SHEAR ANALYSIS OF RECTANGULAR CONCRETE TANKS
CONSIDERING
INTERACTION OF PLATE ELEMENTS

by

Patrick E. Devens

Thesis submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Civil Engineering

APPROVED:

Dr. J. H. Moore, Chairman

Dr. R. M. Barker

Prof. D. A. Garst

December 1984

Blacksburg, Virginia

TABLE OF CONTENTS

		Page
ACKNOWLEDGEMENTS		iii
LIST OF FIGURES		iv
LIST OF TABLES		vi
GLOSSARY		vii
Chapter I.	INTRODUCTION AND SCOPE	1
II.	LITERATURE REVIEW	4
III.	FINITE ELEMENT APPROACH	9
IV.	FINITE ELEMENT PROGRAM DEVELOPMENT	12
V.	FINITE ELEMENT MODEL	34
VI.	FINITE ELEMENT COMPUTER PROGRAM	40
VII.	DISCUSSION OF RESULTS	45
	A. Objective	45
	B. Single Plate Comparisons	45
	C. Two Plate Comparisons	56
	D. Three Plate System	65
VIII.	CONCLUSIONS & RECOMMENDATIONS	76
IX.	BIBLIOGRAPHY	79
X.	APPENDICES	82
	Appendix 1. FE Program Documentation	83
	Appendix 2. FE Program Assumptions	84
	Appendix 3. FE Program Variable Description	85
	Appendix 4. FE Program Input Format	90
	Appendix 5. FE Program Listing	93
XI.	VITA	145

ACKNOWLEDGEMENT

The author wishes to express his appreciation to all those faculty members whose both outstanding and professional instruction has led to his increased understanding of engineering skills. Special thanks is given to Dr. J. Herbert Moore, Dr. Richard M. Barker, and Prof. Don A. Garst for their guidance and assistance throughout the author's tenure at Virginia Tech while acting as both instructors and advisors. Their combined efforts not only manifested an increased learning experience but constantly fostered those high standards associated with professional engineers.

Additional recognition and appreciation is given to Dr. S. M. Holzer, Dr. J. N. Reddy, and Dr. K. Rojiani. Through Dr. Holzer's outstanding instruction and inspiration in conjunction with Dr. Reddy's superb guidance, the author has incurred a firm foundation and great interest in the finite element method. Dr. Rojiani's endeavors in micro-computers has similarily benefited the author. It is the author's opinion that this additional knowledge alone would most certainly have made his graduate studies a most rewarding experience.

Finally, the author wishes to thank his wife, , and his children, , , and , along with both sets of grandparents for their constant support and encouragement.

LIST OF FIGURES

Figure		Page
1.	Typical Finite Element	13
2.	Polynomial From Pascal's Triangle	15
3.	Deformation Mode for Normals to the Midplane	21
4.	Element to System Relationship	31
5.	1/4 Rectangular Tank	37
6.	System Constraints	38
7.	Computer Program Flow Diagram	41
8.	Resultant on a Differential Element	46
9.	Comparison with Known Solutions; one plate, 3 sides fixed, 1 side free, triangular load	53
10.	Comparison with Known Solutions; one plate, 3 sides fixed, 1 side simply supported, triangular load	54
11.	Comparison with Known Solutions; (Fig 10 continued)	55
12.	Comparison with Known Solutions; one plate, 2 sides fixed, 1 side simply supported, 1 side free, triangular load	57
13.	Comparison with Known Solutions; one plate, 2 sides fixed, 2 sides simply supported, triangular load	58
14.	Comparison with Known Solutions; one plate, 3 sides fixed, 1 side free, triangular load, tapered wall	59
15.	Two Plate Displacement Configuration From Uniform Load	61
16.	Two Plate Displacement Configuration From Wall Dead Weight	62
17.	Two Plate Displacement Configuration From Triangular Load; top edge free, bottom edge hinged	63
18.	Two Plate Displacement Configuration From Triangular Load; top and bottom edges hinged	64

Figure | Page

19. Two Plate Shear Coefficients; top edges free bottom edges hinged 68

20. Two Plate Shear Coefficients; top and bottom edges hinged 69

21. Three Plate Displacement Configuration From Triangular Load; top edges free 70

22. Three Plate Shear Coefficients; top edges free 72

23. Three Plate Shear Coefficients; top edges free 73

24. Three Plate Shear Coefficients; top edges hinged . . . 74

25. Three PLate Shear Coefficients; top edges hinged . . . 75

LIST OF TABLES

Tables Page

1. Comparison with Known Solutions; 4 sides simply supported, uniform load 48

2. Comparison with Known Solutions; 4 sides fixed, uniform load 49

3. Comparison with Known Solutions; 4 sides simply supported, triangular load 51

4. Comparison with Known Solutions; 4 sides fixed, triangular load 52

5. Comparison with Known Solutions; two plates, top edges free, bottom edges hinged 66

6. Comparison with Known Solutions; two plates, top and bottom edges hinged 67

GLOSSARY

1, 2, 3	Reference coordinate system
C_{ij}	In-plane constants in constitutive equations
D_{ij}	Flexural rigidities (11, 12, 22); Torsional rigidities (33, 44, 55)
E_{ij}	Moduli of elasticity (11, 12, 21, 22); Shear Moduli of elasticity (33, 44, 55)
N_i	Global nodal points interpolation function
$\hat{N}_i$	Local nodal point interpolation function
N_x, N_y	In-plane normal resultant
N_{xy}, N_{yx}	In-plane shear resultant
M_x, M_y	Moment resultant
M_{xy}	Twisting moment resultant
Q_x, Q_y	Out-of-Plane shear resultant
S	Average height/width of plates being analyzed
S_x, S_y, S_z	Rotation of line element, originally perpendicular to the longitudinal plane, and about the x, y, and z axis
T	Average thickness of plates being analyzed
U	Strain energy of element
W_e	External element energy
W_i	Internal element energy
a, b	In-plane dimensions of rectangular plate
e	Subscript denoting elements
f_x, f_y, f_z	Element local loading function about x, y, or z axis
i	Subscript denoting node within an element

u, v, w	Displacements of an arbitrary point in the local frame of reference
u_o, v_o, w_o	Extensional displacements of plate midplace in x, y, and z directions
$\bar{u}$	Strain energy per unit volume
x, y, z	Reference coordinates for elements
π	Total potential energy
α_i	Constant coefficients in generalized displacement functions
γ_{xy}	In-plane shear strain
γ_{xz}, γ_{yz}	Transverse shear strains
ε_x, ε_y, ε_z	Normal components of strain of an arbitrary point in the x, y, and z direction
η, ξ, ζ	Element local coorrdinate axis
ν	Poisson's ratio
ψ_x, ψ_y	Average transverse shear strains
$[f]_e$	Element local nodal force vector
$[g]$	Global system displacement vector
$[k]_e$	Element local stiffness matrix
$[K]$	Global system stiffness matrix
$[N]$	Interpolation function matrix relating internal element displacements to surrounding nodal displacements
$[q]_e$	Local element nodal displacement vector
$[q_i]$	Local nodal displacement vector
$[Q]$	Global system force vector
$[u]_e$	Element local internal displacement vector
$[U]$	Upper triangular matrix

$[\varepsilon]$	Local element strain vector
$[\varepsilon_i]$	Local element nodal strain vector
$[\sigma]$	Local element stress vector
$[\sigma_i]$	Local element nodal stress vector
$[\ ]^T$	A vector or matrix transpose

I. INTRODUCTION AND SCOPE

Rectangular concrete tanks are generally designed by using a technique illustreated in a Portland Cement Association (PCA) bulletin (1) published in 1969. The design process requires referencing specified tables after making independent plate loading and connection assumptions. The tables provide coefficients which may be used to obtain shear and moment values for plates with various boundary conditions and various ratios of plate width to depth. These shear and moment values are subsequently used to obtain the thickness of walls and footings and the tensile reinforcement.

Since the design process is based on the previously mentioned tables, it is imperative that the provided shear and moment values adequately reflect the actual system forces. The PCA design tables do not, however, account for plate interaction when the tank bottom is constructed integrally with the sides. It appears that a classical plate method, in conjunction with superposition techniques, was followed in deriving the tables. Consequently, each side plate is apparently evaluated independently with its boundaries undergoing various combinations of clamped, free, or hinged constraints. This approach results in no transfer of moment along boundaries between side plates and the bottom plate. Therefore, the transfer of plate forces is not fully addressed. It is also noted that only moment tables are available which account for some side plate interaction whereas shear tables are not presented for the case of interplay between plates.

A more desirable/exact approach would be to accommodate plate interplay between side and bottom plates for shear and moment transfer. This can be accomplished through a finite element analysis. The finite element approach models a complete system rather than a combination of independent plates. Instead of the previously mentioned clamped, free, or hinged boundaries for independent plates, the plate connections become fixed, free, or a hinged variation dependent upon the desired construction technique between plates. The most profound difference between the classical and finite element approach is in the clamped versus fixed boundary. The fixed variation allows for boundary rotation and translation while still requiring a 90 degree angle between plates. The clamped boundary restricts both boundary rotation and translation. When constructing a monolithic wall-to-wall and/or wall-to-floor structure, the fixed boundary best represents the natural system's reaction. As for hinged plate connections, the classical approach restricts all but boundary rotation. The finite element approach allows not only boundary rotation but also allows appropriate boundary displacement synonymous with real world structural displacements resulting from loads.

This paper's purpose and scope is to obtain shear values for rectangular concrete tanks. It is accomplished by implementing the finite element approach which accounts for plate interaction. The author's preliminary assumptions are that the material is elastic, homogeneous, and isotropic. Also, it is assumed that tank walls and floor are thin plates as defined by classical plate analysis

(S/T > 20). Other appropriate assumptions will be stated and expounded upon during the finite element model's development sequence.

This paper is limited to an analysis of shear in rectangular tanks having four walls and a floor, built integrally.

II. LITERATURE REVIEW

Most rectangular concrete tank investigations have not addressed monolithically constructed tanks. They have generally assumed other than fixed plate boundary conditions which simulate a monolithically constructed system. There have ,however, been a limited number of successful attempts at deriving both deflections and moments for this type construction technique. An analysis of shear in rectangular concrete tanks having monolithic walls and floors has not been published. Even when shears have been determined for rectangular plates, the shears were obtained solely along plate boundaries. The following is a chronological review of investigations pertaining to rectangular concrete tanks.

Initial rectangular tank investigations relied upon classical plate theory and the solution of individual plate fourth order differential equations. The solution process was limited to solvable equations but obtained plate displacements, moments, boundary reactions and shears at boundary and center points based upon the plate problem being resolved. The analytical solution techniques included methods such as Levy, Naviers, and Fourier Series. Buchi (2) determined maximum vertical and horizontal bending moments in walls of open square tanks but assumed both sides and the bottom edges of walls were totally clamped. Young (3) utilized an infinite series solution based on Levy's method where his plates were completely clamped, but he only determined deflections and moments. Craemer (4) combined the above methods to determine the bending moments in box shaped reservoirs. The result was that solutions were available for only a limited number of plate boundary and loading cases. Numerous tables produced by Timoshenko and Woinowsky-Kreiger (5) and Bares (6) were derived through these methods. PCA bulletin ST 63 tables (1) reflect moment and shear values along plate edges for various plate boundary constraints ranging from the wall-base connection being hinged to being completely clamped and were developed similarly using equations given by Timoshenko (5). Upon solving the individual plate problem, the method of superposition was implemented to simulate a rectangular concrete tank system of three plates. PCA Bulletin ST 63 tables V and VI are noted as being developed in this manner.

Ghali and Lightfoot addressed the application of numerical methods to rectangular tanks in the 1950's. Ghali (7) initially utilized the finite difference approach for plates in analyzing rectangular tanks but encountered difficulties in defining boundary conditions. Preliminary work by Lightfoot (8) used the same approach on tanks having fully clamped bases to resolve bending moments. Both men, however, later pursued analogy methods since the methods better represent boundary conditions and the effects of deformations.

Lightfoot's (9) analogy method simulated a tank as a single plate having edge conditions analogous to those occurring in the tank. The plate is then treated as two-dimensional structural frame where the in-plane forces and bending moments & torques were obtained separately. The final analysis was culminated by the two systems being superimposed. Other studies included the effects of various Poisson's ratio values on plates as was investigated by Just (10) while Husain

(11) refined Lightfoot's method. Husain utilized a space-frame method similar to Lightfoot's analogy but which allowed for a one step analysis or force determination procedure.

Davies' (12 - 17) rectangular tank investigations encompass a large variety of topics. His beginning studies (12) concerned the evaluation of fixed-end-moments in a floor slab resting on an elastic foundation. The model resembled a cantilever wall which was analyzed by statics. Davies' (13) next endeavor was to determine the critical moments in long rectangular tanks which were again modeled similar to a cantilever for the middle floor moment. He used moment distribution and fixed-end-moments throughout the analysis procedure and restricted the joint boundary from rotating (clamped). In his next paper (14), he then drew upon classsical plate solutions from Timoshenko and superposition to determine the bending moment at limited locations for a monolithic wall-base connection by allowing plate interaction/rotation of the boundary. Prior to this time, investigations were limited to the assumption that the bottom edges of walls were either simply supported or fully clamped. Davies' fourth (15), fifth (16), and sixth (17) papers continued to analyzed the effects of various other support conditions on the determination of selectively placed bending moments.

The finite element method was introduced into rectangular tank analysis by Cheung and Zienkiewicz (18). They applied the finite element method to square plates on an elastic foundation in order to more easily meet boundary conditions. Their paper also covers a similar analysis utilizing square tanks. Cheung and Davies (19) expanded the above analysis to include interaction of wall-wall-base plates. They determined deflections and moments for the bottom wall edges being fully clamped and for the walls being supported by a dwarf wall around the tank edges. The wall-to-wall and wall-to-base boundaries were monolithic. This analysis was further extended to long-walled tanks by Davies (20) to evaluate bending moments in tanks having clamped wall-to-wall connections while ignoring wall-to-base interaction. Davies, Cheung, and Gorecki (21) continued this investigation by analyzing long rectangular closed tanks while Davies and Long (22) continued Cheung and Zienkiewicz's analysis. They obtained bending moments determined through superposition by solving Levy and Naviers problems for a tank supported by an elastic/winkler foundation and combined the results with those previously derived (14). Jofriet (23) then prepared bending moment tables obtained from the finite element method for tank walls having fully clamped wall-to-wall and wall-to-base boundaries. He examined the influence of nonuniform wall thicknesses on long wall bending moments. Beck (24) developed a finite element program which determined moment values for plates being simply supported along the bottom edge in accordance with the referenced PCA ST-63 tables. Fitzpatrick (25) has since refined the analysis and obtained bending moments for monolithically constructed rectangular concrete tanks.

In summary, there has been a great emphasis on determining bending moments for plates acting within a rectangular tank system. Shears have only been addressed, however, for boundaries along single plates. The purpose of this paper will be to develop a method by which the plate shears can be obtained while accounting for the plate interaction not represented in the current methods.

The analysis will draw upon the Hencky-Mindlin first-order shear deformation plate theory (29) which is analogous to the Timoshenko beam theory. The theory is an improvement of the Kirchhoff-Love theory of plates (classical plate theory). The Hencky-Mindlin theory does not assume that plane sections initially normal to the midplane remain plane and normal to the midsurface throughout deformation. The refined theory includes the effects of transverse shear deformations. It assumes plane sections initially perpendicular to the longitudinal plane of the plate remain plane but not necessarily perpendicular to the longitudinal plane.

III. FINITE ELEMENT APPROACH

The finite element method is utilized for this structural analysis problem because of its flexibility. Its approach allows for the idealization of almost any continuum into a discrete mathematical model. The model can then be subjected to varying boundary and load conditions to ascertain element nodal displacements. These displacements are then formulated into respective element forces which completes the structural analysis problem.

The process of continuum discretization is independent of the structure being evaluated. It consists of seven basic steps:

1. Discretize the Continuum: The continuum or actual structural system is systematically broken down into discrete finite elements consisting of similar nodal patterns with each node having a specified number of degrees of freedom.

2. Select the Approximation Model: Interpolation functions are selected which best describe the displacements at an element node in terms of nodal point values.

$$[u]_e = [N][q]_e \tag{1}$$

3. Derive Local Element Equilibrium Equations: Several methods of deriving conditions of equilibrium are available. The principle of virtual work is, however, utilized in this analysis due to its favorable application in structural mechanics. The equations can be broken down into the following format:

$$[f]_e = [k]_e\,[q]_e \tag{2}$$

4. Derive Global System Equilibrium Equations: The system equilibrium equations are derived from the element equilibrium equations upon forcing compatibility of displacements at nodes and equilibrium at the nodal points. The result is the following equation:

$$[Q] = [K][g] \tag{3}$$

5. Impose Boundary Conditions and Solve for System Displacements, [g].

6. Calculate Element Strains, Stresses, and Forces: This is accomplished by substituting known local displacements into previous equations such as Equation 1.

7. Analyze Results: The results of Step 6 are evaluated in terms of classical analysis and engineering expertise to determine if mathematical model alterations are required.

The above mathematical model created through a finite element analysis introduces several approximations as does any model representing a natural structure. The method's only additional approximation verses classical approaches is when interpolation functions are imposed during Step 2. This is caused by each element being connected only at its nodes. Consequently, discontinuities can develop along element boundaries which are not representative of a continuum type behavior if the functions are improperly selected. The interpolation function should account for:

1. Rigid Body Displacement: The element must be able to displace

as a rigid body without developing strains. This is the extreme case of requirement two in insuring convergence.

2. Constant Strain: Constant strain, representative of a continuum, must be obtainable as the finite element mesh is refined.

3. Internal Element Continuity: The need to maintain smooth and appropriate internal element transitions within an element's boundaries is imperative in modeling a continuum.

4. Element Interface Continuity: This requirement is imposed to avoid infinite strains and displacements at element boundaries while also avoiding element overlapping, etc., and thus insure convergence.

IV. FINITE ELEMENT PROGRAM DEVELOPMENT

A quadratic nine-node rectangular element shown in Figure 1 was selected for this model for several reasons.

1. A rectangular element is conducive to the problem boundaries. The system of three interconnected plates is readily discretized into arrays of similar rectangular elements.

