426D-01
  97     0 1023D 00    -0 1562D 01     0 3183D-01     0 4078D-02
  98     0 4609D-01    -0.1554D 01     0 3601D-01     0 1105D-01
  99    -0 1145D-01    -0 1554D 01     0 1868D-01    -0 4734D-01
 100    -0 2224D-01    -0 1501D 01     0 7724D-02    -0 5741D-01
 101    -0 3126D-01    -0 1447D 01     0 7259D-02    -0 5639D-01
 102    -0 3877D-01    -0 1395D 01     0 7929D-02    -0 5701D-01
 103    -0 4702D-01    -0 1345D 01     0 2201D-02    -0 6240D-01
 104    -0 4255D-01    -0 1304D 01    -0 1897D-02    -0 5759D-01
 105    -0 3333D-01    -0 1257D 01    -0 7782D-03    -0 5529D-01
 106    -0 1871D-01    -0 1217D 01    -0 6989D-03    -0 5333D-01
 107    -0 2515D-03    -0 1200D 01     0 1420D-02    -0 5364D-01
```

```
  *******************
  *  SOIL  OUTPUT   *
  *******************


          DISPLACEMENT AND TRACTION AT BOUNDARY NODE

NODE        U              V             TXX            TYY
 108     0 3376D-14    -0 1192D 01     0 0000D 00     0 0000D 00
 109     0 2484D-14    -0 1041D 01     0 0000D 00     0 0000D 00
 110     0 2894D-14    -0 8033D 00     0 0000D 00     0 0000D 00
 111     0 3575D-14    -0 4657D 00     0 0000D 00     0 0000D 00
 112     0 7903D-14     0 6236D-15     0 0000D 00     0 0000D 00
 113     0 7903D-14     0 0000D 00     0 0000D 00     0 8645D-01
 114    -0 3512D-01     0 0000D 00     0 0000D 00     0 8704D-01
 115    -0 5865D-01     0 0000D 00     0 0000D 00     0 8825D-01
 116    -0 7166D-01     0 0000D 00     0 0000D 00     0 8935D-01
 117    -0 7829D-01     0 0000D 00     0 0000D 00     0 9014D-01
 118    -0 8278D-01     0 0000D 00     0 0000D 00     0 9092D-01
 119    -0 8546D-01     0 0000D 00     0 0000D 00     0 9145D-01
```

```
120   -0 8731D-01    0 0000D 00    0 0000D 00    0 9183D-01
121   -0 8911D-01    0 0000D 00    0 0000D 00    0 9213D-01
122   -0 9124D-01    0 0000D 00    0 0000D 00    0 9253D-01
123   -0 9167D-01    0 0000D 00    0 0000D 00    0 9344D-01
124   -0 8129D-01    0 0000D 00    0 0000D 00    0 9520D-01
125   -0 4723D-01    0 0000D 00    0 0000D 00    0 9690D-01
126   -0 2434D-13    0 0000D 00    0 0000D 00    0 9781D-01
127    0 0000D 00    0 4439D-13   -0 4952D-01    0 0000D 00
128    0 0000D 00   -0 5675D 00   -0 4310D-01    0 0000D 00
129    0 0000D 00   -0 9915D 00   -0 3907D-01    0 0000D 00
130    0 0000D 00   -0 1440D 01   -0 3378D-01    0 0000D 00
131    0 0000D 00   -0 1700D 01   -0 3105D-01    0 0000D 00
132    0 0000D 00   -0 1928D 01   -0 2830D-01    0 0000D 00
133    0 0000D 00   -0 2123D 01   -0 2573D-01    0 0000D 00
134    0 0000D 00   -0 2283D 01   -0 2409D-01    0 0000D 00
135    0 0000D 00   -0 2491D 01   -0 1891D-01    0 0000D 00
136    0 0000D 00   -0 2553D 01   -0 1507D-01    0 0000D 00
137    0 0000D 00   -0 2544D 01   -0 8738D-02    0 0000D 00
138   -0 1682D-13   -0 2544D 01    0 0000D 00    0 0000D 00
139    0 1287D 00   -0 2544D 01    0 0000D 00    0 0000D 00
140    0 2734D 00   -0 2607D 01    0 0000D 00    0 0000D 00
141    0 4523D 00   -0 2742D 01    0 0000D 00   -0 3750D-01
142    0 5713D 00   -0 2261D 01    0 0000D 00    0 0000D 00
143    0 5921D 00   -0 1994D 01    0 0000D 00    0 0000D 00
144    0 5882D 00   -0 1768D 01    0 0000D 00    0 0000D 00
145    0 5074D 00   -0 1431D 01    0 0000D 00    0 0000D 00


                ELEMENT DISPLACEMENT AND STRESS

ELEMENT    U              V             SXX            SYY            SXY
134   -0 1064D-01   -0 1111D 01   -0 4788D-01   -0 3576D-01    0 1231D-02
135   -0 1563D-01   -0 9239D 00   -0 3739D-01   -0 6272D-01   -0 3357D-03
136   -0 1759D-01   -0 6370D 00   -0 4241D-01   -0 7114D-01   -0 2777D-03
137   -0 1806D-01   -0 2366D 00   -0 4802D-01   -0 8121D-01   -0 1086D-03
138   -0 3284D-01   -0 1147D 01   -0 4043D-01   -0 4793D-01   -0 1074D-02
139   -0 4185D-01   -0 9473D 00   -0 3743D-01   -0 6388D-01   -0 9319D-03
140   -0 4641D-01   -0 6527D 00   -0 4216D-01   -0 7226D-01   -0 7125D-03
141   -0 4767D-01   -0 2428D 00   -0 4767D-01   -0 8222D-01   -0 2693D-03
142   -0 4677D-01   -0 1192D 01   -0 3573D-01   -0 5688D-01   -0 1451D-02
143   -0 5748D-01   -0 9791D 00   -0 3724D-01   -0 6583D-01   -0 1289D-02
144   -0 6315D-01   -0 6722D 00   -0 4175D-01   -0 7380D-01   -0 8294D-03
145   -0 6552D-01   -0 2499D 00   -0 4724D-01   -0 8347D-01   -0 2939D-03
146   -0 5245D-01   -0 1233D 01   -0 3409D-01   -0 6220D-01   -0 1681D-02
147   -0 6267D-01   -0 1006D 01   -0 3682D-01   -0 6780D-01   -0 9870D-03
148   -0 7042D-01   -0 6875D 00   -0 4142D-01   -0 7506D-01   -0 5933D-03
149   -0 7464D-01   -0 2553D 00   -0 4694D-01   -0 8448D-01   -0 2120D-03
150   -0 4815D-01   -0 1265D 01   -0 3343D-01   -0 6291D-01    0 8165D-03
151   -0 6226D-01   -0 1021D 01   -0 3677D-01   -0 6866D-01   -0 4344D-04
152   -0 7349D-01   -0 6977D 00   -0 4130D-01   -0 7590D-01   -0 1954D-03
153   -0 7985D-01   -0 2587D 00   -0 4678D-01   -0 8525D-01   -0 8979D-04
154   -0 4537D-01   -0 1295D 01   -0 3292D-01   -0 6172D-01    0 5208D-03
155   -0 6191D-01   -0 1030D 01   -0 3704D-01   -0 6876D-01    0 3473D-03
156   -0 7495D-01   -0 7050D 00   -0 4137D-01   -0 7642D-01    0 1500D-03
157   -0 8310D-01   -0 2615D 00   -0 4675D-01   -0 8589D-01    0 2989D-04
158    0 4392D 00   -0 1579D 01    0 1220D-02   -0 2517D-02   -0 6171D-02
159    0 3093D 00   -0 1570D 01   -0 1365D-01   -0 1162D-01   -0 7875D-02
160    0 1915D 00   -0 1533D 01   -0 1627D-01   -0 2188D-01   -0 6136D-02
```

```
161   -0 4188D-01   -0 1326D 01   -0 3276D-01   -0 6180D-01    0 7901D-03
162   -0 6176D-01   -0 1036D 01   -0 3711D-01   -0 6912D-01    0 5861D-03
163   -0 7591D-01   -0 7092D 00   -0 4153D-01   -0 7676D-01    0 3753D-03
164   -0 8510D-01   -0 2632D 00   -0 4685D-01   -0 8633D-01    0 1163D-03
165    0 4424D 00   -0 1871D 01   -0 4974D-02   -0 5229D-02   -0 4043D-02
166    0 2684D 00   -0 1825D 01   -0 1347D-01   -0 1643D-01   -0 8170D-02
167    0 1707D 00   -0 1672D 01   -0 1767D-01   -0 3410D-01   -0 7779D-02
168   -0 3345D-01   -0 1383D 01   -0 3232D-01   -0 6144D-01    0 1686D-02
169   -0 5895D-01   -0 1072D 01   -0 3692D-01   -0 6827D-01    0 1073D-02
170   -0 7715D-01   -0 7113D 00   -0 4177D-01   -0 7692D-01    0 5204D-03
171   -0 8678D-01   -0 2642D 00   -0 4701D-01   -0 8665D-01    0 1596D-03
172    0 4171D 00   -0 2120D 01   -0 1082D-01   -0 7805D-02   -0 5250D-02
173    0 2476D 00   -0 2054D 01   -0 1423D-01   -0 2103D-01   -0 9446D-02
174    0 1705D 00   -0 1872D 01   -0 2070D-01   -0 3358D-01   -0 1226D-01
175    0 8474D-01   -0 1711D 01   -0 2417D-01   -0 2885D-01   -0 1070D-01
176    0 2448D-01   -0 1641D 01   -0 2915D-01   -0 2683D-01   -0 5087D-02
177   -0 3402D-01   -0 1555D 01   -0 2654D-01   -0 3659D-01    0 1613D-02
178   -0 3594D-01   -0 1395D 01   -0 3471D-01   -0 5413D-01    0 2951D-02
179   -0 6142D-01   -0 1101D 01   -0 3736D-01   -0 6648D-01    0 1158D-02
180   -0 7981D-01   -0 7123D 00   -0 4210D-01   -0 7695D-01    0 4382D-03
181   -0 8871D-01   -0 2647D 00   -0 4720D-01   -0 8695D-01    0 1404D-03
182    0.3703D 00   -0 2473D 01   -0 1631D-01   -0 2203D-01   -0 8005D-02
183    0 2372D 00   -0 2327D 01   -0 1441D-01   -0 2717D-01   -0 7476D-02
184    0 1425D 00   -0 2126D 01   -0 2065D-01   -0 3208D-01   -0 6517D-02
185    0 4401D-01   -0 1939D 01   -0 2321D-01   -0 3727D-01   -0 3346D-02
186   -0 1997D-01   -0 1787D 01   -0 2414D-01   -0 4226D-01   -0 2534D-02
187   -0 5652D-01   -0 1609D 01   -0 2713D-01   -0 4890D-01   -0 2414D-02
188   -0 6636D-01   -0 1404D 01   -0 3247D-01   -0 5621D-01   -0 1698D-02
189   -0 7464D-01   -0 1106D 01   -0 3747D-01   -0 6656D-01   -0 4251D-03
190   -0 8478D-01   -0 7154D 00   -0 4226D-01   -0 7738D-01   -0 9215D-04
191   -0 9073D-01   -0 2647D 00   -0 4733D-01   -0 8746D-01    0 5677D-05
192    0 3493D 00   -0 2661D 01   -0 2074D-01   -0 1870D-01    0 5325D-02
193    0 2439D 00   -0 2530D 01   -0 1590D-01   -0 2732D-01    0 1601D-02
194    0 1259D 00   -0 2322D 01   -0 1935D-01   -0 3425D-01   -0 1935D-03
195    0 3477D-01   -0 2095D 01   -0 2233D-01   -0 4220D-01   -0 3142D-03
196   -0 1487D-01   -0 1913D 01   -0 2452D-01   -0 4820D-01   -0 6943D-03
197   -0 4874D-01   -0 1703D 01   -0 2758D-01   -0 5445D-01   -0 1211D-02
198   -0 6746D-01   -0 1468D 01   -0 3147D-01   -0 6082D-01   -0 1394D-02
199   -0 7852D-01   -0 1141D 01   -0 3652D-01   -0 6930D-01   -0 1070D-02
200   -0 8456D-01   -0 7308D 00   -0 4180D-01   -0 7906D-01   -0 5738D-03
201   -0 8752D-01   -0 2676D 00   -0 4718D-01   -0 8863D-01   -0 1045D-03
202    0 2082D 00   -0 2557D 01   -0 2168D-01   -0 2697D-02    0 1806D-02
203    0 1760D 00   -0 2535D 01   -0 1819D-01   -0 1882D-01    0 2746D-02
204    0 1001D 00   -0 2387D 01   -0 2075D-01   -0 3208D-01    0 2159D-02
205    0 3630D-01   -0 2171D 01   -0 2355D-01   -0 4273D-01    0 1258D-02
206    0 1515D-02   -0 1986D 01   -0 2588D-01   -0 4960D-01    0 6493D-03
207   -0 2491D-01   -0 1771D 01   -0 2865D-01   -0 5642D-01    0 1108D-03
208   -0 4305D-01   -0 1526D 01   -0 3184D-01   -0 6319D-01   -0 2665D-03
209   -0 5640D-01   -0 1182D 01   -0 3623D-01   -0 7169D-01   -0 4517D-03
210   -0 6339D-01   -0 7525D 00   -0 4131D-01   -0 8116D-01   -0 3694D-03
211   -0 6604D-01   -0 2729D 00   -0 4700D-01   -0 8998D-01    0 3563D-04
212    0 6503D-01   -0 2533D 01   -0 1922D-01   -0 3715D-02    0 2940D-03
213    0 5778D-01   -0 2516D 01   -0 1728D-01   -0 1668D-01    0 7100D-03
214    0 3666D-01   -0 2389D 01   -0 2144D-01   -0 3025D-01    0 8528D-03
215    0 1605D-01   -0 2189D 01   -0 2461D-01   -0 4180D-01    0 6778D-03
216    0 3614D-02   -0 2011D 01   -0 2694D-01   -0 4928D-01    0 4303D-03
217   -0 6327D-02   -0 1799D 01   -0 2952D-01   -0 5664D-01    0 2232D-03
218   -0 1368D-01   -0 1554D 01   -0 3240D-01   -0 6383D-01    0 7400D-04
219   -0 1952D-01   -0 1205D 01   -0 3639D-01   -0 7269D-01   -0 4988D-04
220   -0 2287D-01   -0 7675D 00   -0 4116D-01   -0 8233D-01   -0 5093D-04
```

```
221   -0 2433D-01   -0 2778D 00   -0 4677D-01   -0 9117D-01    0 5508D-04
```

# APPENDIX E

# SELECTED PRINTOUT SHEETS FOR PROGRAM FINITE

```
                    FINITE  ELEMENT  METHOD

                          APPLIED  TO

               SOIL-STRUCTURE  INTERACTION  PROBLEMS




PROBLEM IDENTIFICATION
----------------------

          THIS PROBLEM IS TO TEST A U-FRAME STRUCTURE BY THE FINITE ELEMENT METHOD
          ALONE  IT INCLUDES 204 NODES AND 185 ELEMENTS   THE RESULTS ARE COMPARED
          WITH THOSE OBTAINED FROM THE COUPLING OF BOUNDARY AND FINITE ELEMENT
          METHODS



STRUCTURE INPUT
---------------

          ELASTIC MODULUS =  0 3000D 04   KSI
          POISSON RATIO   =  0 2500D 00
          UNIT SELFWEIGHT =  0 8681D-04   KCI