2. Plate shear analysis is desired where shear is defined as:

$$Q_x = D\left(\frac{\partial^3 w_o}{\partial x^3} + \nu \frac{\partial^3 w_o}{\partial x^2 \partial y}\right) \tag{4}$$

or

$$Q_x = D_{44}\left(S_x + \frac{\partial w_o}{\partial_x}\right) \tag{5}$$

and

$$Q_y = D\left(\frac{\partial^3 w_o}{\partial y^3} + \nu \frac{\partial^3 w_o}{\partial x \partial y^2}\right) \tag{6}$$

or

$$Q_y = D_{55}\left(S_y + \frac{\partial w_o}{\partial y}\right) \tag{7}$$

The displacement polynomial must therefore accommodate a constant third or second derivative term depending upon whether Equations 4 and 6 or Equations 5 and 7 are utilized, respectively. This finite element

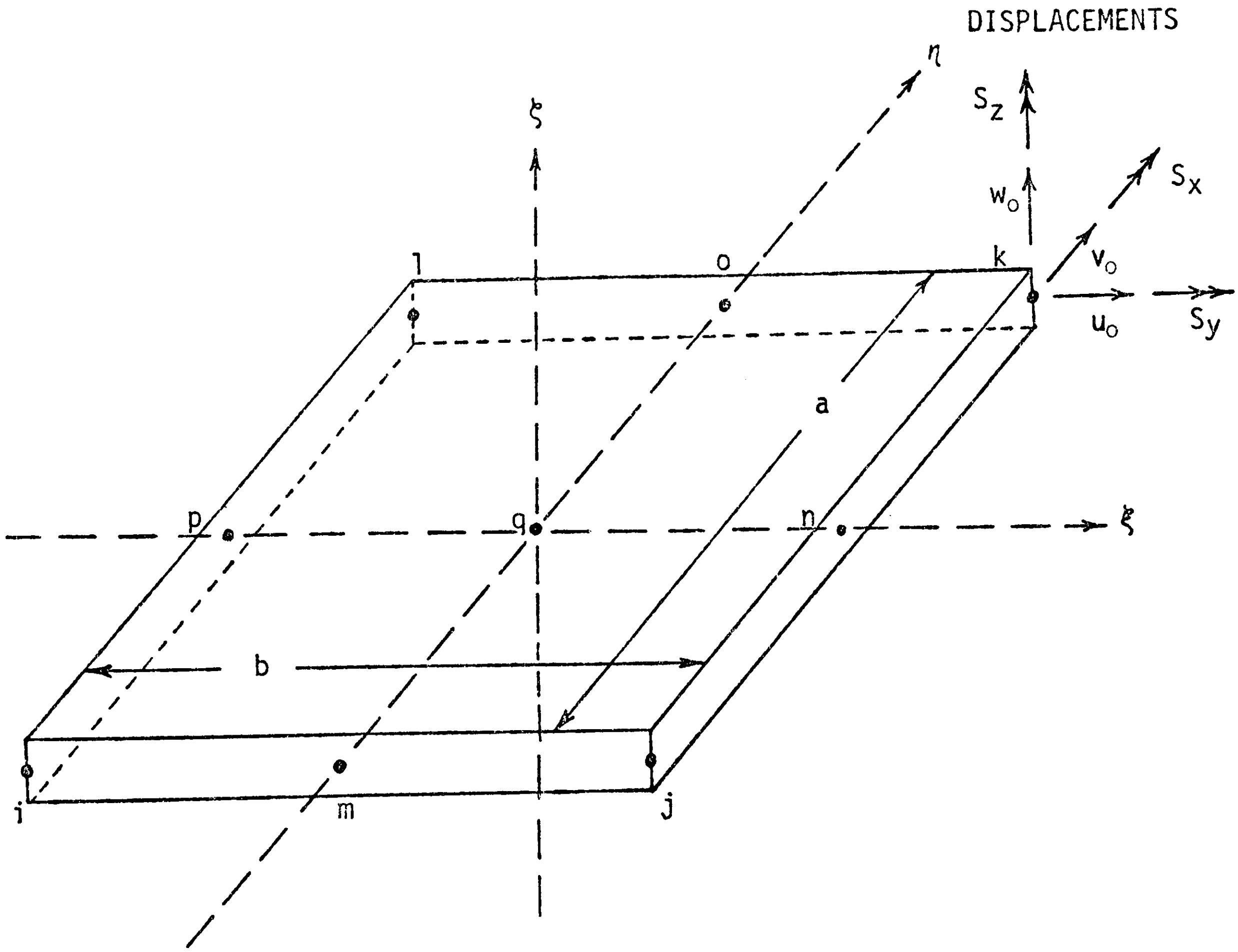


Figure 1: Typical Finite Element

model will employ the polynomials (Equations 5 and 7) associated with a Lagrange family rectangular element and its six degrees of freedom. As developed from Pascal's triangle (Figure 2), the polynominals are:

$$u_o(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 xy + \alpha_5 x^2 + \alpha_6 y^2 + \alpha_7 x^2 y + \alpha_8 xy^2 + \alpha_9 x^2 y^2 \quad (8)$$

$$v_o(x,y) = \alpha_{10} + \alpha_{11} x + \alpha_{12} y + \alpha_{13} xy + \alpha_{14} x^2 + \alpha_{15} y^2 + \alpha_{16} x^2 y + \alpha_{17} xy^2 + \alpha_{18} x^2 y^2 \quad (9)$$

$$w_o(x,y) = \alpha_{19} + \alpha_{20} x + \alpha_{21} y + \alpha_{22} xy + \alpha_{23} x^2 + \alpha_{24} y^2 + \alpha_{25} x^2 y + \alpha_{26} xy^2 + \alpha_{27} x^2 y^2 \quad (10)$$

$$S_x(x,y) = \alpha_{28} + \alpha_{29} x + \alpha_{30} y + \alpha_{31} xy + \alpha_{32} x^2 + \alpha_{33} y^2 + \alpha_{34} x^2 y + \alpha_{35} xy^2 + \alpha_{36} x^2 y^2 \quad (11)$$

$$S_y(x,y) = \alpha_{37} + \alpha_{38} x + \alpha_{39} y + \alpha_{40} xy + \alpha_{41} x^2 + \alpha_{42} y^2 + \alpha_{43} x^2 y + \alpha_{44} xy^2 + \alpha_{45} x^2 y^2 \quad (12)$$

$$S_z(x,y) = \alpha_{46} + \alpha_{47} x + \alpha_{48} y + \alpha_{49} xy + \alpha_{50} x^2 + \alpha_{51} y^2 + \alpha_{52} x^2 y + \alpha_{53} xy^2 + \alpha_{54} x^2 y^2 \quad (13)$$

Consequently, appropriate terms are available for shear determination.

3. The center node precludes abnormal element stiffening called "locking" when element thickness to length ratios become small.

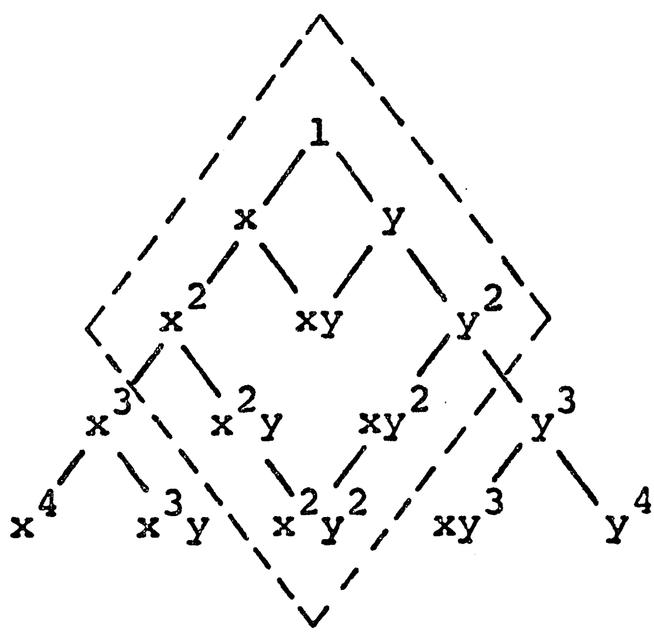


Figure 2: Polynomial from Pascal's Triangle
(Lagrange Rectangular Element)

4. Experience has favored nine-node element utilization for plate analysis (32).

The selected polynomials also satisfy the previously established displacement function criteria. Each polynomial's initial constant insures rigid body displacement while the second term insures the constant strain requirement. The internal element continuity stipulation is accommodated by each polynomial expression being uniquely defined at each element's node. As for element interface continuity, nodal displacement continuity along element boundaries is imposed during the finite element procedure which insures convergence with mesh refinement.

The selected rectangular element (see Figure 1) will have nine-nodes with six degrees of freedom per node resulting in a 54 degree of freedom element. The local element displacement vector, $[q]_e$, can then be defined as:

$$[q]_e = \begin{bmatrix} q_i \\ q_j \\ q_k \\ q_l \\ q_m \\ q_n \\ q_o \\ q_p \\ q_q \end{bmatrix} \qquad (14)$$

where $[q_i]$ through $[q_q]$ represent each node's similarly formed displacement vector. These nodal displacement written as:

$$[q_i] = \begin{bmatrix} u_i \\ v_i \\ w_i \\ Sx_i \\ Sy_i \\ Sz_i \end{bmatrix} \tag{15}$$

which can be futher subdivided into three areas. They are an inplane displacement vector:

$$[q_i]^p = \begin{bmatrix} u_i \\ v_i \end{bmatrix} \tag{16}$$

a plate bending displacement vector:

$$[q_i]^b = \begin{bmatrix} w_i \\ Sx_i \\ Sy_i \end{bmatrix} \tag{17}$$

and an in-plane twisting displacement vector:

$$[q_i]^t = \begin{bmatrix} Sz_i \end{bmatrix} \tag{18}$$

Upon performing "III. Finite Element Approach" steps 1 and 2, the internal element displacements in terms of nodal displacements (Equation 1) or interpolation functions (26) may be written as:

$$N_1 = (1 - \frac{2x}{a})(1 - \frac{x}{a})(1 - \frac{2y}{b})(1 - \frac{y}{b})$$

$$N_2 = \frac{x}{a}(\frac{2x}{a} - 1)(1 - \frac{2y}{b})(1 - \frac{y}{b})$$

$$N_3 = \frac{x}{a}(\frac{2x}{a} - 1)\frac{y}{b}(\frac{2y}{b} - 1)$$

$$N_4 = (1 - \frac{2x}{a})(1 - \frac{x}{b})\frac{y}{b}(\frac{2y}{b} - 1)$$

$$N_5 = \frac{4x}{a}(1 - \frac{x}{a})(1 - \frac{2y}{b})(1 - \frac{y}{b}) \qquad (19)$$

$$N_6 = \frac{x}{a}(\frac{2x}{a} - 1)\frac{4y}{b}(1 - \frac{y}{b})$$

$$N_7 = \frac{4x}{a}(1 - \frac{x}{a})\frac{y}{b}(\frac{2y}{b} - 1)$$

$$N_8 = (1 - \frac{2x}{a})(1 - \frac{x}{a})\frac{4y}{a}(1 - \frac{y}{b})$$

$$N_9 = \frac{4x}{a}(1 - \frac{x}{a})\frac{4y}{b}(1 - \frac{y}{b})$$

when referenced in an x, y global coordinate system. These equations are, however, transfered into a local coordinate system to increase analysis efficiency. The local coordinate system interpolation functions (26) are:

$$\hat{N}_1 = \frac{1}{4}(1 - \xi)(1 - \eta)\xi\eta$$

$$\hat{N}_2 = -\frac{1}{4}(1 + \xi)(1 - \eta)\xi\eta$$

$$\hat{N}_3 = \frac{1}{4}(1 + \xi)(1 + \eta)\xi\eta$$

$$\hat{N}_4 = -\frac{1}{4}(1 - \xi)(1 + \eta)\xi\eta$$

$$\hat{N}_5 = -\frac{1}{2}(1 - \xi^2)(1 - \eta)\eta \tag{20}$$

$$\hat{N}_6 = \frac{1}{2}(1 + \xi)(1 - \eta^2)\xi$$

$$\hat{N}_7 = \frac{1}{2}(1 - \xi^2)(1 + \eta)\eta$$

$$\hat{N}_8 = \frac{1}{2}(1 - \xi)(1 - \eta^2)\xi$$

$$\hat{N}_9 = (1 - \xi^2)(1 - \eta^2)$$

when

$$\xi = \frac{2(x - x_1) - a}{a} \tag{21}$$

and

$$\eta = \frac{2(y - y_1) - b}{b} \tag{22}$$

where

x_1 = x coordinate of node being addressed

y_1 = y coordinate of node being addressed.

Strain-displacement relationships are developed in accordance with the Hencky-Mindlin first-order shear deformation theory. The theory varies from the classical (Kirchhoff-Love) plate theory in that plane sections originally perpendicular to the longitudinal plane of the plate remain plane, but not necessarily perpendicular to the longitudinal plane. The effect is an additional displacement associated with rotation displayed in Figure 3 (30) where:

$$S_x(x,y) = \frac{\partial w}{\partial x} + \psi_x(x,y) \tag{23}$$

$$S_y(x,y) = \frac{\partial w}{\partial y} + \psi_y(x,y) \tag{24}$$

By making the following assumptions;

1. S_x and S_y are assumed small.

2. S_z is negligible due to in-plane rotation being assumed negligible.

3. $\varepsilon_z = \frac{\partial w}{\partial z}$ is assumed negligible since transverse displacement, w, is small and assumed the same for points normal to the midplane.

4. u and v are assumed to vary linearly through the plate thickness.

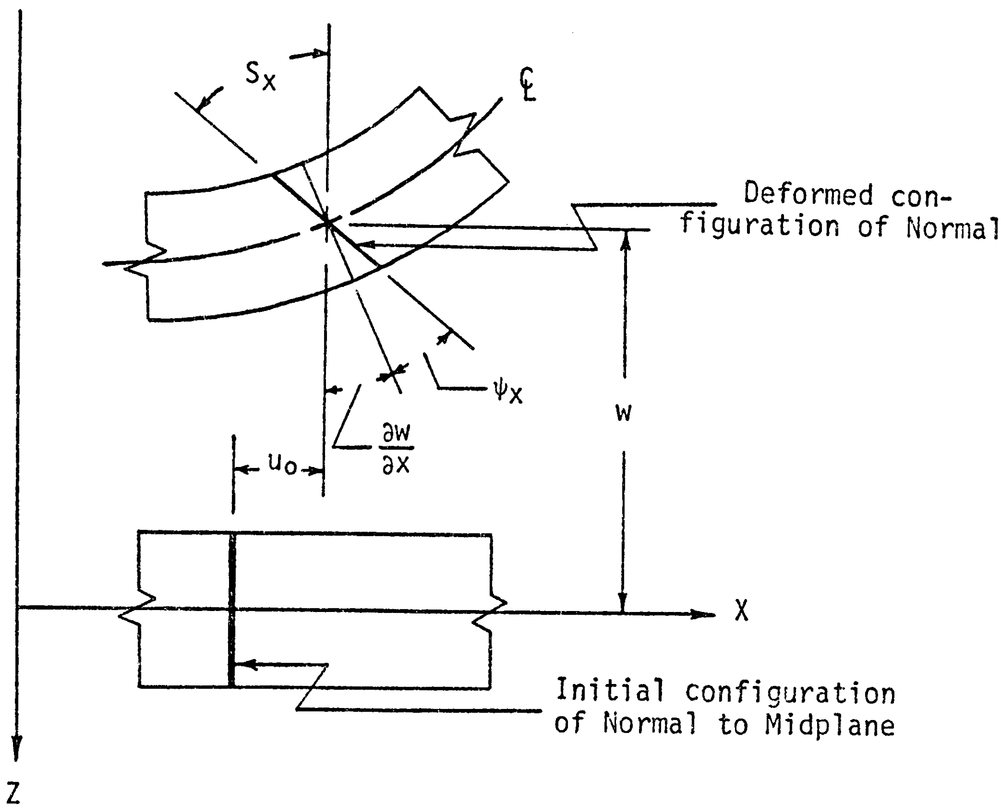


Figure 3: Deformation Mode for Normals to the Midplane

a plate point displacement field can be represented by:

$$u(x,y,z) = u_o(x,y) + zS_x(x,y) \tag{25}$$

$$v(x,y,z) = v_o(x,y) + zS_y(x,y) \tag{26}$$

$$w(x,y,z) = w_o(x,y) \tag{27}$$

from which the strain-displacement relationships may be written:

$$\varepsilon_x = \frac{\partial u}{\partial x} = \frac{\partial u_o}{\partial x} + z\frac{\partial S_x}{\partial x} \tag{28}$$

$$\varepsilon_y = \frac{\partial v}{\partial y} = \frac{\partial v_o}{\partial y} + z\frac{\partial S_y}{\partial y} \tag{29}$$

$$\varepsilon_z = \frac{\partial w}{\partial z} = 0 \tag{30}$$

$$\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} = \frac{\partial u_o}{\partial y} + \frac{\partial v_o}{\partial x} + z\left(\frac{\partial S_x}{\partial y} + \frac{\partial S_y}{\partial x}\right) \tag{31}$$

$$\gamma_{xz} = \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} = S_x + \frac{\partial w_o}{\partial x} \tag{32}$$

$$\gamma_{yz} = \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} = S_y + \frac{\partial w_o}{\partial y} \tag{33}$$

These relationships may be placed in matrix form as:

$$[\varepsilon_i] = \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{xz} \\ \gamma_{yz} \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 & z\frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 & 0 & z\frac{\partial}{\partial y} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 & z\frac{\partial}{\partial y} & z\frac{\partial}{\partial x} & 0 \\ 0 & 0 & \frac{\partial}{\partial x} & 1 & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial y} & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u_o \\ v_o \\ w_o \\ S_x \\ S_y \\ S_z \end{bmatrix} \qquad (34)$$

Stress relationships can now be derived utilizing appropriate constitutive equations in accordance with Hooke's Law. This matrix relationship for an isotropic body can be written as:

$$[\sigma_i] = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{xz} \\ \tau_{yz} \end{bmatrix} = \begin{bmatrix} E_{11} & E_{12} & 0 & 0 & 0 & 0 \\ E_{21} & E_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & E_{33} & 0 & 0 \\ 0 & 0 & 0 & 0 & E_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & E_{55} \end{bmatrix} \begin{bmatrix} \frac{\partial u_o}{\partial x} + z\frac{\partial S_x}{\partial x} \\ \frac{\partial v_o}{\partial y} + z\frac{\partial S_y}{\partial y} \\ 0 \\ \frac{\partial u_o}{\partial y} + \frac{\partial v_o}{\partial x} + z(\frac{\partial S_x}{\partial y} + \frac{\partial S_y}{\partial x}) \\ S_x + \frac{\partial w_o}{\partial x} \\ S_y + \frac{\partial w_o}{\partial y} \end{bmatrix}$$

(35)

where E_{11}, E_{12}, E_{21}, and E_{22} are moduli of elasticity defined by:

$$E_{11} = \frac{E_x}{1 - \nu_x\nu_y} \tag{36}$$

$$E_{12} = E_{21} = \frac{E_x\nu_y}{1 - \nu_x\nu_y} \tag{37}$$

$$E_{22} = \frac{E_y}{1 - \nu_x\nu_y} \tag{38}$$

and E_{33}, E_{44}, and E_{55} are shear moduli of elasticity defined by:

$$E_{33} = \frac{E_x}{2(1 + \nu_x)} \tag{39}$$

$$E_{44} = \frac{E_y}{2(1 + \nu_y)} \tag{40}$$

$$E_{55} = \frac{E_z}{2(1 + \nu_z)} \tag{41}$$

where $E_x = E_y = E_z$ and $\nu_x = \nu_y = \nu_z$ due to initial elastic, homogeneous, and isotropic material property assumptions.

The element's total potential energy, π, is the sum of all external, W_e, and internal, W_i, energy;

$$\pi = W_e + W_i \tag{42}$$

The external potential energy, W_e, expression is;

$$W_e = \int_v [f_x u + f_y v + f_z w] dv \tag{43}$$

for this model since no external moments will be applied. (Note subscript "o" will be dropped from this point forward.) The element internal energy, W_i, is equal to the negative of the strain energy, U.

$$U = \int_v \bar{u} dv \tag{44}$$

where the strain energy per unit volume for a linear case is

$$\bar{u} = \frac{1}{2}[\varepsilon]^T[\sigma] \tag{45}$$

Substituting Equation 45 into Equation 44 and then both 44 and 43 into Equation 42 gives:

$$\begin{aligned}\pi = \frac{1}{2}\int_v \{ & E_{11}\left(\frac{\partial u}{\partial x} + z\frac{\partial S_x}{\partial x}\right)^2 + 2E_{12}\left(\frac{\partial u}{\partial x} + z\frac{\partial S_x}{\partial x}\right)\left(\frac{\partial v}{\partial y} + z\frac{\partial S_y}{\partial y}\right) \\ & + E_{22}\left(\frac{\partial v}{\partial y} + z\frac{\partial S_y}{\partial y}\right)^2 + E_{33}\left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} + z\left(\frac{\partial S_x}{\partial y} + \frac{\partial S_y}{\partial x}\right)\right]^2 \\ & + E_{44}\left(S_x + \frac{\partial w}{\partial x}\right)^2 + E_{55}\left(S_y + \frac{\partial w}{\partial y}\right)^2\} dv + \int_v [f_x u + f_y v + f_z w] dv\end{aligned} \tag{46}$$

By applying the principle of virtual work, a set of equilibrium conditions is determined for the element. The principle states that virtual work of all forces, external and internal, vanishes for any virtual displacement.