          NUMBER OF NODES IN STRUCTURE =  90

          NODE         X COORDINATE           Y COORDINATE
            1           0 0000D 00             0 5760D 03
            2           0 0000D 00             0 5400D 03
            3           0 0000D 00             0 5040D 03
            4           0 0000D 00             0 4680D 03
            5           0 0000D 00             0 4320D 03
            6           0 1440D 03             0 5760D 03
            7           0 1440D 03             0 5400D 03
            8           0 1440D 03             0 5040D 03
            9           0 1440D 03             0 4680D 03
           10           0 1440D 03             0 4320D 03
           11           0 2640D 03             0 5760D 03
           12           0 2640D 03             0 5400D 03
           13           0 2640D 03             0 5040D 03
           14           0 2640D 03             0 4680D 03
           15           0 2640D 03             0 4320D 03
           16           0 3600D 03             0 5760D 03
           17           0 3600D 03             0 5400D 03
           18           0 3600D 03             0 5040D 03
           19           0 3600D 03             0 4680D 03
           20           0.3600D 03             0 4320D 03
           21           0 4320D 03             0 1392D 04
           22           0 4320D 03             0 1272D 04
           23           0 4320D 03             0 1128D 04
           24           0 4320D 03             0 9840D 03
           25           0 4320D 03             0 8400D 03
           26           0 4320D 03             0 7680D 03
           27           0.4320D 03             0 6960D 03
           28           0 4320D 03             0 5760D 03
           29           0.4320D 03             0.5400D 03
           30           0 4320D 03             0 5040D 03
           31           0 4320D 03             0.4680D 03
           32           0 4320D 03             0 4320D 03
           33           0 4560D 03             0 1392D 04
           34           0 4560D 03             0.1272D 04
           35           0 4560D 03             0 1128D 04
           36           0 4620D 03             0 9840D 03
           37           0 4680D 03             0 8400D 03
           38           0 4680D 03             0.7680D 03
           39           0 4800D 03             0 6960D 03
           40           0 4800D 03             0 6240D 03
           41           0 4800D 03             0 5760D 03
           42           0 4800D 03             0 5280D 03
           43           0 5040D 03             0.4800D 03
           44           0 5160D 03             0 4500D 03
           45           0 4800D 03             0 1392D 04
           46           0 4800D 03             0.1272D 04
           47           0 4800D 03             0 1128D 04
           48           0 4920D 03             0 9840D 03
           49           0 5040D 03             0 8400D 03
           50           0 5040D 03             0 7680D 03
           51           0 5280D 03             0 6960D 03
           52           0 5280D 03             0.6480D 03
           53           0 5280D 03             0 6000D 03
           54           0 5280D 03             0 5520D 03
           55           0 5760D 03             0 5040D 03
           56           0 6000D 03             0 4680D 03
           57           0 5040D 03             0 1392D 04
           58           0.5040D 03             0.1272D 04
           59           0 5040D 03             0 1128D 04
           60           0 5220D 03             0 9840D 03
           61           0 5400D 03             0 8400D 03
           62           0 5400D 03             0 7680D 03
           63           0 5640D 03             0 6960D 03
           64           0 6000D 03             0.5520D 03
           65           0 6480D 03             0 5040D 03
           66           0 6840D 03             0.4860D 03
           67           0 5520D 03             0.1392D 04
           68           0 5520D 03             0 1344D 04
           69           0 5280D 03             0 1320D 04
           70           0 5280D 03             0 1272D 04
           71           0 5280D 03             0.1128D 04
           72           0 5520D 03             0 9840D 03
           73           0 5760D 03             0 8400D 03
           74           0 6000D 03             0 7680D 03
           75           0 6000D 03             0 6960D 03
           76           0 6720D 03             0 8160D 03
           77           0 6720D 03             0 7680D 03
           78           0 6360D 03             0 6960D 03
           79           0 6720D 03             0 6960D 03
           80           0 6720D 03             0 6480D 03
           81           0 6720D 03             0 6000D 03
           82           0 6720D 03             0 5520D 03
           83           0 7200D 03             0 7440D 03
           84           0 7200D 03             0 6720D 03
           85           0 7200D 03             0 6000D 03
           86           0 7200D 03             0 5280D 03
```

```
 87            0 7680D 03           0 7920D 03
 88            0 7680D 03           0 6960D 03
 89            0 7680D 03           0 6000D 03
 90            0 7680D 03           0 5040D 03

NUMBER OF ELEMENTS IN STRUCTURE =  65
NUMBER OF JOINT ELEMENTS        =  16

ELEMENT NO    NODE 1      NODE 2      NODE 3      NODE 4
     1           1           2           7           6
     2           2           3           8           7
     3           3           4           9           8
     4           4           5          10           9
     5           6           7          12          11
     6           7           8          13          12
     7           8           9          14          13
     8           9          10          15          14
     9          11          12          17          16
    10          12          13          18          17
    11          13          14          19          18
    12          14          15          20          19
    13          16          17          29          28
    14          17          18          30          29
    15          18          19          31          30
    16          19          20          32          31
    17          21          22          34          33
    18          22          23          35          34
    19          23          24          36          35
    20          24          25          37          36
    21          25          26          38          37
    22          26          27          39          38
    23          27          28          40          39
    24          28          29          41          40
    25          29          30          42          41
    26          30          31          43          42
    27          31          32          44          43
    28          33          34          46          45
    29          34          35          47          46
    30          35          36          48          47
    31          36          37          49          48
    32          37          38          50          49
    33          38          39          51          50
    34          39          40          52          51
    35          40          41          53          52
    36          41          42          54          53
    37          42          43          55          54
    38          43          44          56          55
    39          45          46          58          57
    40          46          47          59          58
    41          47          48          60          59
    42          48          49          61          60
    43          49          50          62          61
    44          50          51          63          62
    45          54          55          65          64
    46          55          56          66          65
    47          57          69          68          67
    48          57          58          70          69
    49          58          59          71          70
    50          59          60          72          71

    51          60          61          73          72
    52          61          62          74          73
    53          62          63          75          74
    54          64          65          86          82
    55          65          66          90          86
    56          73          74          77          76
    57          74          75          78          77
    58          76          77          83          87
    59          77          78          79          83
    60          79          80          84          83
    61          83          84          88          87
    62          80          81          85          84
    63          84          85          89          88
    64          81          82          86          85
    65          85          86          90          89
    66          92          91          70          71     JOINT ELEMENT
    67          93          92          71          72     JOINT ELEMENT
    68          94          93          72          73     JOINT ELEMENT
    69          95          94          73          76     JOINT ELEMENT
    70          96          95          76          87     JOINT ELEMENT
    71          97          96          87          88     JOINT ELEMENT
    72          98          97          88          89     JOINT ELEMENT
    73          99          98          89          90     JOINT ELEMENT
    74         100          99          90          66     JOINT ELEMENT
    75         101         100          66          56     JOINT ELEMENT
    76         102         101          56          44     JOINT ELEMENT
    77         103         102          44          32     JOINT ELEMENT
    78         104         103          32          20     JOINT ELEMENT
    79         105         104          20          15     JOINT ELEMENT
    80         106         105          15          10     JOINT ELEMENT
    81         107         106          10           5     JOINT ELEMENT

  NUMBER OF PRESCRIBED DISPLACEMENT IN STRUCTURE =   5

  NODE    1         X =  0 0000D 00        INCH
  NODE    2         X =  0 0000D 00        INCH
  NODE    3         X =  0 0000D 00        INCH
  NODE    4         X =  0 0000D 00        INCH
  NODE    5         X =  0 0000D 00        INCH

  NUMBER OF PRESCRIBED NON-ZERO LOAD IN STRUCTURE =  11

  NODE    1         Y = -0 1810D 01        KIPS
  NODE    6         Y = -0 3318D 01        KIPS
  NODE   11         Y = -0 2714D 01        KIPS
  NODE   16         Y = -0 2111D 01        KIPS
  NODE   22         X =  0.8750D-01        KIPS
  NODE   23         X =  0 7492D 00        KIPS
  NODE   24         X =  0 1498D 01        KIPS
  NODE   25         X =  0 1592D 01        KIPS
  NODE   26         X =  0 1310D 01        KIPS
  NODE   27         X =  0 2053D 01        KIPS
  NODE   28         X =  0 1422D 01        Y = -0 9050D 00    KIPS


INTERFACE INPUT
---------------
```

```
UNIT NORMAL STIFFNESS          =  0 3000D 04   KPI
UNIT TANGENTIAL STIFFNESS      =  0 5560D 01   KPI

NUMBER OF NODES IN INTERFACE =  17

NODE             X COORDINATE            Y COORDINATE
 91               0 5280D 03              0 1272D 04
 92               0 5280D 03              0 1128D 04
 93               0 5520D 03              0 9840D 03
 94               0 5760D 03              0 8400D 03
 95               0 6720D 03              0 8160D 03
 96               0 7680D 03              0 7920D 03
 97               0 7680D 03              0 6960D 03
 98               0 7680D 03              0 6000D 03
 99               0 7680D 03              0 5040D 03
100               0 6840D 03              0 4860D 03
101               0 6000D 03              0 4680D 03
102               0 5160D 03              0 4500D 03
103               0 4320D 03              0 4320D 03
104               0 3600D 03              0 4320D 03
105               0 2640D 03              0 4320D 03
106               0 1440D 03              0 4320D 03
107               0 0000D 00              0 4320D 03

NUMBER OF ELEMENTS IN INTERFACE =  16

ELEMENT NO     NODE 1      NODE 2
    82           91          92
    83           92          93
    84           93          94
    85           94          95
    86           95          96
    87           96          97
    88           97          98
    89           98          99
    90           99         100
    91          100         101
    92          101         102
    93          102         103
    94          103         104
    95          104         105
    96          105         106
    97          106         107

NUMBER OF PRESCRIBED DISPLACEMENT IN INTERFACE =   1

NODE 107         X =  0 0000D 00         INCH

NUMBER OF PRESCRIBED NON-ZERO LOAD IN INTERFACE =   0


SOIL INPUT
----------
```

```
ELASTIC MODULUS =  0 1500D 02   KSI
POISSON RATIO   =  0 3500D 00
UNIT SELFWEIGHT =  0 7407D-04   KCI

NUMBER OF NODES IN SOIL          =  97

NODE            X COORDINATE           Y COORDINATE
108              0 0000D 00             0 3600D 03
109              0 0000D 00             0 2640D 03
110              0 0000D 00             0 1440D 03
111              0.0000D 00             0 0000D 00
112              0 1440D 03             0 3600D 03
113              0 1440D 03             0.2640D 03
114              0 1440D 03             0 1440D 03
115              0 1440D 03             0 0000D 00
116              0 2640D 03             0 3600D 03
117              0 2640D 03             0 2640D 03
118              0 2640D 03             0 1440D 03
119              0 2640D 03             0 0000D 00
120              0 3600D 03             0 3600D 03
121              0 3600D 03             0 2640D 03
122              0 3600D 03             0 1440D 03
123              0 3600D 03             0.0000D 00
124              0 4320D 03             0 3600D 03
125              0 4320D 03             0 2640D 03
126              0 4320D 03             0 1440D 03
127              0 4320D 03             0 0000D 00
128              0.5160D 03             0 3600D 03
129              0 5160D 03             0 2640D 03
130              0 5160D 03             0 1440D 03
131              0.5160D 03             0 0000D 00
132              0 6000D 03             0.3600D 03
133              0 6000D 03             0 2640D 03
134              0 6000D 03             0 1440D 03
135              0 6000D 03             0 0000D 00
136              0 6720D 03             0 1272D 04
137              0 6720D 03             0.1128D 04
138              0 6720D 03             0 9840D 03
139              0 6840D 03             0 3600D 03
140              0 6840D 03             0 2640D 03
141              0 6840D 03             0 1440D 03
142              0 6840D 03             0.0000D 00
143              0 7680D 03             0 1272D 04
144              0 7680D 03             0 1128D 04
145              0 7680D 03             0 9840D 03
146              0 7680D 03             0 4080D 03
147              0 7680D 03             0 2640D 03
148              0 7680D 03             0.1440D 03
149              0 7680D 03             0 0000D 00
150              0 8640D 03             0 1272D 04
151              0 8640D 03             0 1128D 04
152              0 8640D 03             0 9840D 03
153              0 8640D 03             0 7920D 03
154              0 8640D 03             0.6960D 03
155              0 8640D 03             0 6000D 03
156              0 8640D 03             0 5040D 03
157              0 8640D 03             0 4080D 03
158              0 8640D 03             0 2640D 03
```

```
159            O 8640D 03           O 1440D 03
160            O 8640D 03           O 0000D 00
161            O  1008D 04          O 1272D 04
162            O. 1008D 04          O 1128D 04
163            O  1008D 04          O 9840D 03
164            O  1008D 04          O 7920D 03
165            O  1008D 04          O 6960D 03
166            O  1008D 04          O 6000D 03
167            O  1008D 04          O 5040D 03
168            O  1008D 04          O 4080D 03
169            O  1008D 04          O 2640D 03
170            O  1008D 04          O 1440D 03
171            O  1008D 04          O 0000D 00
172            O  1200D 04          O 1272D 04
173            O  1200D 04          O 1128D 04
174            O  1200D 04          O 9840D 03
175            O  1200D 04          O 7920D 03
176            O  1200D 04          O 6960D 03
177            O  1200D 04          O 6000D 03
178            O  1200D 04          O 5040D 03
179            O  1200D 04          O 4080D 03
180            O  1200D 04          O 2640D 03
181            O  1200D 04          O 1440D 03
182            O  1200D 04          O 0000D 00
183            O  1440D 04          O 1272D 04
184            O  1440D 04          O 1128D 04
185            O. 1440D 04          O 9840D 03
186            O  1440D 04          O 7920D 03
187            O  1440D 04          O 6960D 03
188            O  1440D 04          O 6000D 03
189            O  1440D 04          O 5040D 03
190            O  1440D 04          O 4080D 03
191            O  1440D 04          O 2640D 03
192            O  1440D 04          O 1440D 03
193            O  1440D 04          O 0000D 00
194            O  1680D 04          O 1272D 04
195            O. 1680D 04          O 1128D 04
196            O  1680D 04          O 9840D 03
197            O  1680D 04          O 7920D 03
198            O  1680D 04          O 6960D 03
199            O  1680D 04          O 6000D 03
200            O  1680D 04          O 5040D 03
201            O  1680D 04          O 4080D 03
202            O  1680D 04          O 2640D 03
203            O  1680D 04          O 1440D 03
204            O  1680D 04          O 0000D 00

NUMBER OF ELEMENTS IN SOIL       =  88

ELEMENT NO     NODE 1      NODE 2      NODE 3      NODE 4
    98           107         108         112         106
    99           108         109         113         112
   100           109         110         114         113
   101           110         111         115         114
   102           106         112         116         105
   103           112         113         117         116
   104           113         114         118         117
   105           114         115         119         118
   106           105         116         120         104
   107           116         117         121         120
   108           117         118         122         121
   109           118         119         123         122
   110           104         120         124         103
   111           120         121         125         124
   112           121         122         126         125
   113           122         123         127         126
   114           103         124         128         102
   115           124         125         129         128
   116           125         126         130         129
   117           126         127         131         130
   118           102         128         132         101
   119           128         129         133         132
   120           129         130         134         133
   121           130         131         135         134
   122            91          92         137         136
   123            92          93         138         137
   124            93          94          95         138
   125           101         132         139         100
   126           132         133         140         139
   127           133         134         141         140
   128           134         135         142         141
   129           136         137         144         143
   130           137         138         145         144
   131           138          95          96         145
   132           100         139         146          99
   133           139         140         147         146
   134           140         141         148         147
   135           141         142         149         148
   136           143         144         151         150
   137           144         145         152         151
   138           145          96         153         152
   139            96          97         154         153
   140            97          98         155         154
   141            98          99         156         155
   142            99         146         157         156
   143           146         147         158         157
   144           147         148         159         158
   145           148         149         160         159
   146           150         151         162         161
   147           151         152         163         162
   148           152         153         164         163
   149           153         154         165         164
   150           154         155         166         165
   151           155         156         167         166
   152           156         157         168         167
   153           157         158         169         168
   154           158         159         170         169
   155           159         160         171         170
   156           161         162         173         172
   157           162         163         174         173
   158           163         164         175         174
   159           164         165         176         175
   160           165         166         177         176
   161           166         167         178         177
   162           167         168         179         178
   163           168         169         180         179
   164           169         170         181         180
   165           170         171         182         181
   166           172         173         184         183
```

```
   167          173          174          185          184
   168          174          175          186          185
   169          175          176          187          186
   170          176          177          188          187
   171          177          178          189          188
   172          178          179          190          189
   173          179          180          191          190
   174          180          181          192          191
   175          181          182          193          192
   176          183          184          195          194
   177          184          185          196          195
   178          185          186          197          196
   179          186          187          198          197
   180          187          188          199          198
   181          188          189          200          199
   182          189          190          201          200
   183          190          191          202          201
   184          191          192          203          202
   185          192          193          204          203