$$\delta\pi = \delta W_e - \delta U = 0 \tag{47}$$

The variation with regard to displacements (u, v, w, S_x, S_y) is therefore taken from Equation 46 which is then integrated by parts before grouping terms (δu, δv, δw, δS_x, δS_y) and setting each equal to zero. This gives five equilibrium conditions:

$$\frac{\partial}{\partial x}\left(C_{11}\frac{\partial u}{\partial x} + C_{12}\frac{\partial v}{\partial y}\right) + C_{33}\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + f_x = 0 \tag{48}$$

$$C_{33}\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) + \frac{\partial}{\partial x}\left(C_{12}\frac{\partial u}{\partial x} + C_{22}\frac{\partial v}{\partial y}\right) + f_y = 0 \tag{49}$$

$$D_{44}\frac{\partial}{\partial x}\left(S_x + \frac{\partial w}{\partial x}\right) + D_{55}\frac{\partial}{\partial y}\left(S_y + \frac{\partial w}{\partial y}\right) + f_z = 0 \tag{50}$$

$$\frac{\partial}{\partial x}\left(D_{11}\frac{\partial S_x}{\partial x} + D_{12}\frac{\partial S_y}{\partial y}\right) + D_{33}\frac{\partial}{\partial y}\left(\frac{\partial S_x}{\partial y} + \frac{\partial S_y}{\partial x}\right) - D_{44}\left(S_x + \frac{\partial w}{\partial x}\right) = 0 \tag{51}$$

$$D_{33}\frac{\partial}{\partial x}\left(\frac{\partial S_x}{\partial y} + \frac{\partial S_y}{\partial x}\right) + \frac{\partial}{\partial y}\left(D_{12}\frac{\partial S_x}{\partial x} + D_{22}\frac{\partial S_y}{\partial y}\right) - D_{55}\left(S_y + \frac{\partial w}{\partial y}\right) = 0 \tag{52}$$

where

$$C_{ij} = t \cdot E_{ij} \tag{53}$$

and

$$D_{ij} = \frac{t^3}{12} \cdot E_{ij} \tag{54}$$

Upon substituting the interpolation functions (Equations 8 to 13) into the five equilibrium equations, Equations 48 through 52 may be written as:

$$[k_{11}]\{u\} + [k_{12}]\{v\} = \{f_1\} \tag{55}$$

$$[k_{21}]\{u\} + [k_{22}]\{v\} = \{f_2\} \tag{56}$$

$$[k_{33}]\{w\} + [k_{34}]\{S_x\} + [k_{35}]\{S_y\} = \{f_3\} \tag{57}$$

$$[k_{43}]\{w\} + [k_{44}]\{S_x\} + [k_{45}]\{S_y\} = 0 \tag{58}$$

$$[k_{53}]\{w\} + [k_{54}]\{S_x\} + [k_{55}]\{S_y\} = 0 \tag{59}$$

where

$$k_{ij}^{11} = \int_v (C_{11}\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + C_{33}\frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y})dv \tag{60}$$

$$k_{ij}^{12} = k_{ij}^{21} = \int_v (C_{12}\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial y} + C_{33}\frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial x})dv \tag{61}$$

$$k_{ij}^{22} = \int_v (C_{33}\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + C_{22}\frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y})dv \tag{62}$$

$$k_{ij}^{33} = \int_v (D_{44}\frac{\partial N_i}{\partial x}\frac{\partial N_i}{\partial x} + D_{55}\frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y})dv \tag{63}$$

$$k_{ij}^{34} = k_{ij}^{43} = \int_v D_{44}\frac{\partial N_i}{\partial x}N_j dv \tag{64}$$

$$k_{ij}^{35} = k_{ij}^{53} = \int_v D_{55}\frac{\partial N_i}{\partial y}N_j dv \tag{65}$$

$$k_{ij}^{44} = \int_v (D_{11}\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + D_{33}\frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y} + D_{44}N_iN_j)dv \tag{66}$$

$$k_{ij}^{45} = k_{ij}^{54} = \int_v (D_{12}\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial y} + D_{33}\frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial x})dv \tag{67}$$

$$k_{ij}^{55} = \int_v (D_{33}\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + D_{22}\frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y} + D_{55}N_iN_j)dv \tag{68}$$

$$f_i^1 = \int_A N_i f_x dA \tag{69}$$

$$f_i^2 = \int_A N_i f_y dA \tag{70}$$

$$f_i^3 = \int_A N_i f_z dA \tag{71}$$

The matrix form is:

$$\begin{bmatrix} k_{11} & k_{12} & 0 & 0 & 0 & 0 \\ k_{21} & k_{22} & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{33} & k_{34} & k_{35} & 0 \\ 0 & 0 & k_{43} & k_{44} & k_{45} & 0 \\ 0 & 0 & k_{53} & k_{54} & k_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & k_{66} \end{bmatrix} \begin{bmatrix} u \\ u \\ w \\ S_x \\ S_y \\ S_z \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{72}$$

where k_{11} through k_{22} represents in-plane stiffnesses and k_{33} through k_{55} represent flexural stiffnesses. The equation

$$[k_{66}]\{S_z\} = 0 \tag{73}$$

is inserted to accommodate the required interplay of stiffnesses, displacements, and forces between the three system's plates being modeled.

By evaluating each element at its nodes, the element stiffness matrix (54 x 54),

$$[k]_e = \begin{bmatrix} k_{ii} & k_{ij} & k_{ik} & k_{il} & k_{im} & k_{in} & k_{io} & k_{ip} & k_{iq} \\ k_{ji} & k_{jj} & k_{jk} & k_{jl} & k_{jm} & k_{jn} & k_{jo} & k_{jp} & k_{jq} \\ k_{ki} & k_{kj} & k_{kk} & k_{kl} & k_{km} & k_{kn} & k_{ko} & k_{kp} & k_{kq} \\ k_{li} & k_{lj} & k_{lk} & k_{ll} & k_{lm} & k_{ln} & k_{lo} & k_{lp} & k_{lq} \\ k_{mi} & k_{mj} & k_{mk} & k_{ml} & k_{mm} & k_{mn} & k_{mo} & k_{mp} & k_{mq} \\ k_{ni} & k_{nj} & k_{nk} & k_{nl} & k_{nm} & k_{nn} & k_{no} & k_{np} & k_{nq} \\ k_{oi} & k_{oj} & k_{ok} & k_{ol} & k_{om} & k_{on} & k_{oo} & k_{op} & k_{oq} \\ k_{pi} & k_{pj} & k_{pk} & k_{pl} & k_{pm} & k_{pn} & k_{po} & k_{pp} & k_{pq} \\ k_{qi} & k_{qj} & k_{qk} & k_{ql} & k_{qm} & k_{qn} & k_{qo} & k_{qp} & k_{qq} \end{bmatrix} \tag{74}$$

with each submatrix k_{ii} being similar to Equation 72, and element force vector (1 × 54) are derived.

These unique local element stiffness matrices and force vectors are compiled to form the global system stiffness and force vector after first being appropriately constrained and transferred into a global sense. Additional boundary options are available as program input and reflected in the system JCODE by neglecting (setting equal to zero) the respective degree of freedom associated with a constraint. Upon delineating element nodal numbering sequences (see Figure 4 following right-hand-rule) in accordance with their local and global references, the system MCODE is constituted from the JCODE data. The MCODE relationships are then imposed upon element matrices to derive each element's contribution to the system. The contributions are subsequently added to manifest the system stiffness matrix and force vector.

The result is Equation 3 which is solved by a Linpack (29) equation solver. The solution technique (Cholesky method (29)) incorporates forward substitution by letting

$$[K] = [U]^T[U] \tag{75}$$

and

$$[Y] = [U][g] \tag{76}$$

leading to

$$[U]^T[Y] = [Q] \tag{77}$$

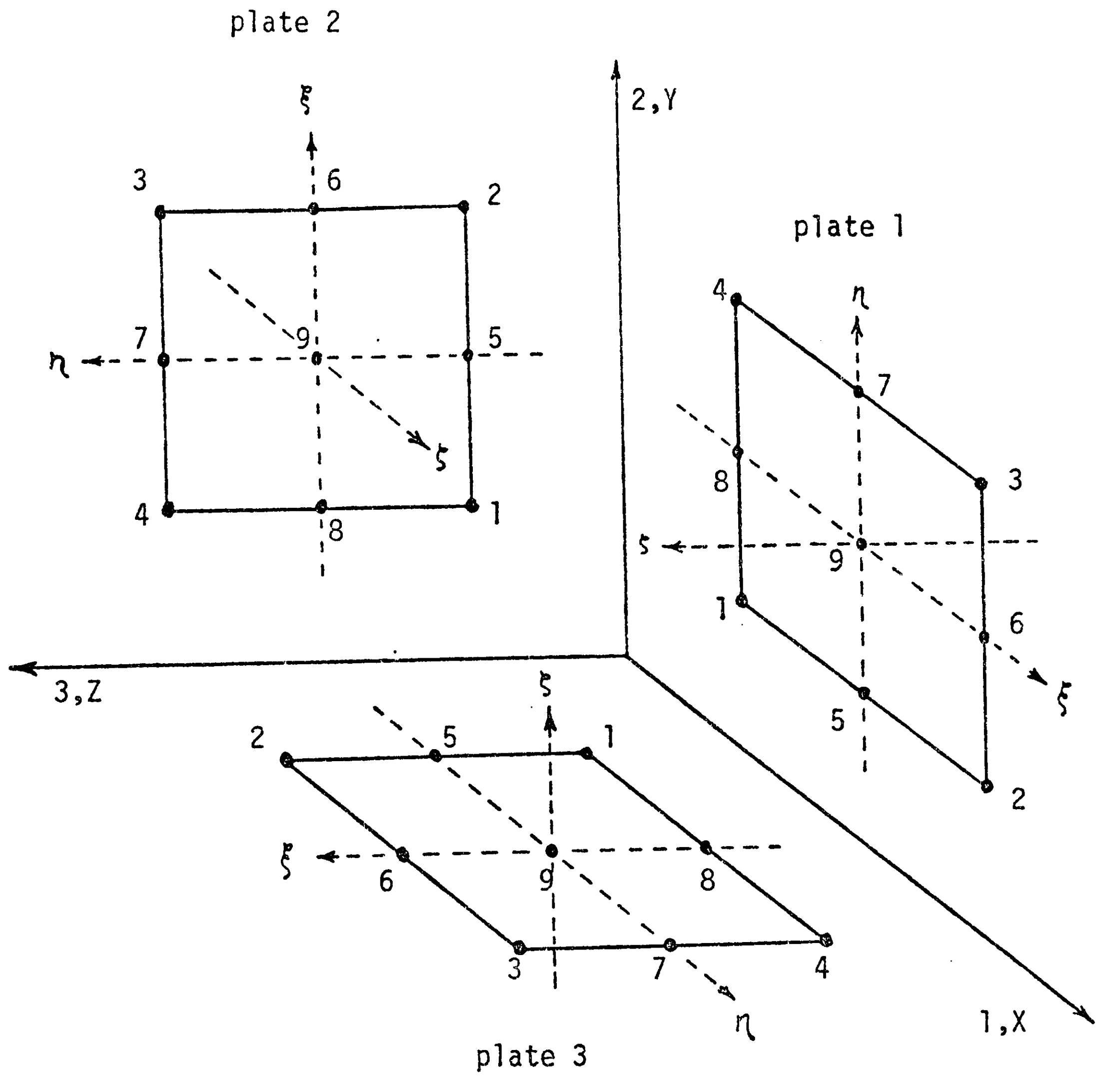


Figure 4: Element to System Relationship

where [Y] is then solved for. Next, the equation solver determines the global displacements, [g], by backward substitution

$$[U][g] = [Y] \tag{78}$$

which are transformed into local displacements by MCODE relationships.

From the initial equilibrium equations and classical plate analysis, the in-plane (x & y) normal forces are:

$$N_x = (C_{11}\frac{\partial u_o}{\partial x} + C_{12}\frac{\partial v_o}{\partial y})\cdot T_1 \tag{79}$$

$$N_y = (C_{12}\frac{\partial u_o}{\partial y} + C_{22}\frac{\partial v_o}{\partial y})\cdot T_2 \tag{80}$$

The in-plane shear forces are:

$$N_{xy} = N_{yx} = C_{33}(\frac{\partial u_o}{\partial y} + \frac{\partial v_o}{\partial x})\cdot T_1 \tag{81}$$

The moment resultants are:

$$M_x = D_{11}\frac{\partial S_x}{\partial x} + D_{12}\frac{\partial S_y}{\partial y} \tag{82}$$

$$M_y = D_{12}\frac{\partial S_x}{\partial x} + D_{22}\frac{\partial S_y}{\partial y} \tag{83}$$

$$M_{xy} = D_{33}(\frac{\partial S_x}{\partial y} + \frac{\partial S_y}{\partial x}) \tag{84}$$

where the partials of S_x and S_y are obtained by differentiating Equations 11 through 13. The out-of-plane shear resultants are:

$$Q_x = D_{44}(S_x + \frac{\partial w_o}{\partial x}) \tag{85}$$

$$Q_y = D_{55}(S_y + \frac{\partial w_o}{\partial y}) \tag{86}$$

Therefore by substituting now known element local displacements, the above element forces are determined.

This completes the finite element analysis of the system and its elements. The process began by discretizing the structure and resolving the global and local displacements based upon the system stiffness and loads. From this point, desired forces were determined.

V. FINITE ELEMENT MODEL

The finite element method is a numerical approximation process. Its objective is to simulate a continuum which is subsequently analyzed. The model must, therefore, represent as many system properties as possible to accommodate a more exact approximation. These properties include both system material and structural characteristics. It also includes appropriate structural loading conditions. The present element model is an expanded version of the model presented by Fitzpatrick (25) in his graduate thesis presented at Virginia Polytechnic Institute and State University.

Material characteristics are accounted for in two ways. The concrete is assumed isotropic and elastic. The initial assumption is reasonable since concrete mixes tend to have random axial characteristics. Consequently, if an averaging process were performed over each element and the system, the results would manifest isotropic trends. The elastic behavior assumption is based upon the system undergoing only small strains. This assumption can be attributed to the requirement for negligible cracking in rectangular concrete tanks to control leakage or the desire for negligible plate deformation in the system. The model implements this approximation by requiring a single Modulus of Elasticity (Young's Modulus) for the tank's construction material as program input.

The tank's continuum behavior is simulated in numerous ways. A common construction technique of placing monolithic concrete wall-to-wall and wall-to-floor systems is imitated. This is accomplished by preserving continuity along plate boundaries. The respective degrees of freedom can, therefore, transfer appropriate forces from plate to plate as occurs in the continuum. The model's top edges are allowed total flexibility associated with the unconstrained actual system's top edges. The interaction between the rectangular tank and its supporting soil is also represented, as described in Fitzpatrick's thesis (25). The tank is assumed to be supported by a homogeneous soil which reacts with uniform pressure to tank loads. This assumption manifests uniform tank settlement from which vertical displacements are then determined. Finally, each plate's in-plane rotation is assummed negligible. The assumption is inherent in classical plate theory since the in-plane rotation is assumed negligible as compared with other displacements.

Loadings on the continuum are defined by three modeling parameters. They are load type, direction, and location. The load types are triangular, uniform, and gravitational. The triangular loads are representative of hydrostatic forces on the structure from contained liquid as well as possible water table forces encountered by normally buried tanks. Triangular loadings also result from idealized earth pressures on buried tanks. The uniform loads imitate surcharge forces. Gravitational or vertical wall dead weight loads are simulated by a base strip load which is transferred through the base plate at a 2.5 to 1 slope. This procedure parallels the current AISC steel code (32) and tends to manifest a more realistic transfer of wall loads to the base plate. Each condition may be imposed separately or in conjunction with others. The loading location and direction are put into the program by noting the plate number and its local coordinate system imposed load direction. All imposed loads must be, however, perpendicular to the loaded plate. The result is an ability to more closely represent the loadings on a rectangular concrete tank.

The inputed plate loadings are individually addressed and transformed into nodal loads through a three step procedure. First, a plate load is subdivided into element loads. Second, the element load is averaged over its area and equated to an element uniform load. This step's assumption is that finite element mesh refinement will foster the actual loading condition. Third, the element uniform load is distributed to nodal loads using an isoparametric evaluation. A final element nodal load is then obtained by summing its nodal load contribution from each loading condition.

In order to develop a more efficient model, only a quarter of a rectangular tank is analyzed (see Figure 5). This is possible because of double symmetry. It is assumed that each quarter tank will react in a similar manner if appropriately constrained along divisional boundaries and if tank loadings are symmetric. These constraints (see Figure 6) include nodal normal and rotational displacements which are in-plane and perpendicular to the division lines. Due to a symmetric structure under symmetric loading, the center line normal and rotational (slope) in-plane displacements will both be zero.

For user information, the local and global axes are displayed in Figure 4. Plates 1, 2, and 3 are in the X-Y, Y-Z, and X-Z planes, respectively (see Figure 5). The element numbering sequence begins in

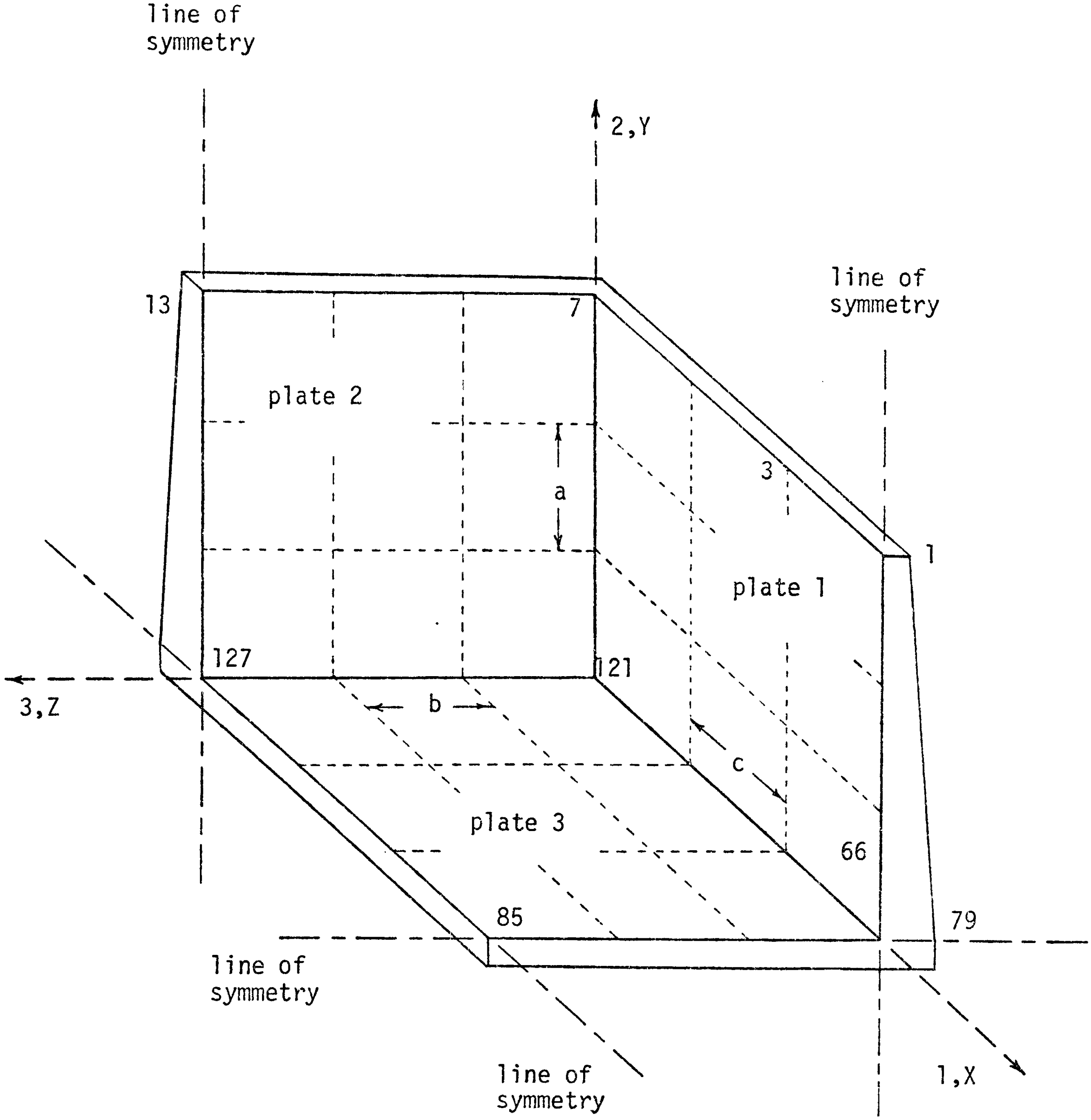


Figure 5: 1/4 Rectangular Tank

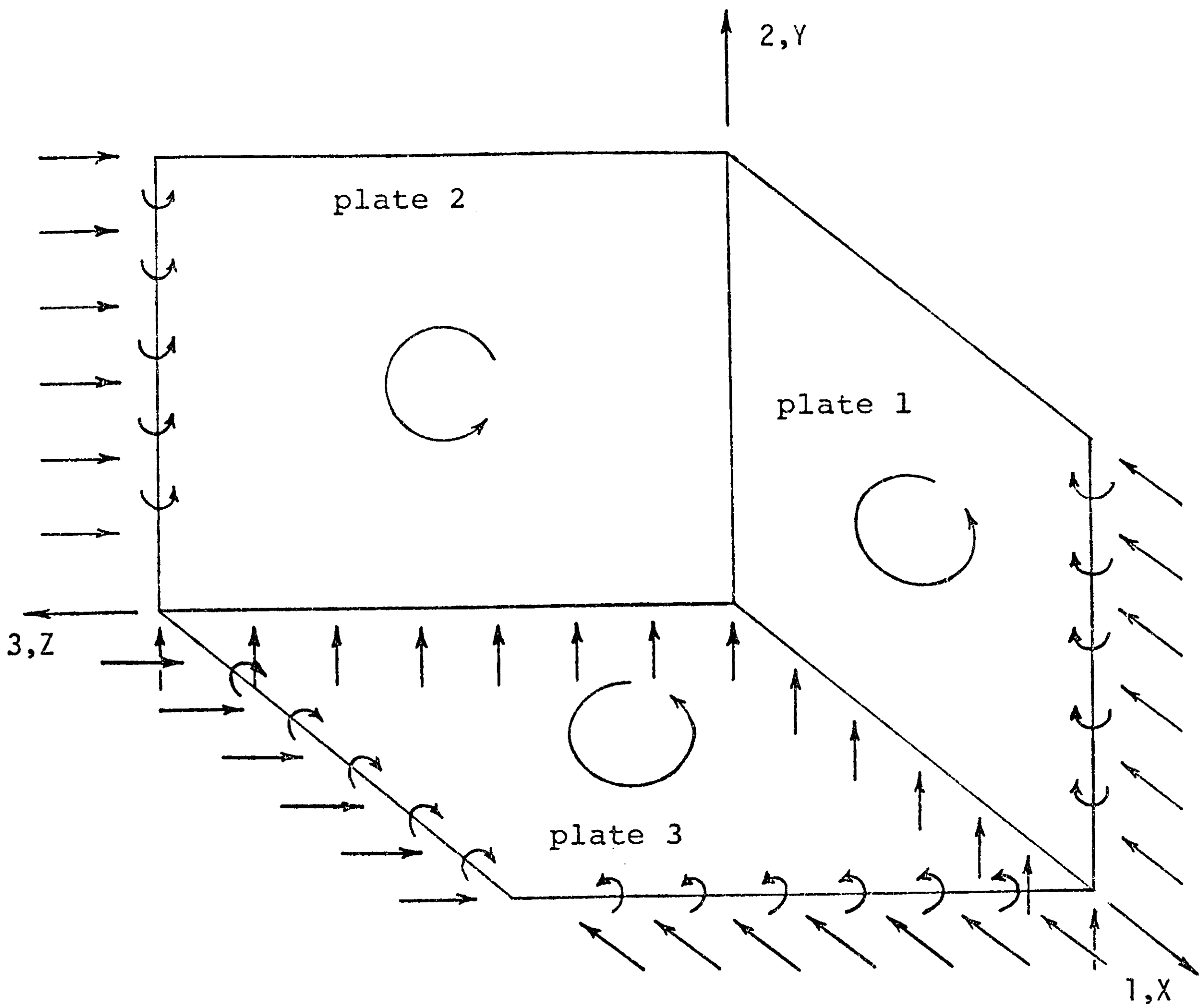


Figure 6: System Constraints

the upper right-hand corner of plate 1 and extends horizontally until all row plate 1 and plate 2 elements are numbered before dropping to the right most element of the next horizontal row and continuing as before. This sequence is repeated until all plate 1 and 2 elements are numbered. Plate 3 elements are then numbered. This is accomplished by numbering all elements in sequence. The process begins with the element having the largest X and smallest Z global coordinate values and proceeds along a constant X axis until all plate 3 elements are numbered before decreasing the X value and continuing as before until all elements are numbered (see Figure 5). The nodal numbering procedure follows the same sequence utilized for element numbering except that those nodes located on plate boundaries will receive their nodal numbers as if it were a floor node rather than a wall node (see Figure 5).