NUMBER OF PRESCRIBED DISPLACEMENT IN SOIL =  27

NODE 108         X =  0 0000D 00        INCH
NODE 109         X =  0 0000D 00        INCH
NODE 110         X =  0 0000D 00        INCH
NODE 111         X =  0 0000D 00        Y =  0 0000D 00     INCH
NODE 115         Y =  0 0000D 00        INCH
NODE 119         Y =  0 0000D 00        INCH
NODE 123         Y =  0 0000D 00        INCH
NODE 127         Y =  0 0000D 00        INCH
NODE 131         Y =  0 0000D 00        INCH
NODE 135         Y =  0 0000D 00        INCH
NODE 142         Y =  0 0000D 00        INCH
NODE 149         Y =  0 0000D 00        INCH
NODE 160         Y =  0 0000D 00        INCH
NODE 171         Y =  0 0000D 00        INCH
NODE 182         Y =  0 0000D 00        INCH
NODE 193         Y =  0 0000D 00        INCH
NODE 194         X =  0 0000D 00        INCH
NODE 195         X =  0 0000D 00        INCH
NODE 196         X =  0 0000D 00        INCH
NODE 197         X =  0 0000D 00        INCH
NODE 198         X =  0 0000D 00        INCH
NODE 199         X =  0 0000D 00        INCH
NODE 200         X =  0 0000D 00        INCH
NODE 201         X =  0 0000D 00        INCH
NODE 202         X =  0 0000D 00        INCH
NODE 203         X =  0 0000D 00        INCH
NODE 204         X =  0 0000D 00        Y =  0 0000D 00     INCH

NUMBER OF PRESCRIBED NON-ZERO LOAD IN SOIL =   3

NODE 150         Y = -0 9000D 00        KIPS
NODE 161         Y = -0 4200D 01        KIPS
NODE 172         Y = -0 1200D 01        KIPS
```

```
*********************
*  STRUCTURE OUTPUT *
*********************


        NODAL DISPLACEMENT

NODE         U               V
   1    0 0000D 00    -0 1203D 01
   2    0 0000D 00    -0 1202D 01
   3    0 0000D 00    -0 1202D 01
   4    0 0000D 00    -0 1202D 01
   5    0 0000D 00    -0 1203D 01
   6    0 1411D-01    -0 1221D 01
   7    0 5848D-02    -0 1220D 01
   8   -0 2324D-02    -0 1219D 01
   9   -0 1050D-01    -0 1219D 01
  10   -0 1875D-01    -0 1220D 01
  11    0 2479D-01    -0 1261D 01
  12    0 1009D-01    -0 1260D 01
  13   -0 4281D-02    -0 1259D 01
  14   -0 1863D-01    -0 1259D 01
  15   -0 3334D-01    -0 1260D 01
  16    0 3109D-01    -0 1307D 01
  17    0 1212D-01    -0 1307D 01
  18   -0 5940D-02    -0 1306D 01
  19   -0 2390D-01    -0 1306D 01
  20   -0 4248D-01    -0 1306D 01
  21    0 5794D 00    -0 1353D 01
  22    0 4877D 00    -0 1353D 01
  23    0 3794D 00    -0 1354D 01
  24    0 2792D 00    -0 1355D 01
  25    0 1877D 00    -0 1355D 01
  26    0 1438D 00    -0 1355D 01
  27    0 1012D 00    -0 1354D 01
  28    0 3328D-01    -0 1350D 01
  29    0 1286D-01    -0 1348D 01
  30   -0 7003D-02    -0 1348D 01
  31   -0 2666D-01    -0 1347D 01
  32   -0 4672D-01    -0 1347D 01
  33    0 5794D 00    -0 1372D 01
  34    0 4877D 00    -0 1371D 01
  35    0 3793D 00    -0 1371D 01
  36    0 2791D 00    -0 1375D 01
  37    0 1876D 00    -0 1378D 01
  38    0 1438D 00    -0 1377D 01
  39    0 1012D 00    -0 1382D 01
  40    0 6070D-01    -0 1380D 01
  41    0 3363D-01    -0 1378D 01
  42    0 6330D-02    -0 1376D 01
  43   -0 2171D-01    -0 1389D 01
  44   -0 3952D-01    -0 1396D 01
  45    0 5795D 00    -0 1390D 01
  46    0 4877D 00    -0 1390D 01
  47    0 3793D 00    -0 1389D 01
  48    0 2791D 00    -0 1395D 01
  49    0 1876D 00    -0 1400D 01
  50    0 1440D 00    -0 1399D 01
  51    0 1009D 00    -0 1412D 01
  52    0 7483D-01    -0 1408D 01
  53    0 4778D-01    -0 1406D 01
  54    0 2038D-01    -0 1404D 01
  55   -0 9228D-02    -0 1433D 01
  56   -0 3164D-01    -0 1448D 01
  57    0 5794D 00    -0 1408D 01
  58    0 4878D 00    -0 1408D 01
  59    0 3794D 00    -0 1406D 01
  60    0 2791D 00    -0 1415D 01
  61    0 1877D 00    -0 1423D 01
  62    0 1442D 00    -0 1421D 01
  63    0 1004D 00    -0 1436D 01
  64    0 1942D-01    -0 1448D 01
  65   -0 1078D-01    -0 1478D 01
  66   -0 2230D-01    -0 1501D 01
  67    0 5795D 00    -0 1445D 01
  68    0 5428D 00    -0 1445D 01
  69    0 5245D 00    -0 1427D 01
  70    0 4880D 00    -0 1427D 01
  71    0 3795D 00    -0 1423D 01
  72    0 2793D 00    -0 1435D 01
  73    0 1877D 00    -0 1445D 01
  74    0 1438D 00    -0 1459D 01
  75    0 1001D 00    -0 1458D 01
  76    0 1719D 00    -0 1502D 01
  77    0 1430D 00    -0 1502D 01
  78    0 9997D-01    -0 1480D 01
  79    0 9996D-01    -0 1501D 01
  80    0 7183D-01    -0 1499D 01
  81    0 4439D-01    -0 1497D 01
  82    0 1757D-01    -0 1494D 01
  83    0 1284D 00    -0 1530D 01
  84    0 8607D-01    -0 1528D 01
  85    0 4494D-01    -0 1525D 01
  86    0 2992D-02    -0 1524D 01
  87    0 1569D 00    -0 1561D 01
  88    0 1002D 00    -0 1557D 01
  89    0 4496D-01    -0 1554D 01
  90   -0 1138D-01    -0 1552D 01


                   ELEMENT DISPLACEMENT AND STRESS

ELEMENT     U              V              SXX            SYY            SXY
   1     0 4990D-02    -0 1211D 01     0 2125D 00    -0 2775D-01    -0 6968D-02
   2     0 8810D-03    -0 1211D 01     0 2737D-01    -0 3536D-01    -0 8699D-02
   3    -0 3206D-02    -0 1210D 01    -0 1569D 00    -0 4335D-01    -0 8539D-02
   4    -0 7313D-02    -0 1211D 01    -0 3419D 00    -0 5063D-01    -0 7018D-02
   5     0 1371D-01    -0 1240D 01     0 1899D 00    -0 2733D-01    -0 1840D-01
   6     0 2334D-02    -0 1239D 01     0 1912D-01    -0 3411D-01    -0 2651D-01
   7    -0 8933D-02    -0 1239D 01    -0 1491D 00    -0 4385D-01    -0 2710D-01
   8    -0 2030D-01    -0 1240D 01    -0 3204D 00    -0 5284D-01    -0 1865D-01
   9     0 1953D-01    -0 1284D 01     0 1310D 00    -0 2342D-01    -0 2215D-01
  10     0 2998D-02    -0 1283D 01    -0 5394D-02    -0 3476D-01    -0 4578D-01
  11    -0 1319D-01    -0 1283D 01    -0 1308D 00    -0 4558D-01    -0 4764D-01
  12    -0 2959D-01    -0 1283D 01    -0 2580D 00    -0 5323D-01    -0 2862D-01
```

```
 13     0 2234D-01    -0 1328D 01     0 4405D-01    -0 6283D-01    -0 4482D-01
 14     0 3009D-02    -0 1327D 01    -0 2906D-01    -0 6542D-01    -0 5916D-01
 15    -0 1588D-01    -0 1327D 01    -0 1050D 00    -0 6040D-01    -0 5588D-01
 16    -0 3494D-01    -0 1327D 01    -0 1758D 00    -0 6049D-01    -0 3184D-01
 17     0 5336D 00    -0 1362D 01     0 4115D-03    -0 3809D-02     0 1109D-02
 18     0 4335D 00    -0 1362D 01    -0 2490D-02     0 2046D-02     0 8619D-02
 19     0 3292D 00    -0 1364D 01    -0 7083D-02     0 1121D-01     0 3147D-02
 20     0 2334D 00    -0 1366D 01    -0 1415D-01    -0 1375D-01    -0 7441D-02
 21     0 1657D 00    -0 1366D 01    -0 1548D-01    -0 4059D-01    -0 5442D-02
 22     0 1225D 00    -0 1367D 01    -0 1995D-01    -0 6536D-01    -0 4626D-04
 23     0 7407D-01    -0 1366D 01    -0 3025D-01    -0 1251D 00    -0 2732D-01
 24     0 3512D-01    -0 1364D 01    -0 1806D-01    -0 1245D 00    -0 2608D-01
 25     0 1146D-01    -0 1362D 01    -0 1998D-01    -0 1018D 00    -0 2995D-01
 26    -0 1226D-01    -0 1365D 01    -0 7957D-01    -0 7942D-01    -0 2951D-01
 27    -0 3365D-01    -0 1370D 01    -0 1212D 00    -0 6237D-01    -0 2385D-01
 28     0 5336D 00    -0 1381D 01     0 2073D-02    -0 5983D-02     0 1485D-02
 29     0 4335D 00    -0 1380D 01    -0 1129D-02    -0 1608D-01     0 1161D-01
 30     0 3292D 00    -0 1383D 01    -0 1036D-01    -0 2622D-01     0 4632D-02
 31     0 2333D 00    -0 1387D 01    -0 1636D-01    -0 4501D-01    -0 9544D-02
 32     0 1658D 00    -0 1388D 01    -0 1050D-01    -0 6394D-01    -0 1363D-01
 33     0 1225D 00    -0 1392D 01    -0 3483D-01    -0 9847D-01    -0 2078D-01
 34     0 8439D-01    -0 1395D 01    -0 3726D-01    -0 1659D 00    -0 4540D-01
 35     0 5423D-01    -0 1393D 01    -0 3804D-02    -0 1330D 00    -0 1023D-01
 36     0 2703D-01    -0 1391D 01    -0 1133D-01    -0 1176D 00    -0 1595D-02
 37    -0 1059D-02    -0 1401D 01    -0 6406D-01    -0 5109D-01    -0 5820D-02
 38    -0 2553D-01    -0 1417D 01    -0 1122D 00    -0 5602D-01    -0 1693D-01
 39     0 5336D 00    -0 1399D 01     0 3876D-02    -0 7609D-02     0 2493D-03
 40     0 4336D 00    -0 1398D 01     0 2527D-03    -0 3522D-01     0 1111D-01
 41     0 3292D 00    -0 1401D 01    -0 1344D-01    -0 6261D-01     0 6191D-02
 42     0 2334D 00    -0 1408D 01    -0 2005D-01    -0 7629D-01    -0 6620D-02
 43     0 1659D 00    -0 1411D 01    -0 1471D-01    -0 7595D-01    -0 2034D-01
 44     0.1224D 00    -0 1417D 01    -0 4553D-01    -0 9566D-01    -0 4975D-01
 45     0 4945D-02    -0 1441D 01    -0 6660D-01    -0 3213D-01    -0 1318D-01
 46    -0 1849D-01    -0 1465D 01    -0 8596D-01    -0 6560D-01    -0 2489D-01
 47     0 5566D 00    -0 1431D 01     0 3023D-04    -0 4335D-02    -0 9068D-04
 48     0 5199D 00    -0 1417D 01     0 6736D-02    -0 5814D-02    -0 3958D-02
 49     0 4337D 00    -0 1416D 01     0 2243D-02    -0 5588D-01     0 7080D-02
 50     0.3293D 00    -0 1420D 01    -0 1515D-01    -0 9975D-01     0 7513D-02
 51     0 2334D 00    -0 1429D 01    -0 2497D-01    -0 1103D 00     0 3044D-02
 52     0 1659D 00    -0 1437D 01    -0 3309D-01    -0 6711D-01    -0 1423D-01
 53     0 1221D 00    -0 1443D 01    -0 3383D-01    -0 2892D-01    -0 2363D-01
 54     0 7299D-02    -0 1486D 01    -0 7982D-01    -0 7602D-01    -0 3982D-01
 55    -0 1037D-01    -0 1514D 01    -0 4407D-01    -0 8437D-01    -0 7656D-02
 56     0 1616D 00    -0 1477D 01    -0 5272D-01    -0 3476D-01     0 5495D-02
 57     0 1217D 00    -0 1475D 01    -0 3341D-01    -0 1834D-01     0 2537D-02
 58     0 1500D 00    -0 1524D 01    -0 3873D-01    -0 6206D-01    -0 7179D-02
 59     0 1178D 00    -0 1503D 01    -0 2888D-01    -0 5162D-01     0 1243D-01
 60     0 9655D-01    -0 1515D 01    -0 2098D-01    -0 1025D 00     0 6591D-02
 61     0 1179D 00    -0 1544D 01    -0 2711D-01    -0 1106D 00    -0 8338D-02
 62     0 6181D-01    -0 1513D 01    -0 7056D-02    -0 1284D 00    -0 1499D-01
 63     0 6905D-01    -0 1541D 01    -0 2712D-01    -0 1225D 00    -0 1855D-01
 64     0 2747D-01    -0 1510D 01    -0 6082D-01    -0 1542D 00    -0 4486D-01
 65     0 2038D-01    -0 1539D 01    -0 4030D-01    -0 8966D-01    -0 1298D-01


          RELATIVE DISPLACEMENT AND LOCAL STRESS AT JOINT ELEMENT

ELEMENT      LOCAL X AXIS        TAN  DISP      NOR  DISP         TAU          SIGMA
```

```
 66      NODE  92  TO  NODE  91     0 3565D-03    -0 3314D-05     0 1982D-02    -0 9941D-02
 67      NODE  93  TO  NODE  92     0 1187D-02    -0 3764D-05     0 6601D-02    -0 1129D-01
 68      NODE  94  TO  NODE  93     0 1245D-02    -0 6521D-05     0 6922D-02    -0 1956D-01
 69      NODE  95  TO  NODE  94    -0 8264D-03    -0 7165D-05    -0 4595D-02    -0 2150D-01
 70      NODE  96  TO  NODE  95    -0 8234D-03    -0 2621D-04    -0 4578D-02    -0 7864D-01
 71      NODE  97  TO  NODE  96     0 1272D-02    -0 7570D-05     0 7073D-02    -0 2271D-01
 72      NODE  98  TO  NODE  97     0 1622D-02    -0 1145D-04     0 9017D-02    -0.3434D-01
 73      NODE  99  TO  NODE  98     0 3514D-03    -0 8523D-05    -0 1954D-02    -0 2557D-01
 74      NODE 100  TO  NODE  99    -0 4274D-03    -0 2822D-04    -0 2376D-02    -0 8466D-01
 75      NODE 101  TO  NODE 100    -0 7950D-03    -0 1609D-04    -0 4420D-02    -0 4827D-01
 76      NODE 102  TO  NODE 101    -0.9462D-03    -0 2098D-04    -0.5261D-02    -0 6294D-01
 77      NODE 103  TO  NODE 102    -0 6478D-03    -0 1483D-04    -0 3602D-02    -0 4450D-01
 78      NODE 104  TO  NODE 103    -0 1822D-03    -0 2639D-04    -0 1013D-02    -0 7916D-01
 79      NODE 105  TO  NODE 104    -0 2008D-03    -0 1743D-04    -0 1117D-02    -0 5228D-01
 80      NODE 106  TO  NODE 105    -0 1232D-03    -0 1853D-04    -0 6852D-03    -0 5558D-01
 81      NODE 107  TO  NODE 106    -0 3549D-04    -0 1773D-04    -0 1973D-03    -0 5319D-01



 **********************
 *  INTERFACE OUTPUT  *
 **********************



          NODAL DISPLACEMENT

NODE        U                V
 91     0 4880D 00    -0 1427D 01
 92     0 3795D 00    -0 1423D 01
 93     0 2797D 00    -0 1437D 01
 94     0 1877D 00    -0 1445D 01
 95     0 1703D 00    -0 1502D 01
 96     0 1569D 00    -0 1561D 01
 97     0 1002D 00    -0 1560D 01
 98     0 4494D-01    -0 1555D 01
 99    -0 1139D-01    -0 1552D 01
100    -0 2147D-01    -0 1500D 01
101    -0 3092D-01    -0 1448D 01
102    -0 3840D-01    -0 1396D 01
103    -0 4658D-01    -0 1347D 01
104    -0 4225D-01    -0 1306D 01
105    -0 3316D-01    -0 1260D 01
106    -0 1868D-01    -0 1220D 01
107     0 0000D 00    -0 1203D 01