VI. FINITE ELEMENT COMPUTER PROGRAM

This finite element program encompasses several logic sequences utilized by Brenneman (27), Beck (24), Fitzpatrick (25), and Reddy (26) while addressing a separate objective. All computational steps required to analyze various rectangular tanks by size and load are accomplished through program subroutines. A description of each subroutine's function is outlined below, while Figure 7 depicts the sequence and relation of the routines. Necessary input is delineated in Appendix 1, at the program's beginning. The program input and output is in kips, inches, and radians.

MAIN - An executive routine whose purpose is to supervise the sequential execution of appropriate subroutines.

DATA - Reads in, checks, and echoes input data (plate dimensions, element meshes, plate thicknesses, and material properties). Calls subroutine THICK.

THICK - Computes average element plate thicknesses.

GEN - Calculates the number of nodes in the system. Generates and prints both element and nodal numbers for one to three plates.

GLOBXY - Generates relationships between elements and their respective nodes while coorelating each node to a global reference frame.

PROCESS - Generates and prints (JCODE) automatically and user

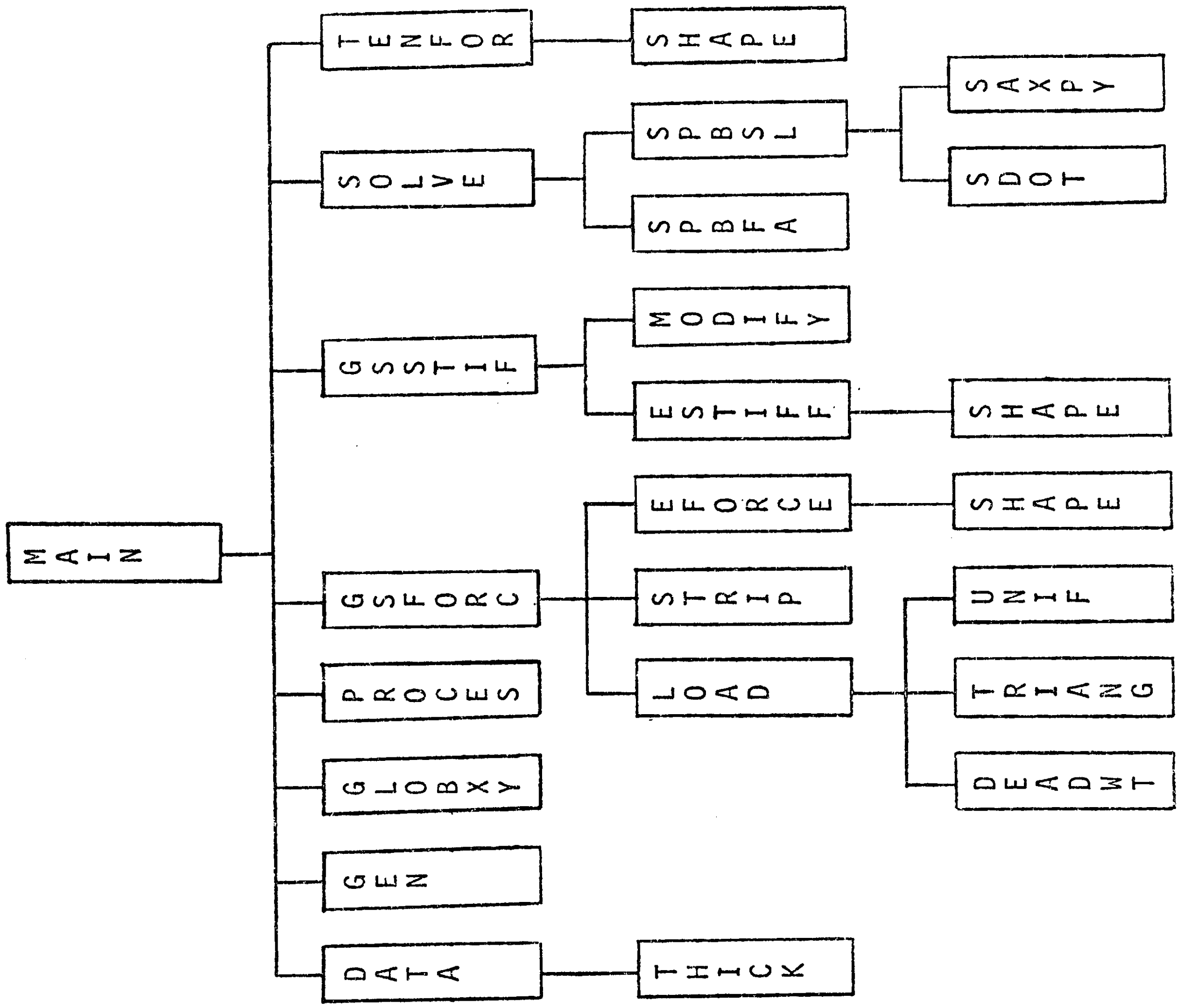


Figure 7: Computer Flow Diagram

input joint constraints. Constructs and prints the system MCOPDE which containes the associated degrees of freedom by element and node. Calculates the half band width, number of system elements and system's number of degrees of freedom.

GSFORC - Reads and echoes user imposed system loads. Generates the global force vector by manipulating (MCODE) the local element force vectors provided by subroutine EFORC into a global sense and then summing them. It also calls subroutines LOAD and STRIP whose calculated data is used by subroutine EFORC.

EFORC - Generates the local element force vectors by associating element loads provided from subroutines LOAD and STRIP with nodal coefficients determined by full gauss point numerical integration. This process is supplemented by calling subroutine SHAPE.

SHAPE - Evaluates the interpolation functions and their derivatives with regard to natural coordinates for a nine-node quadrilateral element.

LOAD - Supervises the determination of overall element load magnitudes as specified by user input load, type, direction, and location. The appropriate subroutine or combination of subroutines TRIANG, UNIF, and/or DEADWT are called.

TRIANG - Calculates the magnitude of an element load induced by a triangular load (hydrostatic and/or soil) upon a rectangular tank wall. The wall load may be at any height from the floor.

UNIF - Calculates the magnitude of an element load induced by a uniform load (surcharge, etc.) upon any rectangular tank plate. Uniform wall loads may vary in height but floor loads must encompass the entire floor.

DEADWT - Calculates the magnitude of gravitational loads imposed upon a rectangular tank wall element due to its weight.

STRIP - Calculates the magnitude of a uniform load upon elements within 2.5 times the floor thickness around the perimeter of the floor slab. This uniform load is determined from wall dead weight being distributed over both the wall base thickness and the 2.5 floor thickness.

GSSTIF - Generates the global system stiffness matrix by manipulating (MCODE) local element stiffness matrices provided by subroutine ESTIF into a global sense and then summing them. Subroutine MODIFY is called when analyzing a rectangular tank where the supporting soil has a specified soil stiffness coefficient.

MODIFY - Modifies the global system stiffness matrix to account for a soil stiffness coefficient.

ESTIF - Generates the local element stiffness matrices using gauss points and numerical integration. The in-plane and bending terms are determined using full integration while the shear terms are calculated through reduced integration. This process is supplemented by calling subroutine SHAPE.

SOLVE - Calculates the global system displacements utilizing a symmetric banded equation solution routine or Linpack equation solver.

TENFOR - Transforms global displacements into local displacements which may be printed. Determines and prints ;the in-plane normal (X & Y), in-plane shearing (X & Y), bending shear (X & Y), bending moment (X & Y), and twisting moment for each node by element at gauss point using full integration.

SPBFA, SPBSL, SDOT, SAXPY - Subroutines of SOLVE which use backward and then forward substitution in determining global displacements.

VII DISCUSSION OF RESULTS

A. Objective

The objective of this thesis is to develop a finite element program which will determine shear forces within the plates of a monolithically constructed concrete tank. It is, therefore, imperative that the program results be compared with previously derived plate solutions to insure program accuracy and correctness prior to the program's acceptance. Consequently, the program verification procedure encompassed three steps. The steps were to compare known results of one, two, and three plate problems with values obtained from this finite element program. Since plate shear evaluations are extremely limited in the literature, the verification procedure addressed displacements and/or force resultants (see Figure 8) with known solutions so that there could be some measure of the program's correctness.

B. Single Plate Comparisons

The single plate analysis encompassed nine comparisons between the results of this program and those published by Bares (6), PCA Bulletin ST-63 (1) and Jofriet (23). The first two comparisons involved plates with uniform loads and symmetric boundary conditions. These comparisons were to verify program derived values and to check symmetric plate displacements and resultants. The third and fourth comparisons were on symmetrically supported plates with triangular loads. Their purpose was to again check appropriate values and to

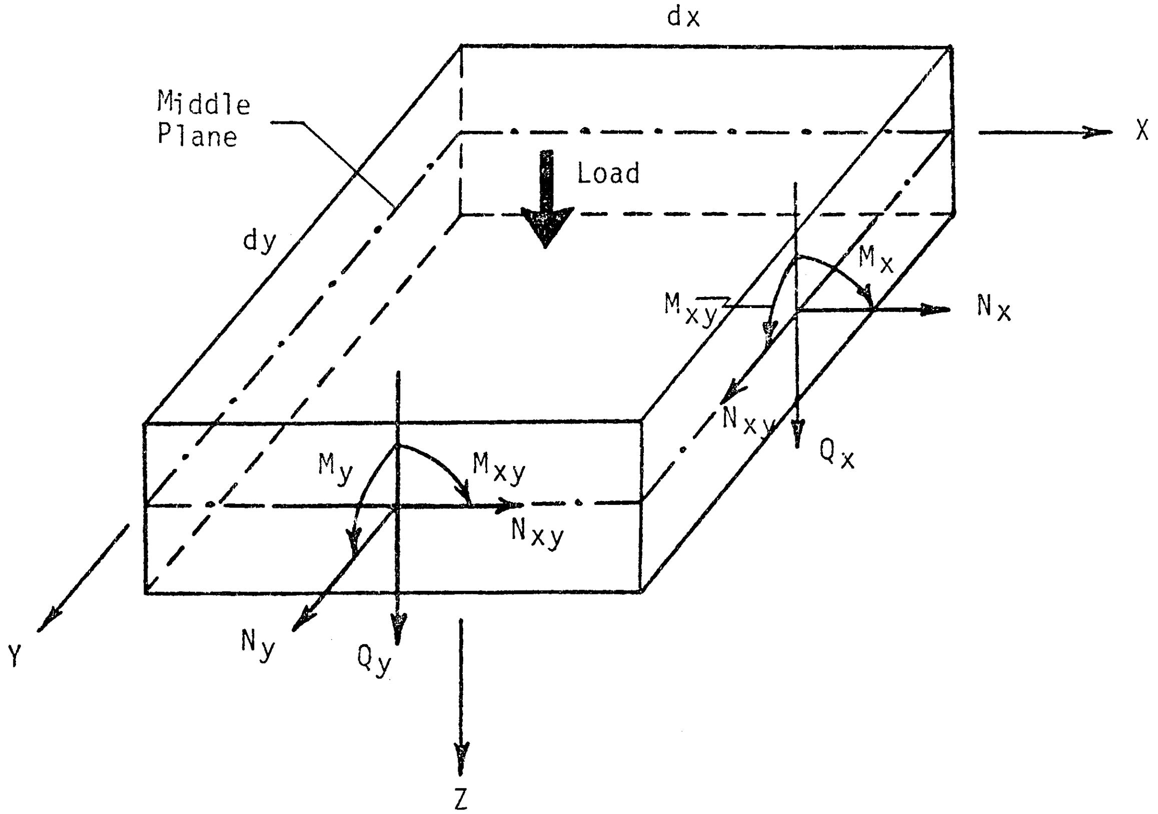


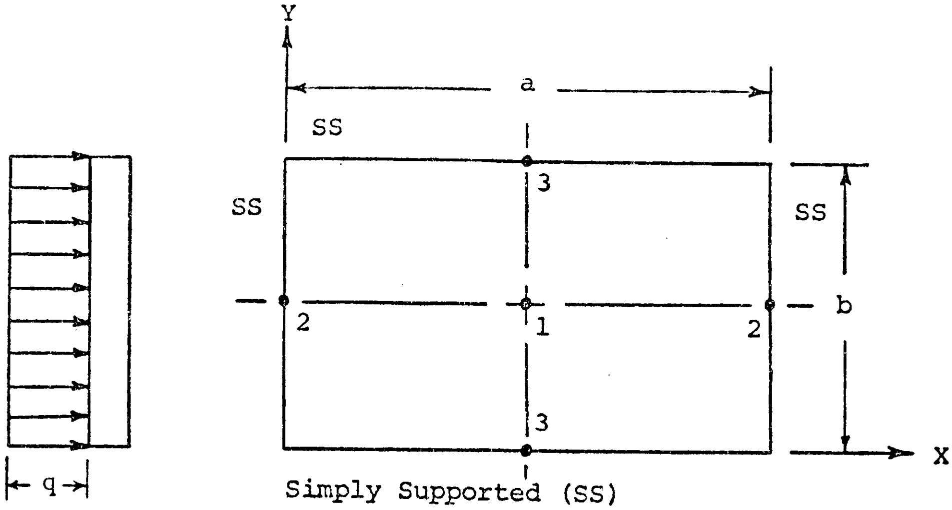
Figure 8: Resultants on a Differential Element

verify the program's ability to accurately simulate a triangular load. The next four comparisons served as a verification that the program would provide correct results with unsymmetric boundary conditions while indicating the finite element mesh refinement necessary to facilitate an appropriate response. The final comparison was to insure that the program properly accounted for tapered walls.

The initial comparison of results was of a uniformly loaded square plate which was simply supported on all four sides. The center out-of-plane displacement and moments plus mid-side shear resultants were analyzed. The results were very favorable and are shown in Table 1. The analysis did reveal, however, the necessity for a highly refined finite element mesh to obtain accurate (shear coefficients within 5.0 %) shear values for the simplest plate problem. Due to plate symmetric loading and boundary conditions, only one quarter of the plate was analyzed with three mesh sizes. The finite element mesh sizes were a 2x2, 4x4, and 8x8 over one quarter of the plate. Program results displayed increased correlation with exact solutions for boundary moment and shear values with increased mesh refinement. Plate center point displacements and moments were consistently good.

The second comparison was for a uniformly loaded square plate having all four sides fixed and undergoing the same mesh refinement sequence as the first comparison. Again, the center out-of-plane displacement and moments plus mid-side shear resultants were compared. The comparisons are shown in Table 2 and the results are good. Variations are zero when utilizing an 8x8 mesh on one quarter of a plate in all but shear values, and shear coefficients are within a 5.0 %

TABLE 1: Comparison with Known Solutions

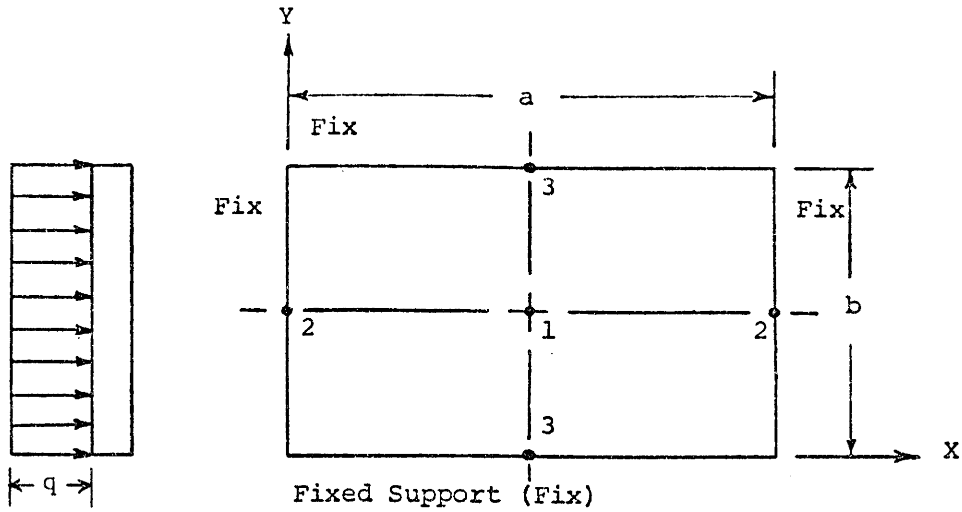


Uniform Load
  b/a = 1.0
Constant Plate Thickness

Description	FE	Bares (6)	% Diff	M.F.*
w_1	0.0443	0.0443	0.0	$\times \frac{qb^4}{Eh^3}$
M_{x1}	0.0478	0.0478	0.0	$\times qb^2$
M_{y1}	0.0478	0.0478	0.0	$\times qa^2$
Q_{x2}	0.330	0.338	2.4	$\times qb$
Q_{y3}	0.330	0.338	2.4	$\times qa$

* M.F. - Multiplication Factor

TABLE 2: Comparison with Known Solutions



Uniform Load
b/a = 1.0
Constant Plate Thickness

Description	FE	Bares (6)	% Diff	M.F.*
w_1	0.0139	0.0139	0.0	$\times \frac{qb^4}{Eh^3}$
M_{x1}	0.0229	0.0229	0.0	$\times\ qb^2$
M_{y1}	0.0229	0.0229	0.0	$\times\ qa^2$
Q_{x2}	0.428	0.446	4.2	$\times\ qb$
Q_{y3}	0.428	0.446	4.2	$\times\ qa$

* M.F. - Multiplication Factor

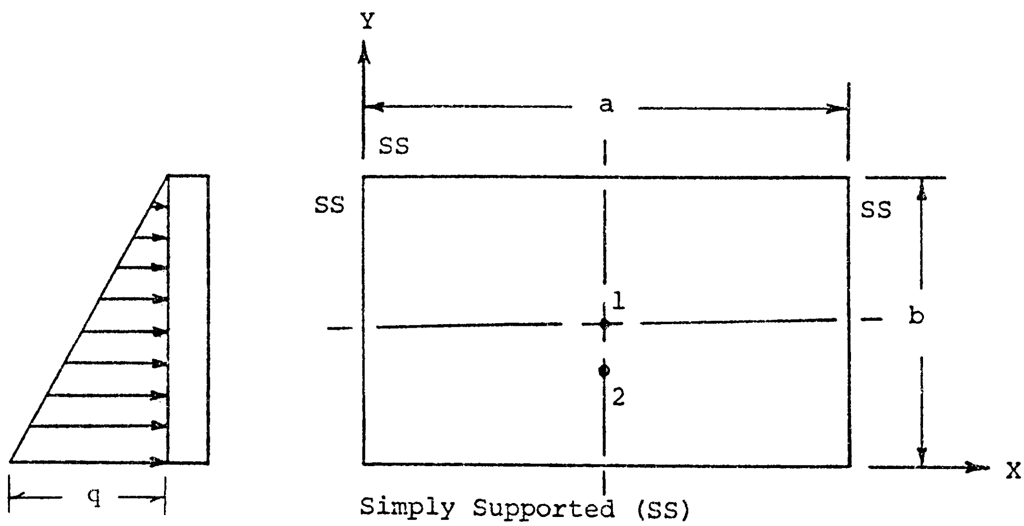
allowance. For both this and the first comparison, the plate displacements and resultants were also symmetric as expected.

The third and fourth comparisons were similar to the first two except that a triangular load was imposed on the plate, requiring the finite element mesh to encompass the whole plate. Upon comparing the center and maximum out-of-plane displacements plus their associated moments utilizing an 8x8 mesh, the results were favorable and are displayed in Tables 3 and 4. It is noted, however, that the triangular load does manifest increased differences in moment coefficients. When taken in relation to the first two comparisons, this increased variation is similarly expected with the shear coefficients and is accounted for by a coarser finite element mesh being utilized.

The fifth and sixth comparisons compared shear coefficients along plate edges using a full plate 8x8 finite element mesh. The fifth comparison consisted of a plate having three fixed sides and one free side while the sixth had three fixed sides and one simply supported side. The results are given in Figures 9 through 11 and again show good correlation. It is again expected that the differences are due to lack of refinement of the finite element mesh. Program results consistently converge toward Bares' (6) values as the modeled plate became less stiff through finite element mesh refinement.

The next two single plate comparisons were of program values obtained from a full plate 8x8 mesh with Figures 1 and 2 of PCA Bulletin ST-63 (1). The first analysis was of a plate having two

TABLE 3: Comparison with Known Solutions

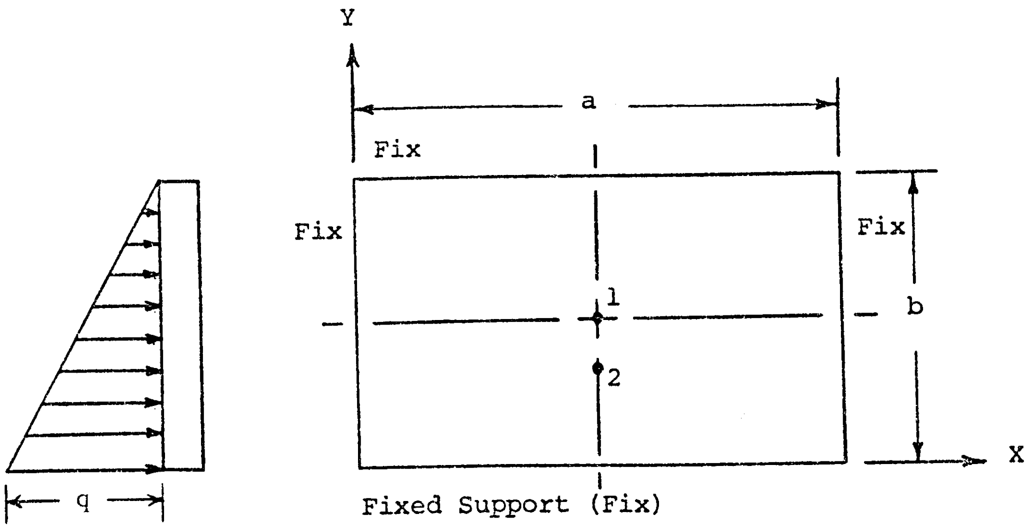


Triangular Load
b/a = 1.0
Constant Plate Thickness

Description	FE	Bares (6)	% Diff	M.F.*
w_1	0.0238	0.0238	0.0	$\times \frac{qb^4}{Eh^3}$
M_{x1}	0.0219	0.0212	3.3	$\times qb^2$
M_{y1}	0.0213	0.0212	0.5	$\times qa^2$
w_2	0.0240	0.0238	0.8	$\times \frac{qb^4}{Eh^3}$
M_{x2}	0.0236	0.0241	2.1	$\times qb^2$
M_{y2}	0.0213	0.0212	0.5	$\times qa^2$

* M.F. - Multiplication Factor

TABLE 4: Comparison with Known Solutions



Triangular Load
$b/a = 1.0$
Constant Plate Thickness

Description	FE	Bares (6)	% Diff	M.F.*
w_1	0.0074	0.0075	0.8	$\times \frac{qb^4}{Eh^3}$
M_{x1}	0.0105	0.0101	3.5	$\times qb^2$
M_{y1}	0.0100	0.0101	1.0	$\times qa^2$
w_2	0.0075	0.0075	0.0	$\times \frac{qb^4}{Eh^3}$
M_{x2}	0.0102	0.0103	1.0	$\times qb^2$
M_{y2}	0.0113	0.0114	0.9	$\times qa^2$

* M.F. - Multiplication Factor

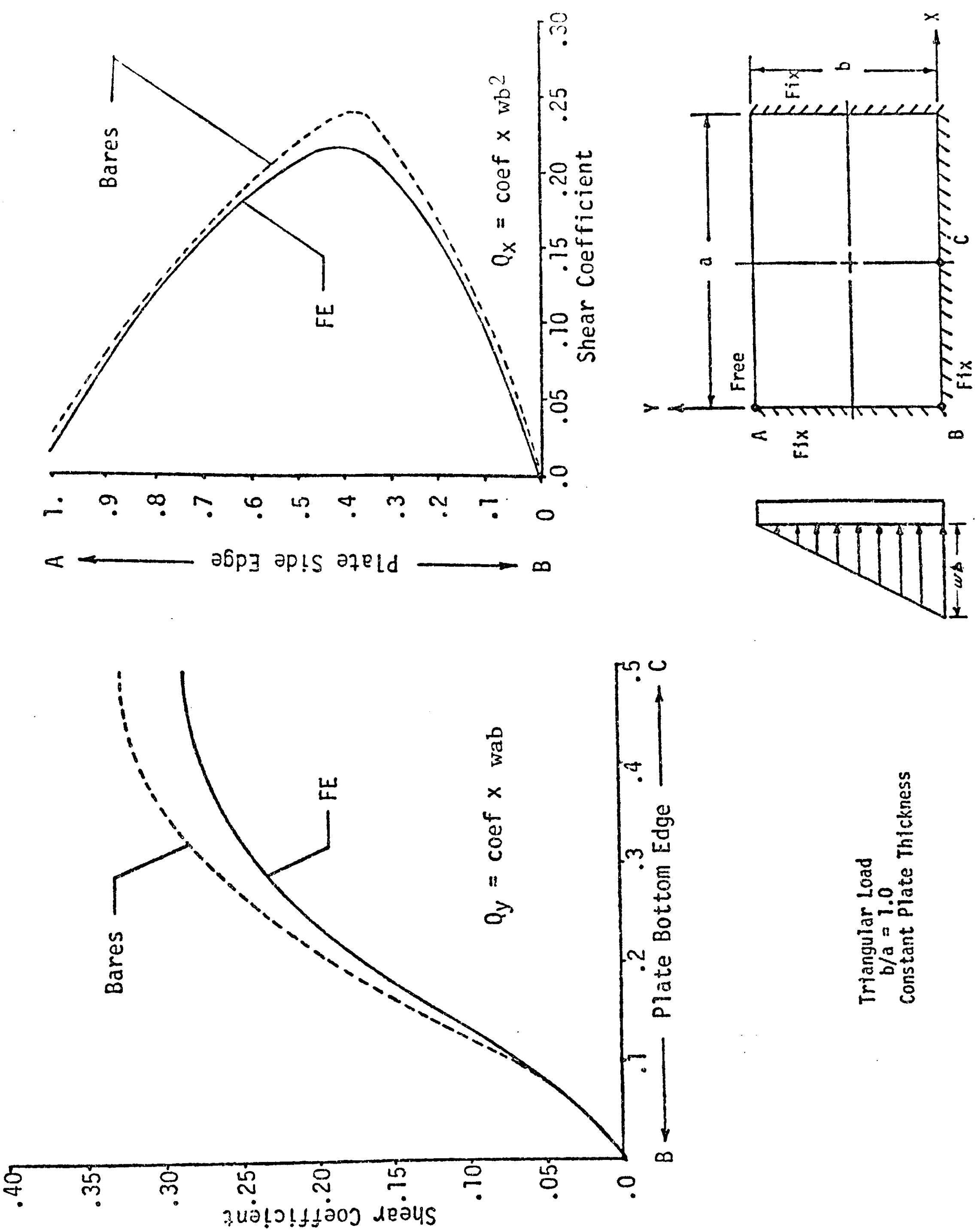


Figure 9: Comparison with Known Solutions

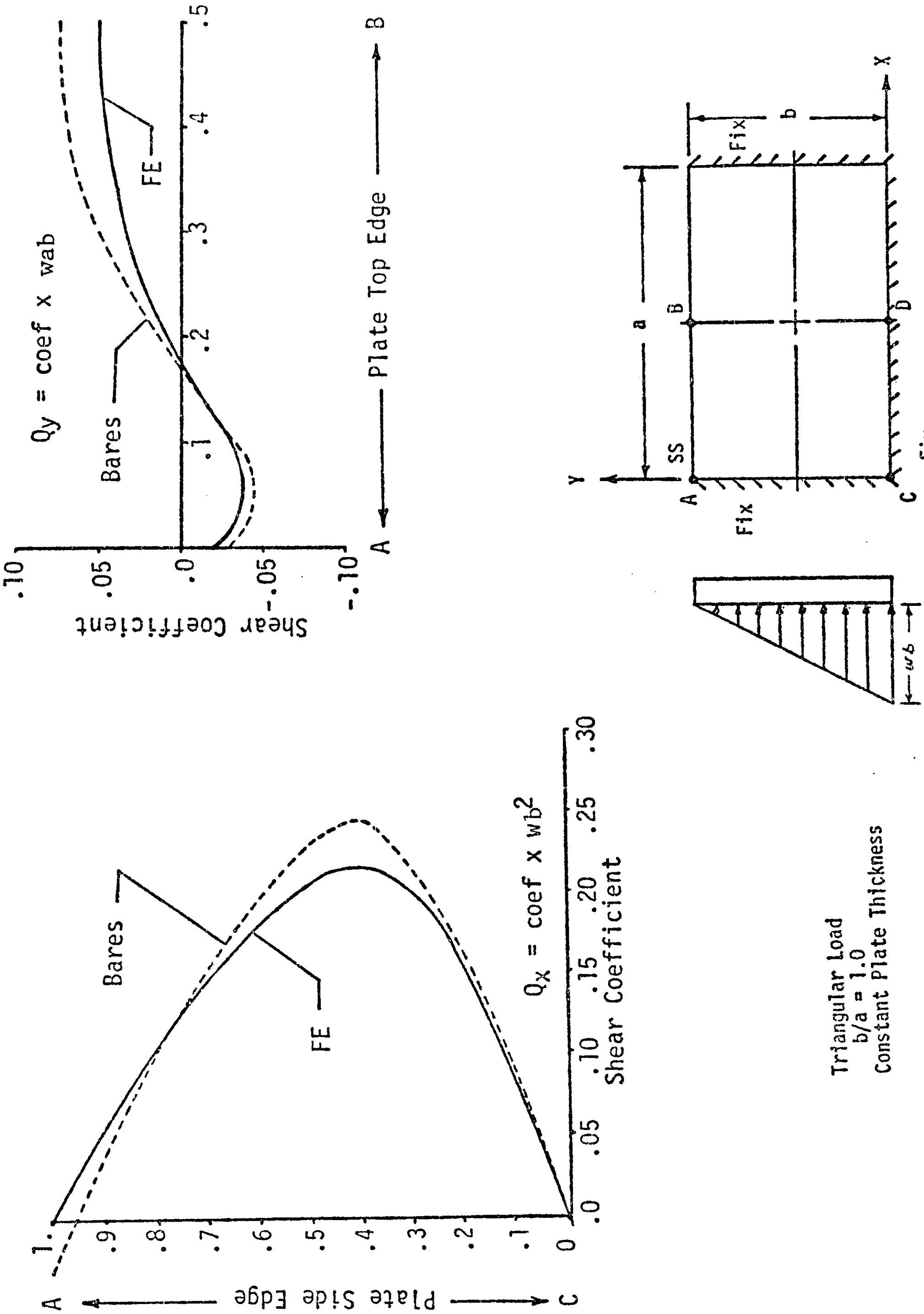
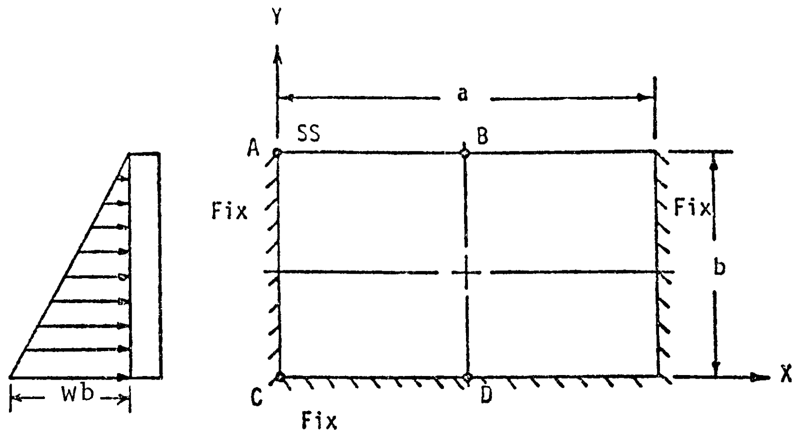


Figure 10: Comparison with Known Solutions



Triangular Load
b/a = 1.0
Constant Plate Thickness

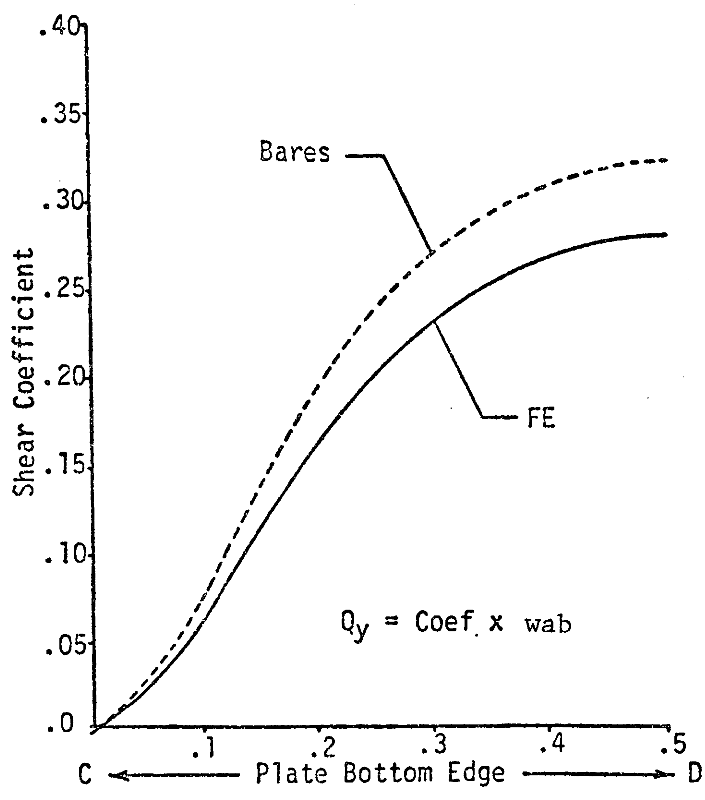


Figure 11: Comparison with Known Solutions

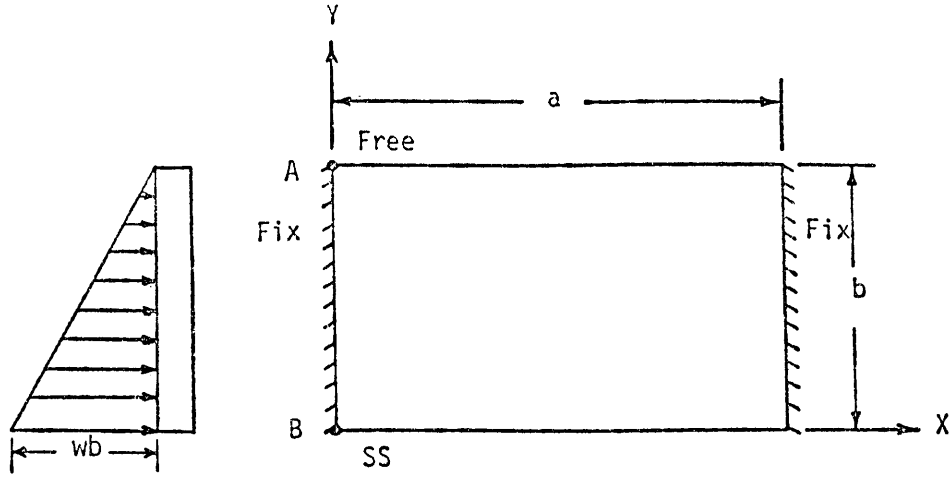
fixed, one simply supported, and one free side. The second analysis was of a plate having two fixed and two simply supported sides. Both plates were loaded with a triangular load and the comparison was good as shown in Figures 12 and 13.

The last single plate comparison analyzed tapered walls to insure that the program could be extended for such cases. The moment coefficients obtained from the finite element program were compared with those published by Jofriet (23). Again, the correlations were favorable. The results are displayed in Figure 14 and show good program response.

C. Two Plate Comparisons

The two plate analysis involved five comparisons. The comparisons investigated one quarter of a rectangular tank having rigid ninety degree angle wall-to-wall and hinged wall-to-base connections with varying top of wall constraints. The first considered two plates under an internal uniform load. The purpose was to verify symmetric displacements while accounting for increased plate stiffnesses resulting from plate interaction. The wall plate dead weights were addressed in the second comparison with the same objective as the first one. The third analysis considered two clamped togrther plates. The objective was to compare results with a similarly constrained single plate and verify two plate program coordinate transformations. The last two comparisons then focused on deflections and compared program moment coefficients with PCA Tables V and VI (1).