 *****************
 *  SOIL  OUTPUT *
 *****************


           NODAL DISPLACEMENT
```

```
NODE        U              V
 108     0 0000D 00    -0 1042D 01
 109     0 0000D 00    -0 8039D 00
 110     0 0000D 00    -0 4658D 00
 111     0 0000D 00     0 0000D 00
 112    -0 2717D-01    -0 1057D 01
 113    -0 3234D-01    -0 8148D 00
 114    -0 3421D-01    -0 4717D 00
 115    -0 3456D-01     0 0000D 00
 116    -0 4626D-01    -0 1090D 01
 117    -0 5370D-01    -0 8389D 00
 118    -0 5681D-01    -0 4840D 00
 119    -0 5774D-01     0 0000D 00
 120    -0 5551D-01    -0 1128D 01
 121    -0 6342D-01    -0 8627D 00
 122    -0 6866D-01    -0 4956D 00
 123    -0 7052D-01     0 0000D 00
 124    -0 5458D-01    -0 1152D 01
 125    -0 6619D-01    -0 8778D 00
 126    -0 7386D-01    -0 5033D 00
 127    -0 7675D-01     0 0000D 00
 128    -0 5242D-01    -0 1163D 01
 129    -0 6684D-01    -0 8894D 00
 130    -0 7715D-01    -0 5102D 00
 131    -0 8115D-01     0 0000D 00
 132    -0 5324D-01    -0 1171D 01
 133    -0 6699D-01    -0 8967D 00
 134    -0 7863D-01    -0 5144D 00
 135    -0 8364D-01     0 0000D 00
 136     0 5644D 00    -0 1770D 01
 137     0 3406D 00    -0 1736D 01
 138     0 2135D 00    -0 1678D 01
 139    -0.5576D-01    -0 1178D 01
 140    -0 6600D-01    -0 9002D 00
 141    -0 7986D-01    -0 5169D 00
 142    -0 8514D-01     0 0000D 00
 143     0 5686D 00    -0 1992D 01
 144     0 3147D 00    -0 1975D 01
 145     0 2019D 00    -0 1873D 01
 146    -0 4592D-01    -0 1308D 01
 147    -0 6711D-01    -0 9018D 00
 148    -0 8116D-01    -0 5188D 00
 149    -0 8637D-01     0 0000D 00
 150     0 5368D 00    -0 2262D 01
 151     0 2895D 00    -0 2223D 01
 152     0 2051D 00    -0 2085D 01
 153     0 1084D 00    -0 1880D 01
 154     0 2242D-01    -0 1765D 01
 155    -0 3612D-01    -0 1644D 01
 156    -0 5372D-01    -0 1477D 01
 157    -0 5262D-01    -0 1285D 01
 158    -0 7257D-01    -0 9018D 00
 159    -0 8312D-01    -0 5205D 00
 160    -0 8764D-01     0 0000D 00
 161     0 4385D 00    -0 2705D 01
 162     0 2951D 00    -0 2538D 01
 163     0 1916D 00    -0 2360D 01
 164     0 6860D-01    -0 2100D 01
 165     0 1078D-02    -0 1944D 01
 166    -0 4669D-01    -0 1757D 01
```

```
 167    -0 6866D-01    -0 1548D 01
 168    -0 7465D-01    -0 1314D 01
 169    -0 8023D-01    -0 9140D 00
 170    -0 8529D-01    -0 5268D 00
 171    -0 8771D-01     0 0000D 00
 172     0 2889D 00    -0 2609D 01
 173     0 2660D 00    -0 2600D 01
 174     0 1714D 00    -0 2490D 01
 175     0 5823D-01    -0 2235D 01
 176     0 9310D-02    -0 2063D 01
 177    -0 2816D-01    -0 1862D 01
 178    -0 5284D-01    -0 1633D 01
 179    -0 6646D-01    -0 1379D 01
 180    -0 7487D-01    -0 9484D 00
 181    -0 7710D-01    -0 5437D 00
 182    -0 7811D-01     0 0000D 00
 183     0 1313D 00    -0 2544D 01
 184     0 1322D 00    -0 2559D 01
 185     0 9990D-01    -0 2497D 01
 186     0 3998D-01    -0 2278D 01
 187     0 1302D-01    -0 2114D 01
 188    -0 8411D-02    -0 1916D 01
 189    -0 2420D-01    -0 1686D 01
 190    -0 3487D-01    -0 1425D 01
 191    -0 4280D-01    -0 9801D 00
 192    -0 4537D-01    -0 5609D 00
 193    -0 4609D-01     0 0000D 00
 194     0 0000D 00    -0 2543D 01
 195     0 0000D 00    -0 2551D 01
 196     0 0000D 00    -0 2490D 01
 197     0 0000D 00    -0 2283D 01
 198     0 0000D 00    -0 2123D 01
 199     0 0000D 00    -0 1928D 01
 200     0 0000D 00    -0 1699D 01
 201     0 0000D 00    -0 1438D 01
 202     0 0000D 00    -0 9901D 00
 203     0 0000D 00    -0 5668D 00
 204     0 0000D 00     0 0000D 00
```

```
                  ELEMENT DISPLACEMENT AND STRESS

ELEMENT    U              V              SXX            SYY            SXY
  98    -0 1146D-01    -0 1130D 01    -0 3299D-01    -0 5621D-01    -0 2937D-03
  99    -0 1488D-01    -0 9293D 00    -0 3737D-01    -0 6285D-01    -0 3494D-03
 100    -0 1664D-01    -0 6390D 00    -0 4236D-01    -0 7133D-01    -0 2824D-03
 101    -0 1719D-01    -0 2344D 00    -0 4794D-01    -0 8146D-01    -0 1071D-03
 102    -0 3132D-01    -0 1157D 01    -0 3332D-01    -0 5744D-01    -0 8728D-03
 103    -0 3986D-01    -0 9502D 00    -0 3736D-01    -0 6404D-01    -0 9706D-03
 104    -0 4426D-01    -0 6524D 00    -0 4212D-01    -0 7240D-01    -0 7272D-03
 105    -0 4583D-01    -0 2389D 00    -0 4761D-01    -0 8236D-01    -0 2612D-03
 106    -0 4430D-01    -0 1196D 01    -0 3368D-01    -0 5951D-01    -0.1404D-02
 107    -0 5472D-01    -0 9799D 00    -0 3724D-01    -0 6602D-01    -0 1318D-02
 108    -0 6065D-01    -0 6703D 00    -0 4170D-01    -0 7387D-01    -0 8291D-03
 109    -0 6343D-01    -0 2449D 00    -0 4718D-01    -0 8355D-01    -0 2821D-03
 110    -0 4973D-01    -0 1233D 01    -0 3421D-01    -0 6278D-01    -0 1678D-02
 111    -0 5992D-01    -0 1005D 01    -0 3670D-01    -0 6776D-01    -0 9625D-03
 112    -0 6803D-01    -0 6848D 00    -0 4138D-01    -0 7510D-01    -0 5811D-03
```