The first arrangement analyzed was two equal plates with each



Triangular Load
b/a = 1.0
Constant Plate Thickness

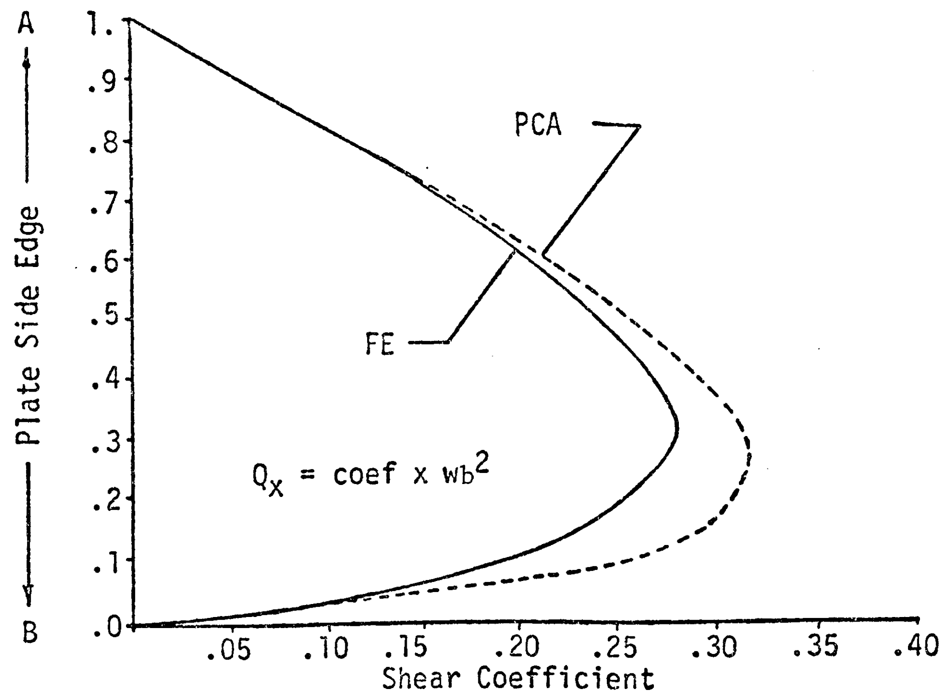


Figure 12: Comparison with Known Solutions

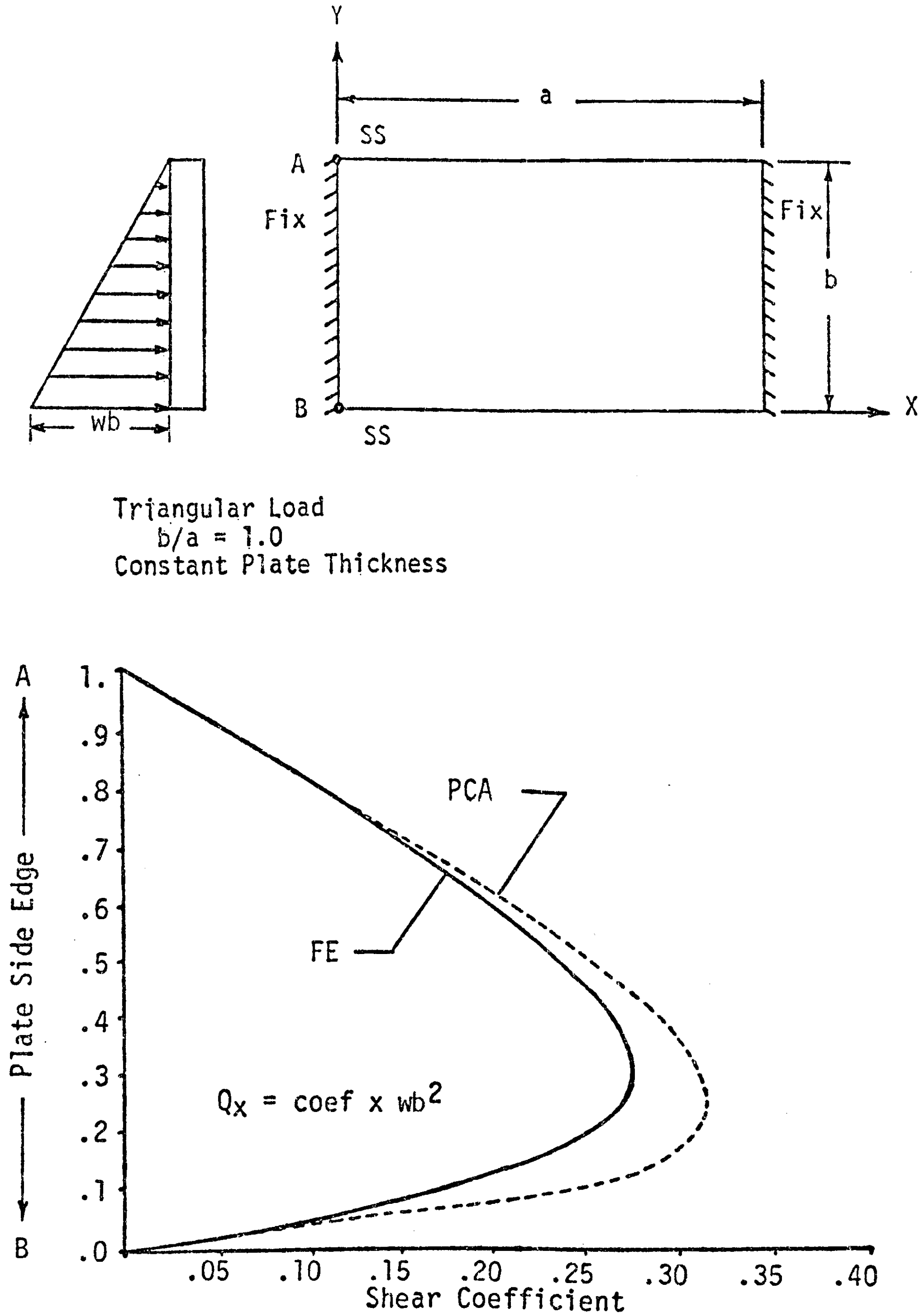


Figure 13: Comparison with Known Solutions

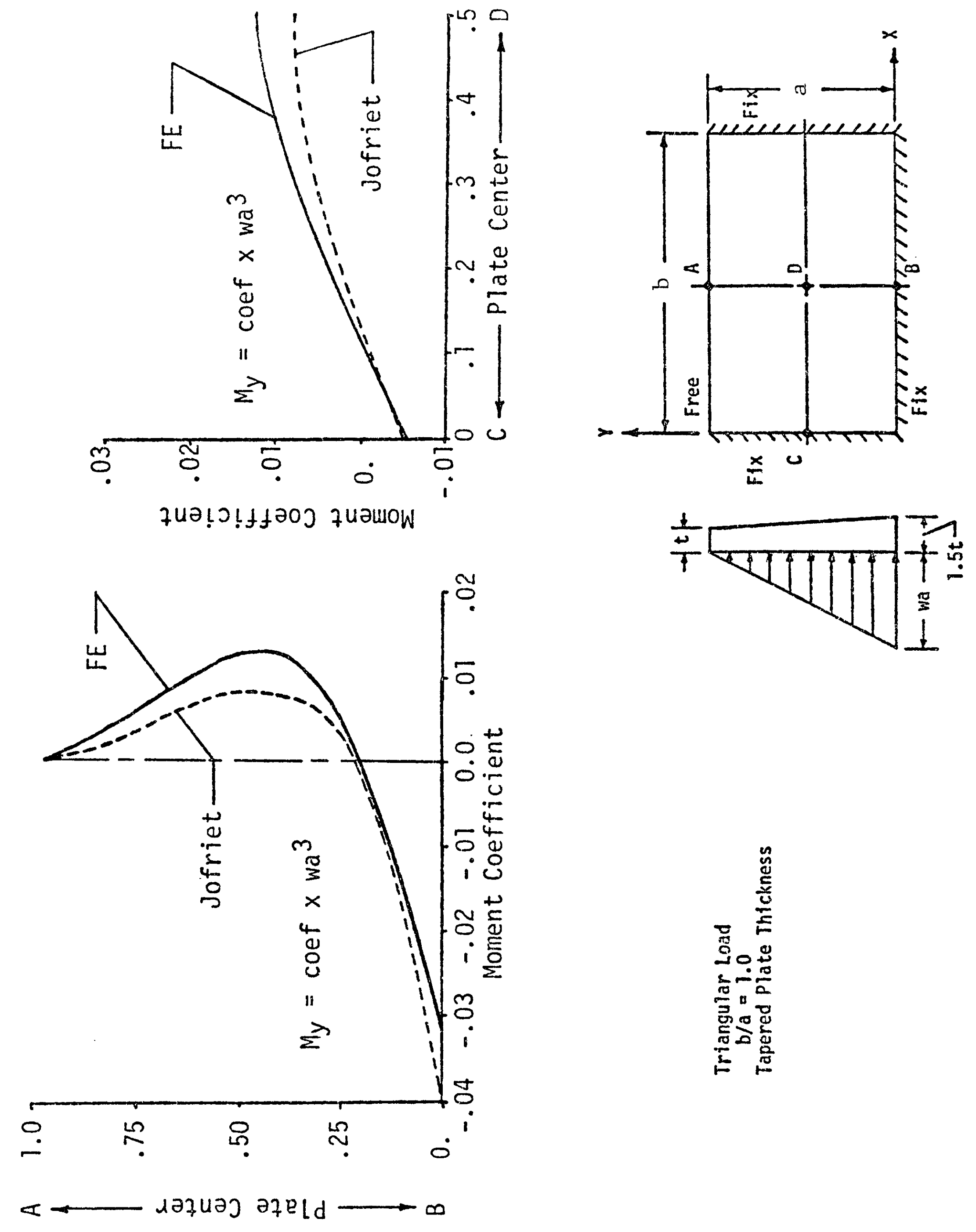


Figure 14: Comparison with Known Solutions

plate having a 4x4 finite element mesh. Upon imposing internal uniform loads against each plate and imposing boundary conditions depicted in Figure 15, deflections were obtained and analyzed. The results are depicted in Figure 15 and display the system's expected displaced configuration. The increased plate stiffness effects due to plate interaction are visible in the decreased deflection along the plate's horizontal center line.

The second arrangement again considered two equal plates with each plate having a 4x4 mesh. The only load condition was the inherent plate dead load incurred from each plate's concrete wall weight. The system's boundary conditions and deformed shape is shown in Figure 16. The results show uniform vertical displacement.

The third investigation analyzed the same two plates as in the first arrangement but with different boundary conditions and an internal triangular load. Wall-to-wall connections were clamped while wall-to-top and bottom edges were free or hinged, respectively. The derived results were compared with single plate data determined from a plate having two opposing edges clamped, one free, and one hinged. The correlations were excellent and show that program coordinate transformation sequences were properly performed.

The last two plate problems addressed equal plates each having an 8x8 mesh with an imposed internal triangular load. The fourth and fifth system's plates were free or hinged along their top edges, respectively. The displaced configurations were as expected and are displayed with boundary constraints in Figures 17 and 18. Upon comparing program results with those provided by PCA tables V and VI,

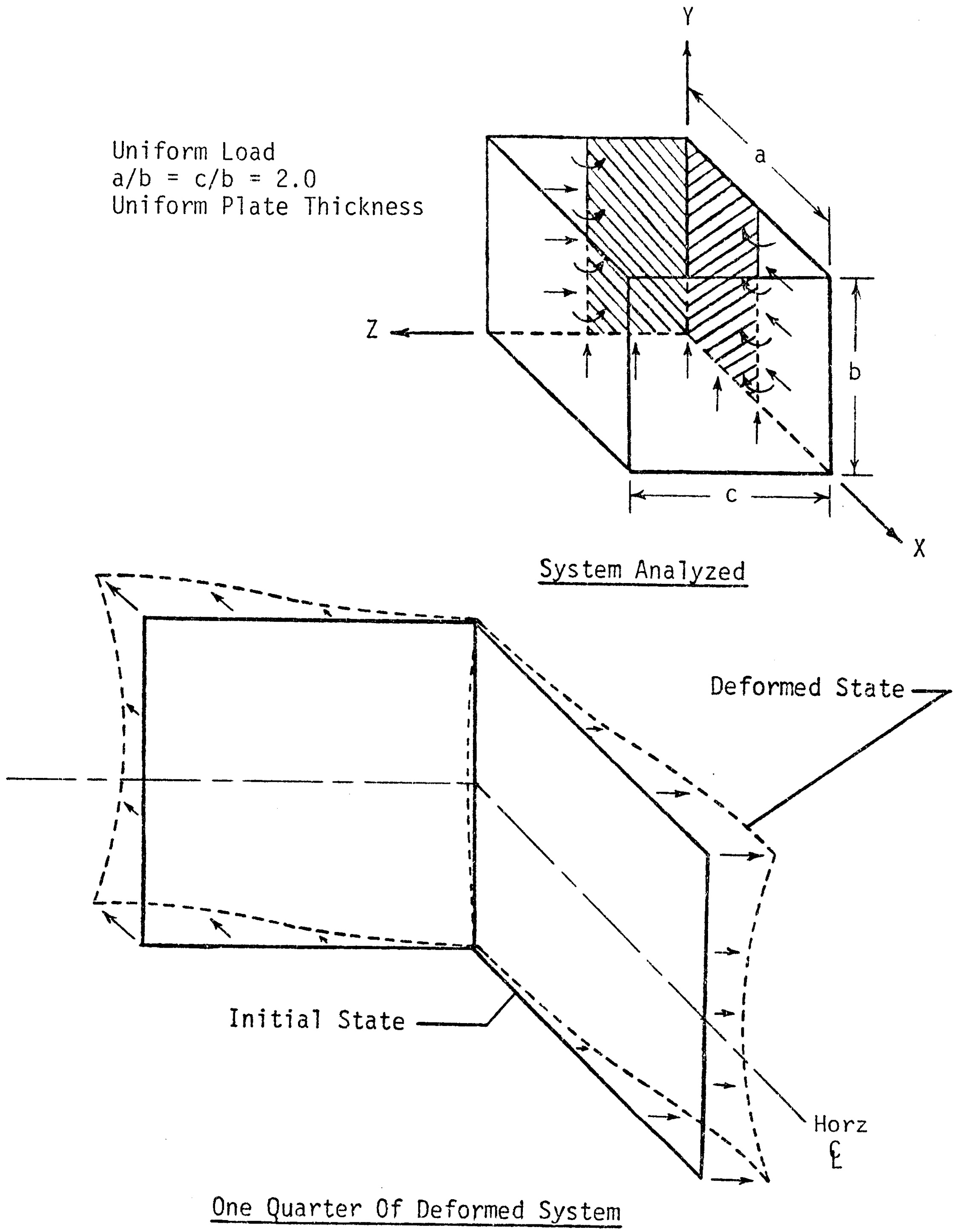
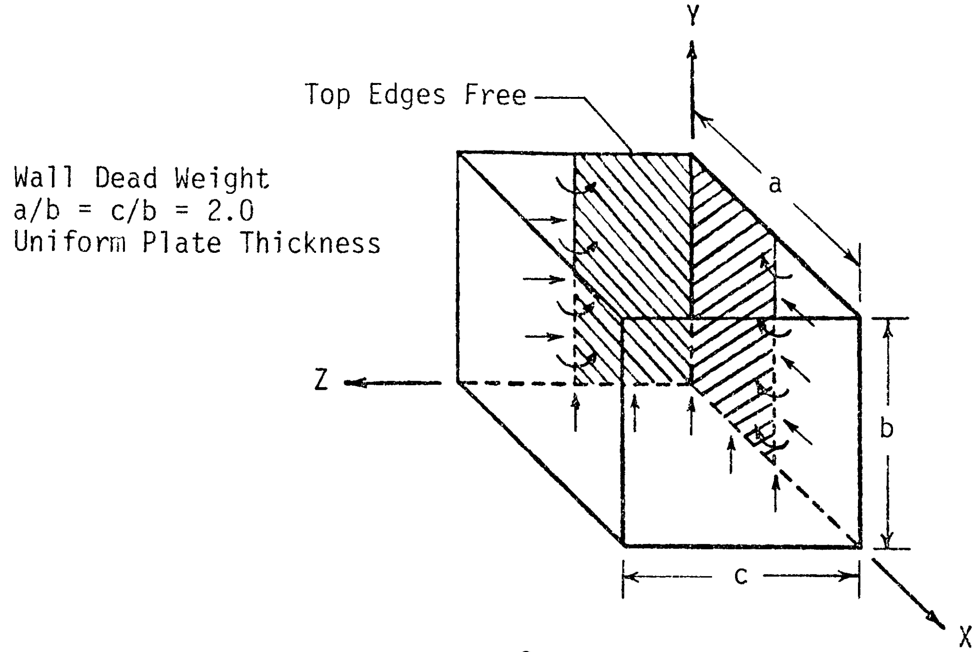
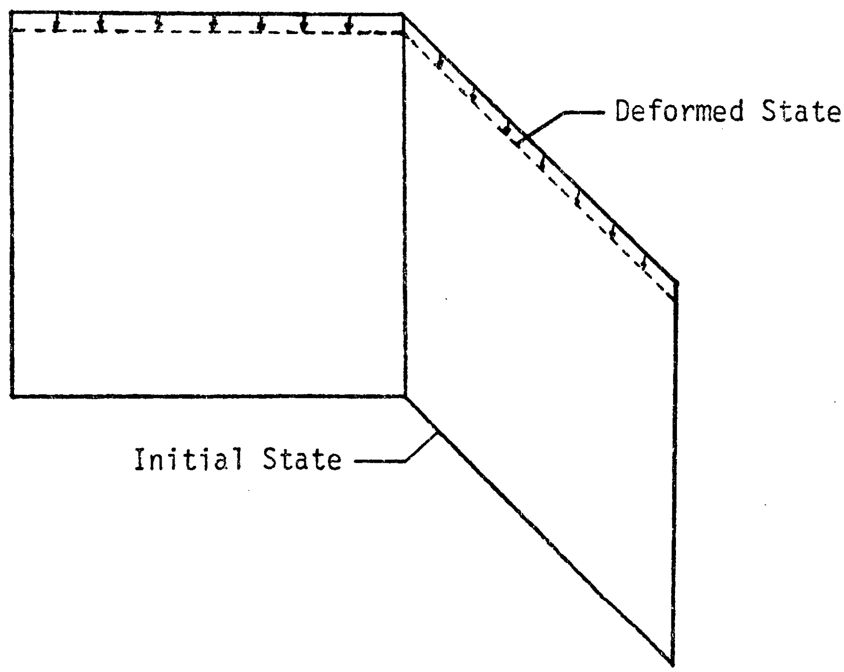


Figure 15: Two Plate Displacement Configuration From Uniform Loading

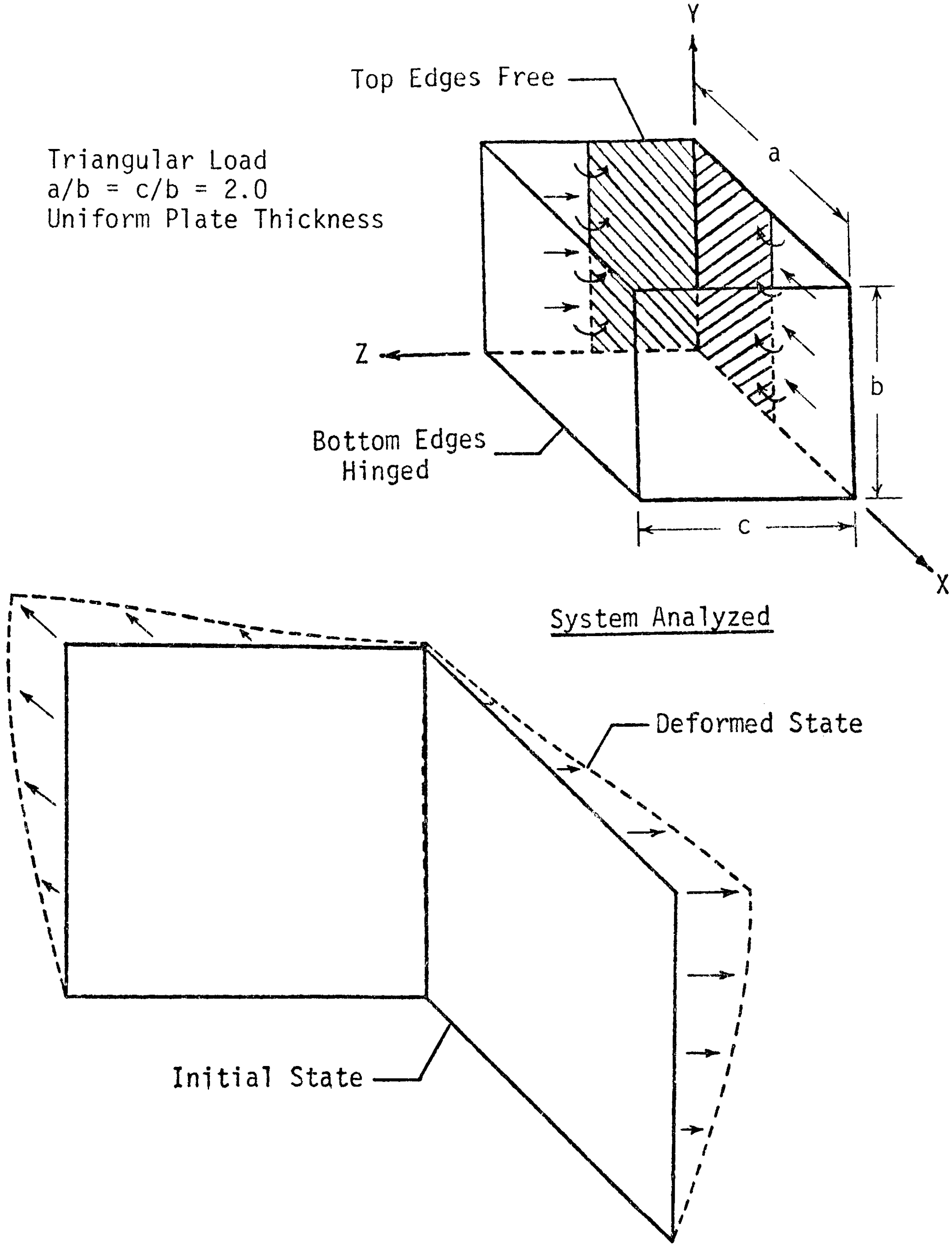


System Analyzed



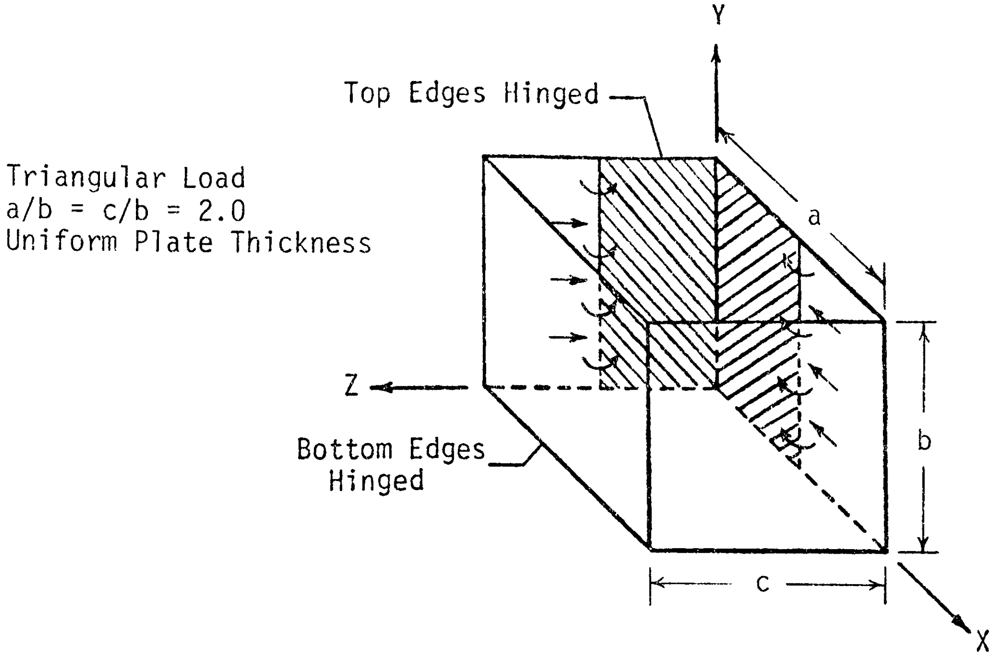
One Quarter Of Deformed System

Figure 16: Two Plate Displacement Configuration
From Wall Dead Weight

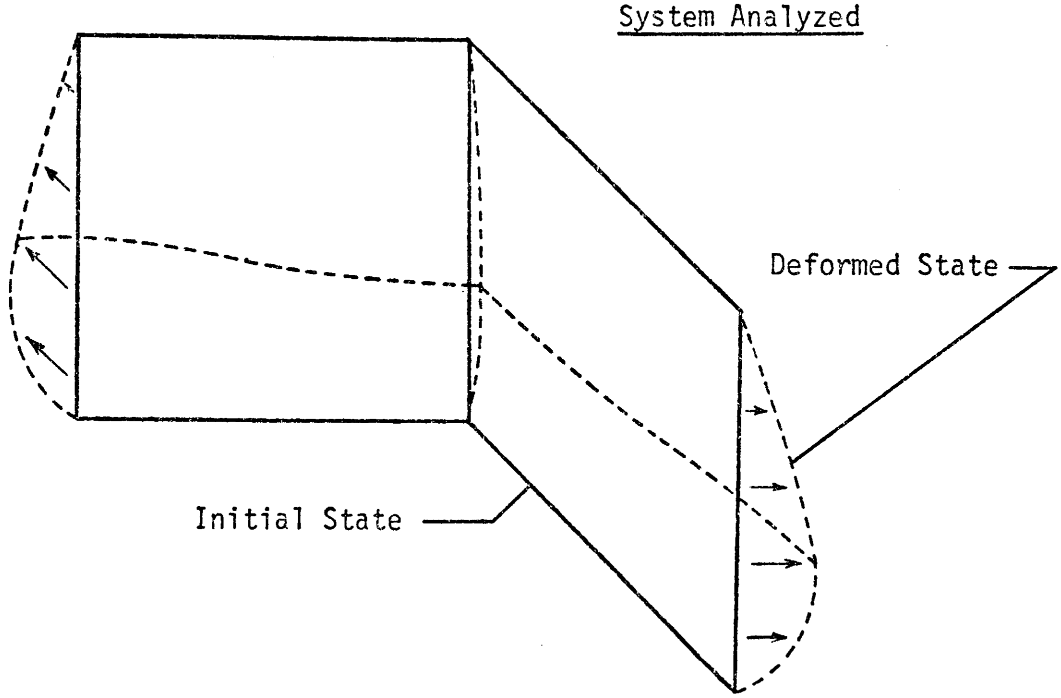


One Quarter Of Deformed System

Figure 17: Two Plate Displacement Configuration
From Triangular Loading



System Analyzed



One Quarter Of Deformed System

Figure 18: Two Plate Displacement Configuration From Triangular Loading

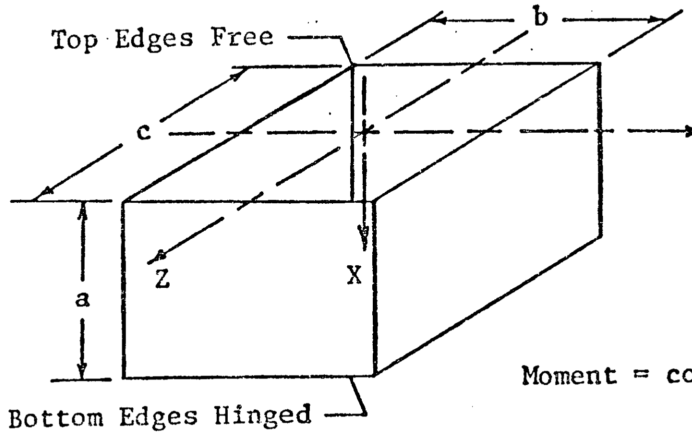
good correlation in moment coefficients was obtained for both systems analyzed (Tables 5 and 6). Figures 19 and 20 display corner shear coefficients for the two analyzed systems.

D. Three Plate System

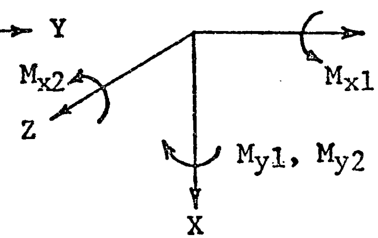
The three plate analysis investigated one quarter of a rectangular tank where wall-to-wall and wall-to-base connections were rigid ninety degree angles. The analysis was restricted by the storage capacity offered by the Virginia Polytechnic Institute and State University computer system. The largest three plate finite element problem possible with this program was a system of three plates with each plate consisting of 16 elements, requiring 8.7 megabits of computer storage. Resultant accuracy is consequently questionable since prior analysis had shown that a two plate problem required a minimal 8x8 finite element mesh. The analysis of three plate system displacements was performed, however.

Various loads including triangular and strip were imposed on the three plate system and the system's displaced configuration observed. The deformed state of three equal plates joined rigidly at ninety degree angles with imposed internal triangular loads on the two vertical plates is shown in Figure 21. The system boundary conditions are those previously discribed and automatically imposed by the program. In each three plate situation, the system reacted as expected. These results imply appropriate program simulation of a monolithic rectangular tank system and which with additional finite element mesh refinement would determine accurate shear values.

Table 5: Comparison with Known Solutions



Triangular Load
Uniform Plate Thickness



Moment = coef x wa^3

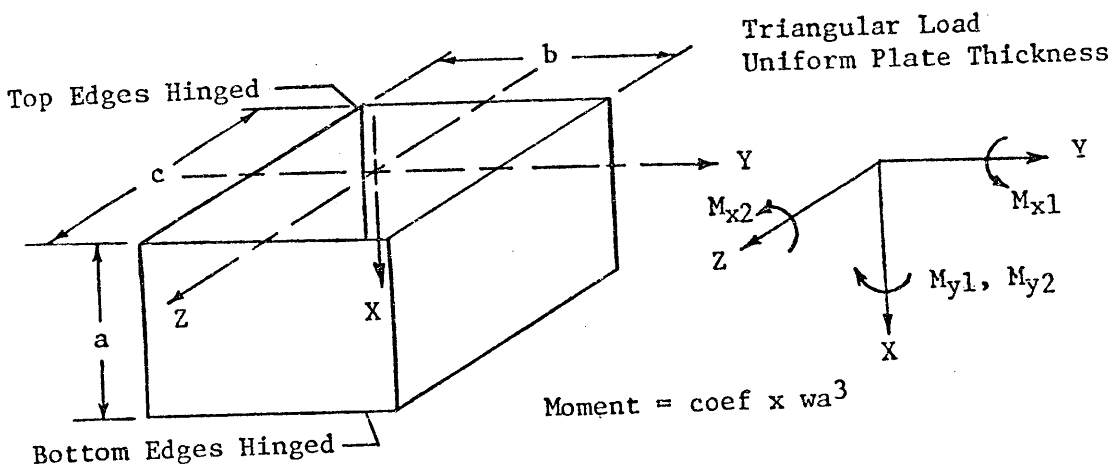
b/a = 2.0 t = 10.0 in. c/a = 2.0

		y=0		y=b/4		y=b/2		z=c/4		z=0	
	x/a	M_{x1}	M_{y1}	M_{x1}	M_{y1}	M_{x1}	M_{y1}	M_{x2}	M_{y2}	M_{x2}	M_{y2}
F.E.	0	+.000	+.046	+.000	+.012	+.000	-.089	+.000	+.012	+.000	+.046
	1/4	+.016	+.042	+.006	+.014	-.017	-.088	+.006	+.014	+.016	+.042
	1/2	+.033	+.036	+.020	+.016	-.015	-.080	+.020	+.016	+.033	+.036
	3/4	+.035	+.024	+.025	+.013	-.010	-.056	+.025	+.013	+.035	+.024
PCA (1)	0	+.000	+.045	+.000	+.011	+.000	-.091	+.000	+.011	+.000	+.045
	1/4	+.016	+.042	+.006	+.014	-.019	-.094	+.006	+.014	+.016	+.042
	1/2	+.033	+.036	+.020	+.016	-.018	-.089	+.020	+.016	+.033	+.036
	3/4	+.036	+.024	+.025	+.014	-.013	-.065	+.025	+.014	+.036	+.024

Minus sign indicates tension on the loaded side.

$M_z = M_{y2}$

Table 6: Comparison with Known Solutions



b/a = 2.0 t = 10.0 in. c/a = 2.0

		y=0		y=b/4		y=b/2		z=c/4		z=0	
	x/a	M_{x1}	M_{y1}	M_{x1}	M_{y1}	M_{x1}	M_{y1}	M_{x2}	M_{y2}	M_{x2}	M_{y2}
F.E.	1/4	+.025	+.013	+.016	+.009	-.006	-.030	+.016	+.009	+.025	+.013
	1/2	+.042	+.019	+.028	+.015	-.008	-.047	+.028	+.015	+.042	+.019
	3/4	+.040	+.016	+.029	+.013	-.006	-.041	+.029	+.013	+.040	+.016
PCA (1)	1/4	+.025	+.013	+.015	+.009	-.007	-.037	+.015	+.009	+.025	+.013
	1/2	+.042	+.020	+.028	+.015	-.012	-.059	+.028	+.015	+.042	+.020
	3/4	+.040	+.016	+.029	+.013	-.011	-.053	+.029	+.013	+.040	+.016

Minus sign indicates tension on the loaded side.

$M_z = M_{y2}$

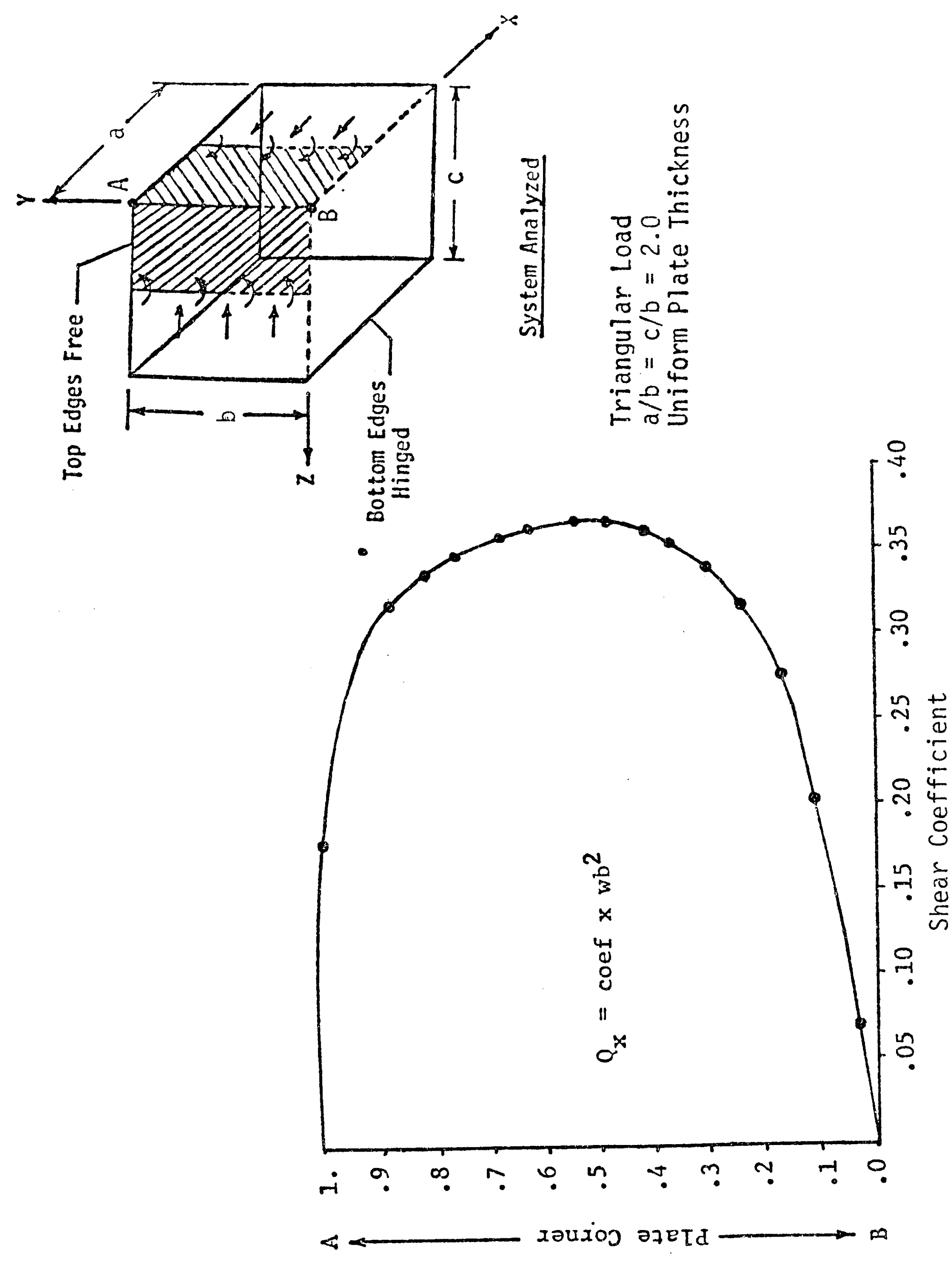


Figure 19: Two Plate Corner Shear

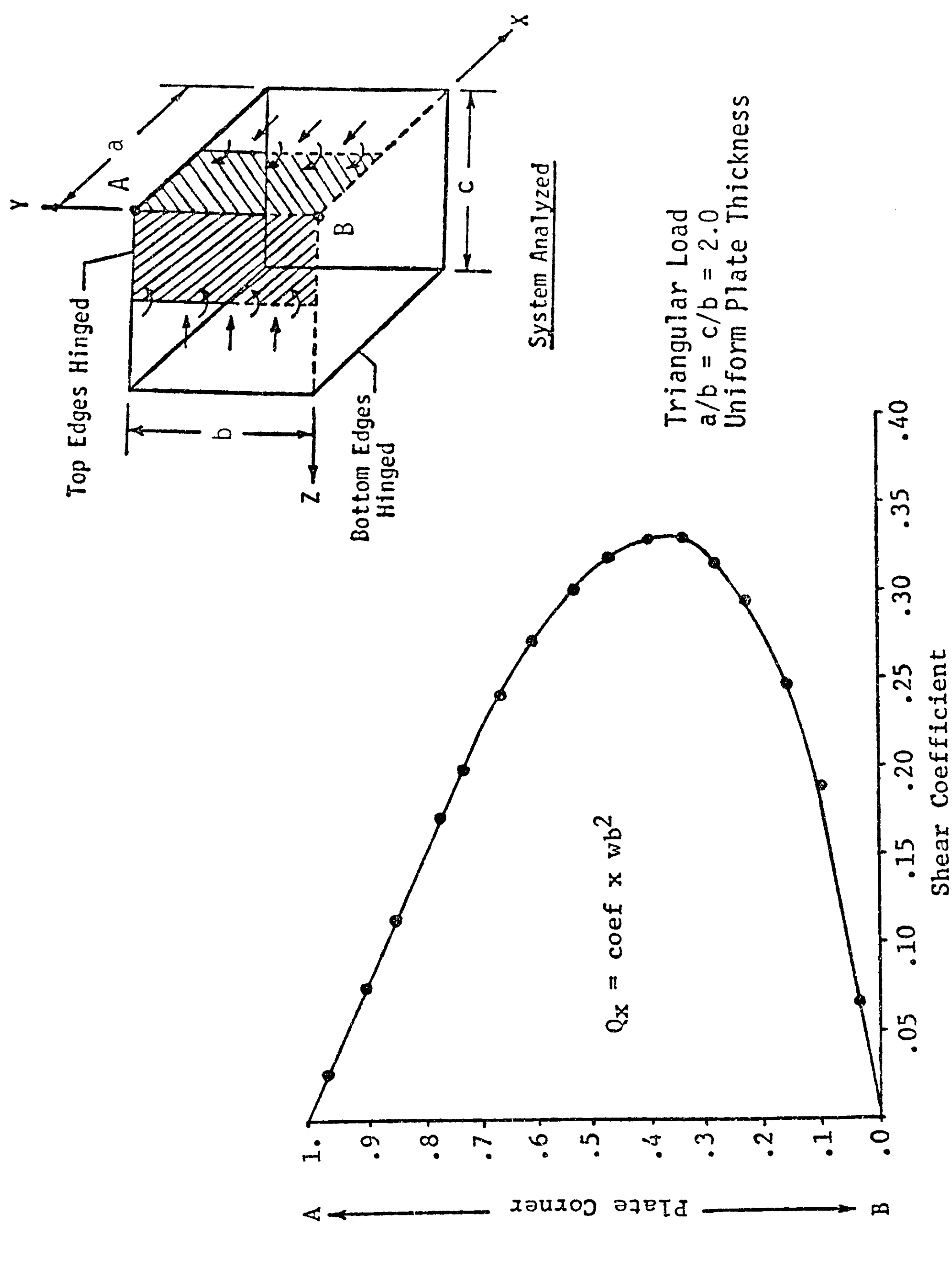


Figure 20: Two Plate Corner Shear

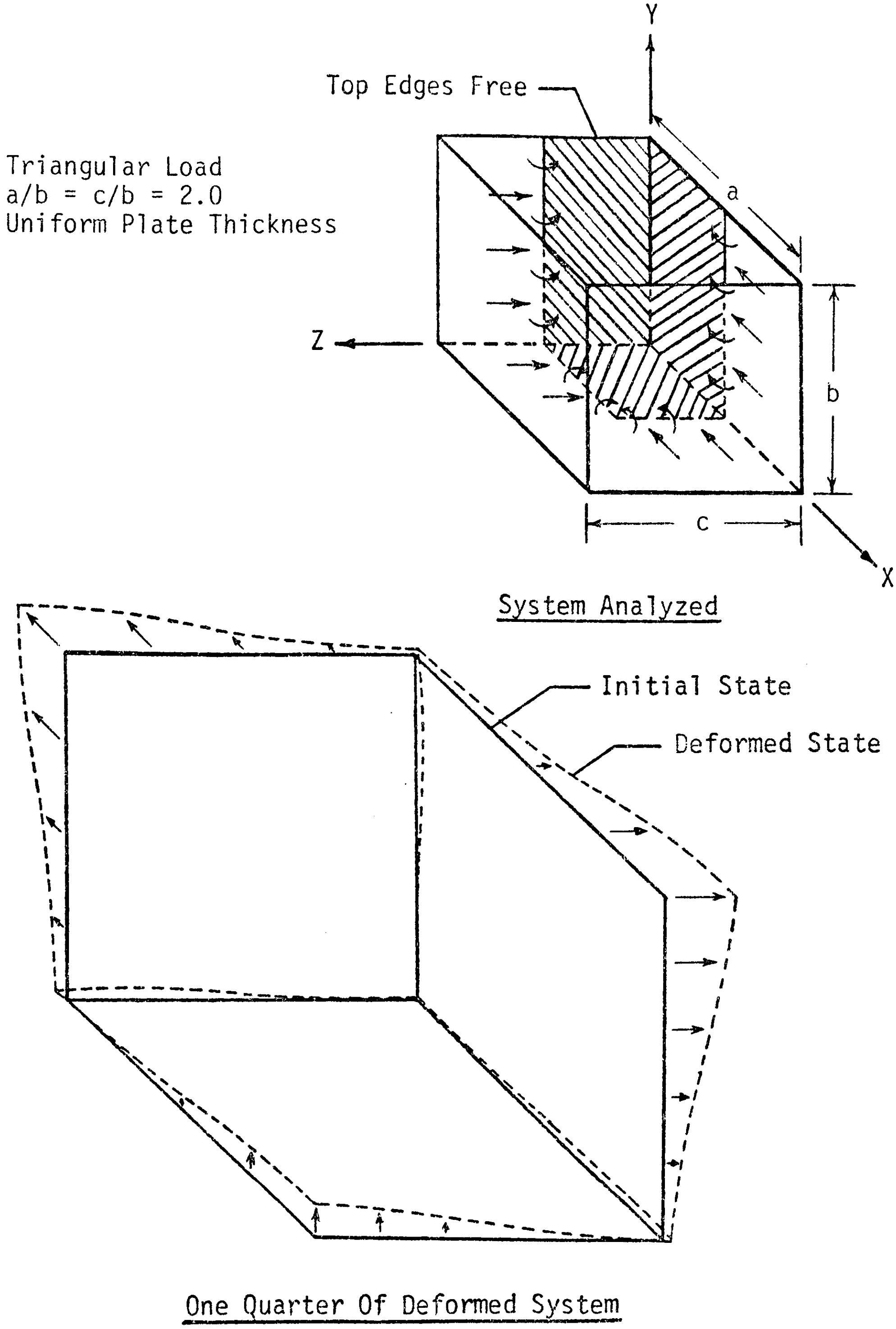


Figure 21: Three Plate Displacement Configuration

The upward base plate displacement shown in Figure 21 is considered inaccurate. There are gravitational forces not accounted for in the program which would insure base plate contact with the supporting foundation while not creating additional forces upon the system. The difference in shear diagrams shown in Figures 22 and 24 versus Figures 19 and 20 is attributed to this shortcomming in idealizing the system.