```
113   -0 7245D-01   -0 2497D 00   -0 4687D-01   -0 8453D-01   -0 2058D-03
114   -0 4799D-01   -0 1264D 01   -0 3308D-01   -0 6292D-01    0 3225D-03
115   -0 6000D-01   -0 1021D 01   -0 3679D-01   -0 6860D-01   -0 4507D-05
116   -0 7101D-01   -0 6952D 00   -0 4127D-01   -0 7590D-01   -0 1949D-03
117   -0 7723D-01   -0 2534D 00   -0 4672D-01   -0 8531D-01   -0 9375D-04
118   -0 4374D-01   -0 1294D 01   -0 3289D-01   -0 6171D-01    0 6052D-03
119   -0 5987D-01   -0 1030D 01   -0 3712D-01   -0 6876D-01    0 3286D-03
120   -0 7240D-01   -0 7027D 00   -0 4137D-01   -0 7651D-01    0 1243D-03
121   -0 8014D-01   -0 2561D 00   -0 4669D-01   -0 8595D-01    0 3256D-04
122    0 4431D 00   -0 1589D 01    0 1418D-02   -0 1496D-02   -0 6216D-02
123    0 3033D 00   -0 1569D 01   -0 1379D-01   -0 1297D-01   -0 7450D-02
124    0 2128D 00   -0 1516D 01   -0 1673D-01   -0 2021D-01   -0 5811D-02
125   -0 4035D-01   -0 1324D 01   -0 3284D-01   -0 6147D-01    0 8827D-03
126   -0 6050D-01   -0 1036D 01   -0 3746D-01   -0 6927D-01    0 3407D-03
127   -0 7287D-01   -0 7071D 00   -0 4138D-01   -0 7681D-01    0 3922D-03
128   -0 8182D-01   -0 2578D 00   -0 4681D-01   -0 8642D-01    0 1153D-03
129    0 4471D 00   -0 1868D 01   -0 4991D-02   -0 5684D-02   -0 4146D-02
130    0 2677D 00   -0 1816D 01   -0 1187D-01   -0 1586D-01   -0 7936D-02
131    0 1856D 00   -0 1653D 01   -0 2001D-01   -0 3401D-01   -0 6915D-02
132   -0 3363D-01   -0 1385D 01   -0 3311D-01   -0 6139D-01    0 1266D-02
133   -0 5870D-01   -0 1072D 01   -0 3661D-01   -0 6846D-01    0 8807D-03
134   -0 7353D-01   -0 7094D 00   -0 4174D-01   -0 7705D-01    0 5331D-03
135   -0 8313D-01   -0 2589D 00   -0 4698D-01   -0 8677D-01    0 1424D-03
136    0 4274D 00   -0 2113D 01   -0 9612D-02   -0 8423D-02   -0 5321D-02
137    0 2528D 00   -0 2039D 01   -0 1361D-01   -0 2163D-01   -0 9499D-02
138    0 1681D 00   -0 1850D 01   -0 2314D-01   -0 3548D-01   -0 1332D-01
139    0 9697D-01   -0 1691D 01   -0 2363D-01   -0 2300D-01   -0 1107D-01
140    0 3287D-01   -0 1631D 01   -0 2846D-01   -0 2659D-01   -0 5246D-02
141   -0 1407D-01   -0 1557D 01   -0 2692D-01   -0 2959D-01    0 1709D-02
142   -0 4091D-01   -0 1406D 01   -0 3552D-01   -0 5785D-01    0 3788D-02
143   -0 5956D-01   -0 1099D 01   -0 3710D-01   -0 6689D-01    0 1461D-02
144   -0 7599D-01   -0 7107D 00   -0 4221D-01   -0 7717D-01    0 5195D-03
145   -0 8457D-01   -0 2598D 00   -0 4718D-01   -0 8709D-01    0 1377D-03
146    0 3900D 00   -0 2432D 01   -0 1698D-01   -0 2131D-01   -0 7075D-02
147    0 2453D 00   -0 2301D 01   -0 1491D-01   -0 2683D-01   -0 7747D-02
148    0 1434D 00   -0 2106D 01   -0 2012D-01   -0 3149D-01   -0 6378D-02
149    0 5012D-01   -0 1923D 01   -0 2337D-01   -0 3667D-01   -0 3263D-02
150   -0 1483D-01   -0 1778D 01   -0 2351D-01   -0 4014D-01   -0 2560D-02
151   -0 5130D-01   -0 1607D 01   -0 2750D-01   -0 4825D-01   -0 2408D-02
152   -0 6241D-01   -0 1406D 01   -0 3186D-01   -0 5510D-01   -0 1772D-02
153   -0 7002D-01   -0 1104D 01   -0 3775D-01   -0 6683D-01   -0 2884D-03
154   -0 8030D-01   -0 7158D 00   -0 4233D-01   -0 7752D-01    0 4611D-05
155   -0 8594D-01   -0 2618D 00   -0 4733D-01   -0 8765D-01    0 1165D-04
156    0 3221D 00   -0 2613D 01   -0 1914D-01   -0 2078D-01    0 3691D-02
157    0 2310D 00   -0 2497D 01   -0 1604D-01   -0 2571D-01    0 1037D-02
158    0 1225D 00   -0 2296D 01   -0 1930D-01   -0 3331D-01   -0 4110D-03
159    0 3430D-01   -0 2085D 01   -0 2228D-01   -0 4120D-01   -0 2839D-03
160   -0 1612D-01   -0 1907D 01   -0 2451D-01   -0 4772D-01   -0 7601D-03
161   -0 4909D-01   -0 1700D 01   -0 2742D-01   -0 5376D-01   -0 1392D-02
162   -0 6565D-01   -0 1468D 01   -0 3148D-01   -0 6046D-01   -0 1599D-02
163   -0 7405D-01   -0 1139D 01   -0 3651D-01   -0 6893D-01   -0 1169D-02
164   -0 7937D-01   -0 7332D 00   -0 4192D-01   -0 7897D-01   -0 5740D-03
165   -0 8205D-01   -0 2676D 00   -0 4707D-01   -0 8889D-01   -0 1784D-03
166    0 2046D 00   -0 2578D 01   -0 1437D-01   -0 7407D-02    0 1649D-02
167    0 1674D 00   -0 2536D 01   -0 1803D-01   -0 1990D-01    0 2837D-02
168    0 9239D-01   -0 2375D 01   -0 2052D-01   -0 3217D-01    0 1923D-02
169    0 3013D-01   -0 2172D 01   -0 2342D-01   -0 4253D-01    0 1098D-02
170   -0 3560D-02   -0 1989D 01   -0 2576D-01   -0 4938D-01    0 4861D-03
171   -0 2840D-01   -0 1774D 01   -0 2857D-01   -0 5626D-01   -0 6203D-04
172   -0 4459D-01   -0 1531D 01   -0 3174D-01   -0 6293D-01   -0 4423D-03
```

```
173   -0 5475D-01   -0 1183D 01   -0 3620D-01   -0 7144D-01   -0 5871D-03
174   -0 6004D-01   -0 7583D 00   -0 4130D-01   -0 8092D-01   -0 4545D-03
175   -0 6167D-01   -0 2762D 00   -0 4652D-01   -0 9061D-01   -0 1657D-03
176    0 6587D-01   -0 2549D 01   -0 1217D-01   -0 5166D-02    0 7980D-04
177    0 5803D-01   -0 2524D 01   -0 1720D-01   -0 1660D-01    0 7923D-03
178    0 3497D-01   -0 2387D 01   -0 2139D-01   -0 3048D-01    0 8887D-03
179    0 1325D-01   -0 2200D 01   -0 2451D-01   -0 4202D-01    0 6143D-03
180    0 1151D-02   -0 2020D 01   -0 2679D-01   -0 4946D-01    0 3672D-03
181   -0 8152D-02   -0 1807D 01   -0 2937D-01   -0 5669D-01    0 1566D-03
182   -0 1477D-01   -0 1562D 01   -0 3225D-01   -0 6381D-01   -0 3537D-06
183   -0 1942D-01   -0 1208D 01   -0 3630D-01   -0 7254D-01   -0 1155D-03
184   -0 2204D-01   -0 7745D 00   -0 4109D-01   -0 8214D-01   -0 1238D-03
185   -0 2287D-01   -0 2819D 00   -0 4617D-01   -0 9179D-01   -0 5371D-04
```

2

# VITA

Chi-an Hong

Candidate for the Degree of

Doctor of Philosophy

Thesis: STRESS ANALYSIS OF A U-FRAME STRUCTURE BY THE COUPLING OF BOUNDARY AND FINITE ELEMENT METHODS

Major Field: Civil Engineering

Biographical:

Personal Data: Born in Taipei, Taiwan, Republic of China, January 6, 1955, the son of Mr. and Mrs. Tsun-yuan Hong.

Education: Graduated from Chen-kuo Hign School, Taipei Taiwan, in July, 1973; received the Bachelor of Science degree with a major in Agricultral Engineering from National Taiwan University, in June, 1978; received the Master of Science degree with a major in Civil Engineering from Oklahoma State University, in August, 1983; completed the requirements for Doctor of Philosophy degree at Oklahoma State University, in May, 1986.

Professional Experience: Surveying Engineer, the Army of Republic of China, 1978-1980; Programmer, National Taiwan Institute of Technology, 1980-1981; Graduate Teaching Assistant, School of Civil Engineering, Oklahoma State University, 1982-date.