Shear coefficients for two monolithically constructed rectangular tanks undergoing internal triangular wall loading with either a free or hinged top edge are displayed in Figures 22 through 25. The results were determined from plates having a 4x4 finite element mesh. The values show shear trends for the systems being analyzed, but the shear values are only approximately correct since a 4x4 mesh was used. With increased mesh refinement, tables can be prepared with various ratios of tank width, length, and depth. Other parameters which could be included are plate thicknesses and tapered or stepped walls.

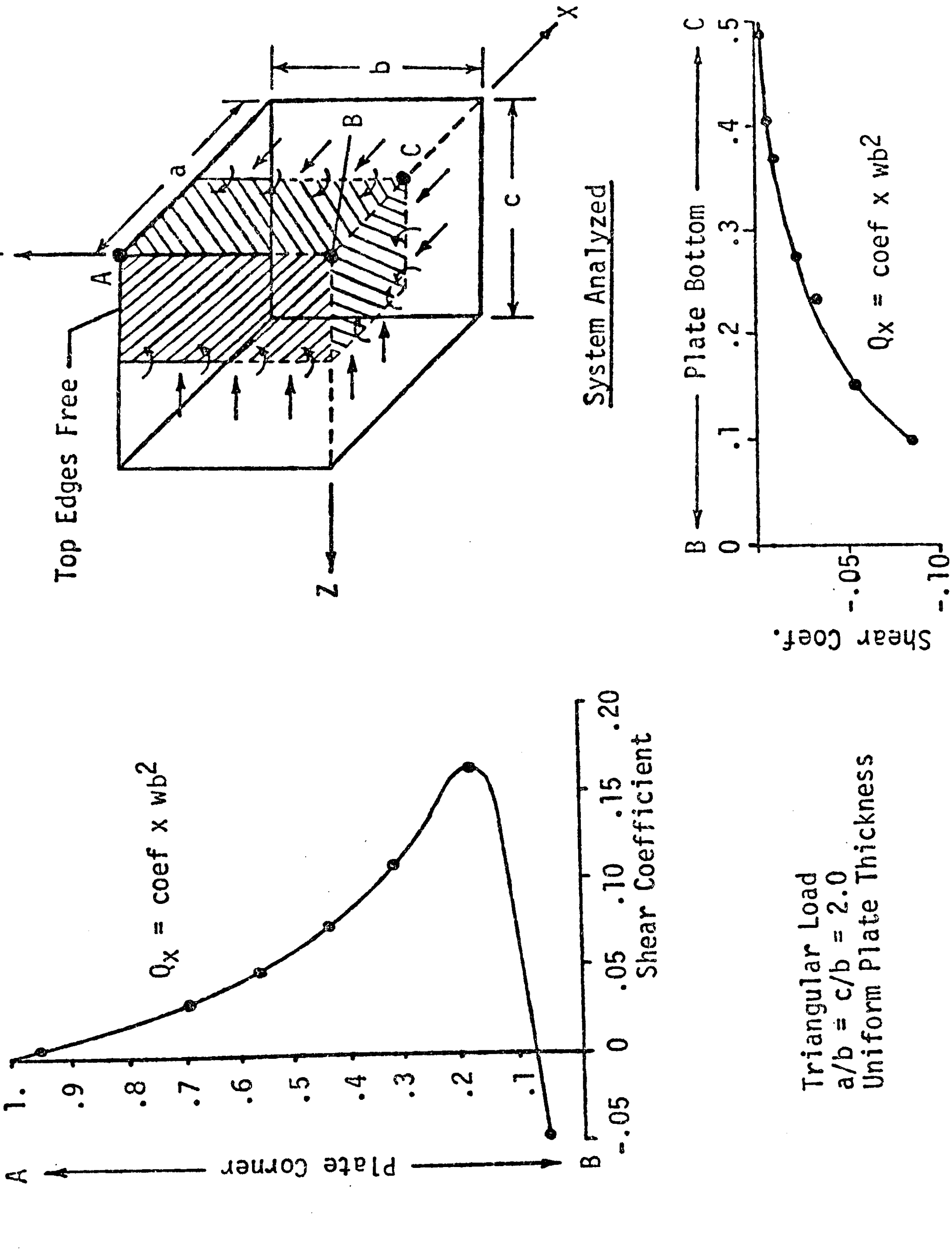
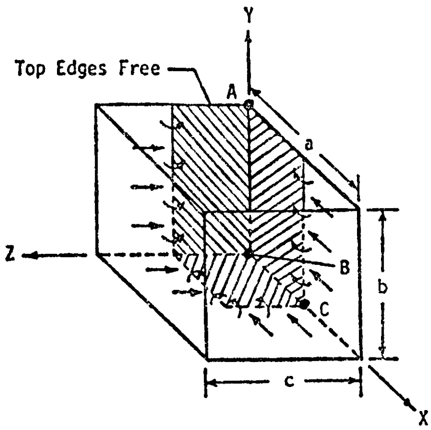


Figure 22: Three Plate Shear Coefficients



System Analyzed

Triangular Load
a/b ≐ c/b = 2.0
Uniform Plate Thickness

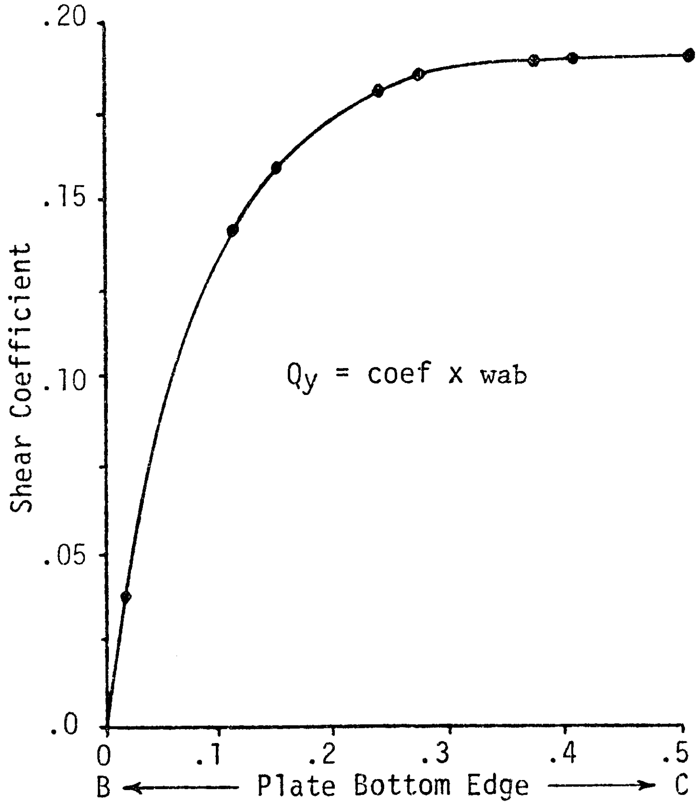


Figure 23: Three Plate Shear Coefficients

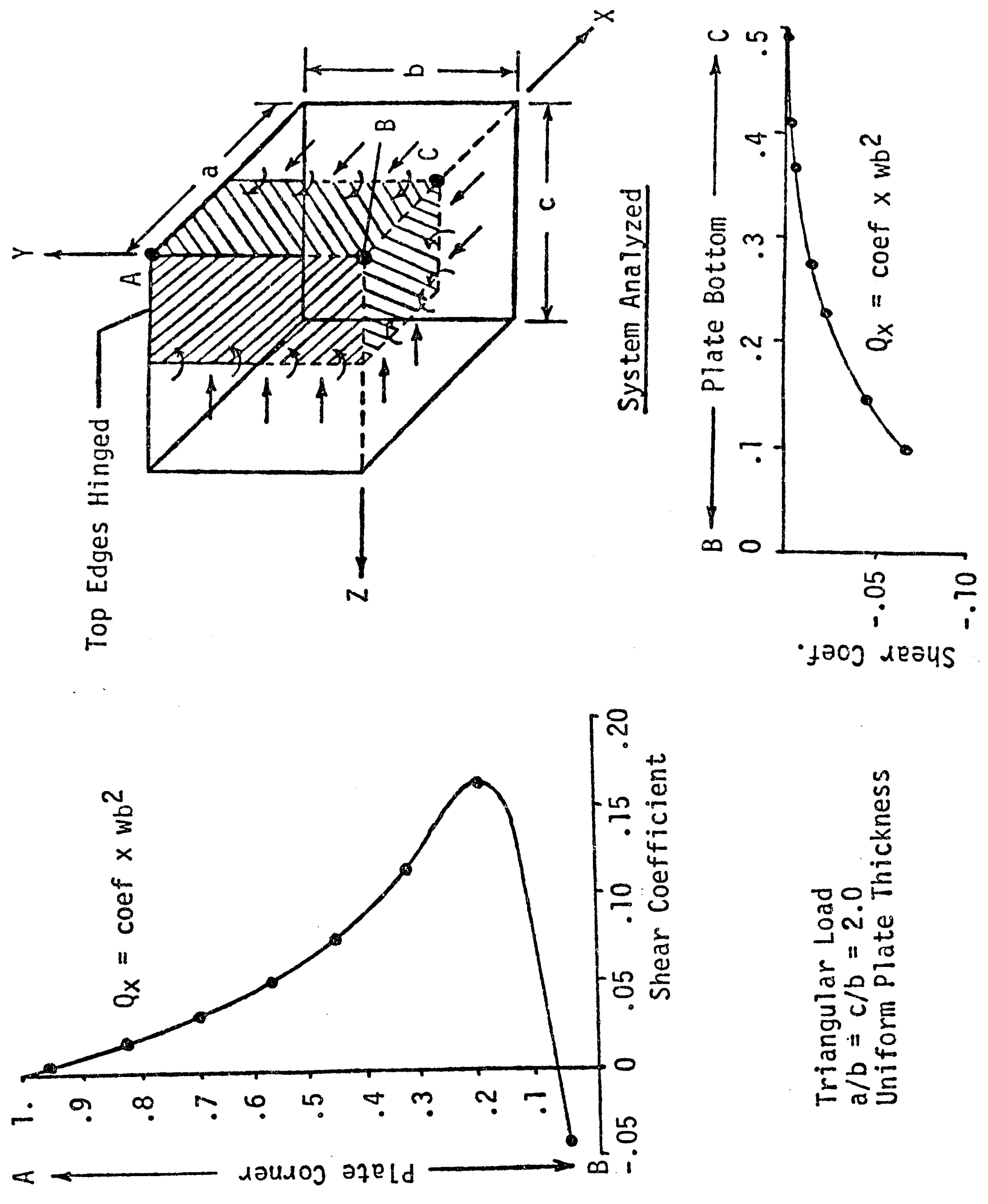


Figure 24: Three Plate Shear Coefficients

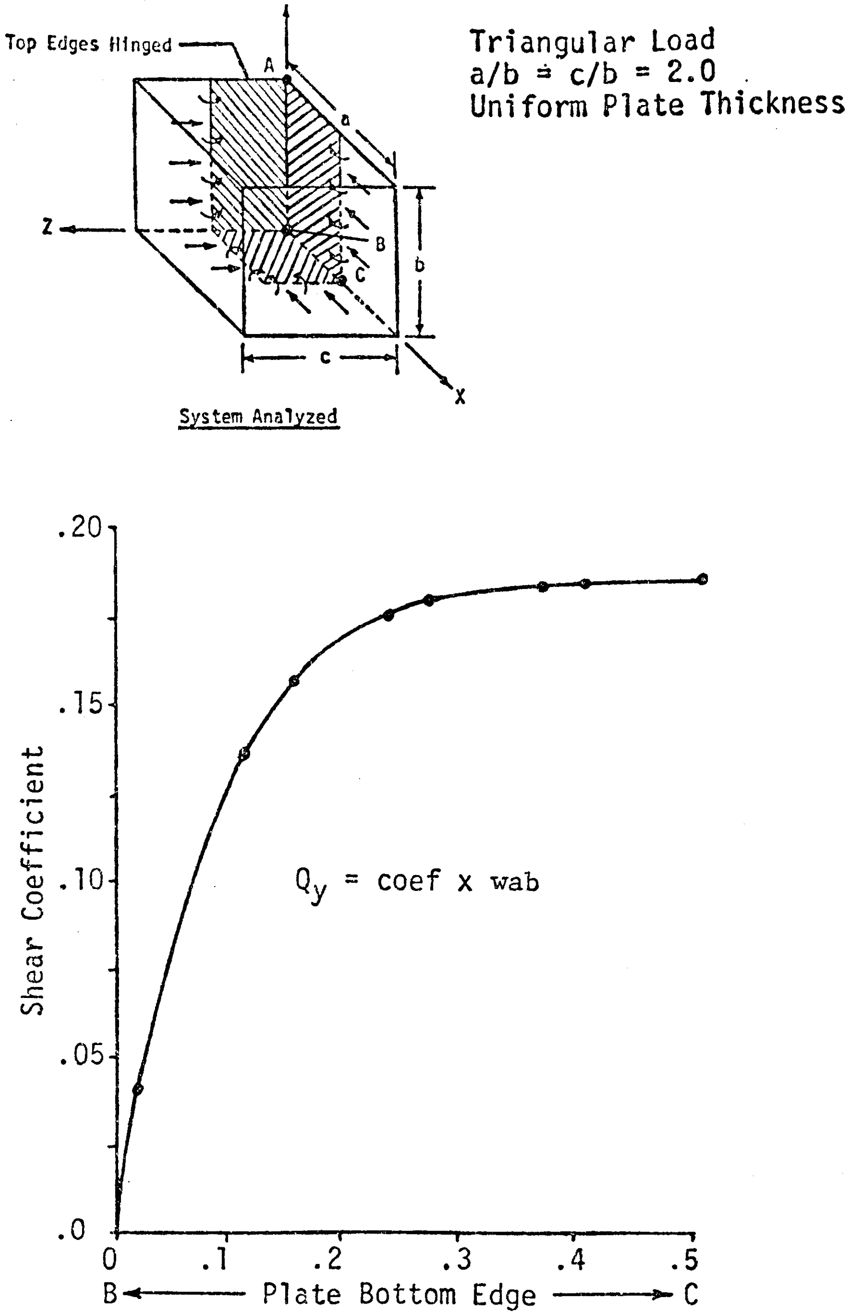


Figure 25: Three Plate Shear Coefficients

VIII CONCLUSIONS & RECOMMENDATIONS

The program described in this paper is capable of determining horizontal and vertical shear resultants at any point within a one, two, or three plate system where plates are joined at ninety degree angles. Program options allow for tapered walls and combinations of both uniform and triangular loads for each type plate problem. The three plate solution process automatically imposes boundary conditions conducive to one quarter of a monolithically constructed rectangular tank system due to tank symmetry. The plate dead weight effects and their associated shear force upon the floor plate are also accounted for when analyzing the three plate problem. Difficulties are encountered in the refinement of plate finite element meshes due to program computer storage requirements. The result is that derived resultant values do correlate with published solutions for one and two plate problems. Three plate problems require at least an 8x8 plate mesh, 8.7 megabits of computer storage, to provide acceptable resultant accuracy.

Program improvement in simulating tank/foundation interplay is recommended. At present the plate weight and the weight of fluid in the tank are not imposed on the base plate of the tank since it was assumed that upward soil pressure would equal these forces. However, the wall weights are distributed over a strip of the footing. These latter loads lead to an upward base plate displacement in the middle of the tank, and consequently some distortion in the expected shear coefficient diagrams. To correct this, it will be necessary to add to the model and program the fluid and base plate weights, as well as a function representing the soil support.

Single plate analysis will show good results. The results will, however, depict a decrease in resultant accuracy as problem complexity increases. This trend is attributed to the finite element mesh refinement which is limited by computer storage capabilities. The one plate results will reflect exact correlations with classical plate solutions for symmetric boundary and loading conditions. Unsymmetric conditions will also incur favorable yet less accurate comparisons. The differences with published moment coefficients increases when analyzing tapered plates but still displays favorable results. The decrease in accuracy is attributed to the requirement for a more refined finite element mesh as the plate problem complexity increases.

The two and three plate analysis will respond similarly to the single plate problems. The dual plate results will display favorable correlation with published PCA (1) solutions utilizing plate 8x8 finite element meshes. Three plate test results will foster expected displacement configurations. The resultants determined are not accurate. Prior testing has shown that 64 element plate meshes (8x8) are a minimum requirement for unsymmetric boundary and/or load conditions. Consequently, the VPI computer storage limitation of 16 elements per plate while analyzing a three plate system will not accommodate acceptable resultant values.

Since the present program's accuracy is limited by finite element mesh refinement, the need for computer storage must be overcome in obtaining accurate results. Therefore, it is recommended that the present program be modified to accommodate computer hardware restriction. The recommended modifications are:

1. Implement a more efficient equation solver. Possible recommendations are the skyline banding method developed by Hughes (31) or an indirect finite element solution method. The indirect method is recommended since it utilizes considerably less storage by only addressing element stiffness matrices. The skyline method is still undergoing verification procedures. It addresses the complete global system matrix, but has shown some inadequacies in dealing with thin plates.

2. Rewrite those program sections addressing mesh development to refine the finite element mesh where accurate resultant values are desired. Finite element experience has shown this procedure to foster more accurate displacements in the refined elements. Consequently, those refined elements will reflect more accurate resultant values.

3. Subdivide the present program into consecutively executed subprograms. Each subprogram should then create a data file. The data files would be utilized as subprogram transitions. This method would curtail overall program storage requirements.

The recommended priority in program modification would be as listed above. The most benefit, however, would come from a combination of all the recommendations.

BIBLIOGRAPHY

1. "Rectangular Concrete Tanks," Bulletin ST-63, Structural Bureau, Portland Cement Association, 1947 (1969).

2. Buchi, J. F., "The Design of Walls of Open Rectangular Tanks," Concrete and Construction Engineering, July, 1951.

3. Young, D., "Analysis of Clamped Rectangular Plates," Journal of Applied Mechanics, Dec, 1940, p. A-139.

4. Craemer, H., "Bending Moments in Box-Shaped Reservoirs," The Structural Engineer, July, 1953, pp. 177-178.

5. Timoshenko, S., and S. Woinowsky-Krieger, S., Theory of Plates and Shells, McGraw-Hill Book Company, Second Edition, New York, 1959.

6. Bares, R. B., Tables for the Analysis of Plates, Slabs and Diaphragms, Library of Congress, CAT # 68-25531, 1969.

7. Ghali, A., "The Structural Analysis of Circular and Rectangular Concrete Tanks," Dissertation presented to the University of Leeds, Leeds, England, 1957.

8. Lightfoot, E., and A. Ghali, "The Analysis of Rectangular Concrete Tanks," Proceedings, 50th Anniversary Conference, Institution of Structural Engineers, 1958.

9. Lightfoot, E., and Sawko, F., "The Analysis of Grid Frameworks and Floor Systems by the Electronic Computer," The Structural Engineer, Mar, 1960, pp. 79-87.

10. Just, D. J., "Analysis of Structures Composed of Single and Connected Flat Plates," Dissertation presented to the University of Leeds, Leeds, England, 1964.

11. Husain, H. M., "The Analysis of Rectangular Plates and Cellular Structures," Dissertation presented to the University of Leeds, Leeds, England, 1964.

12. Davies, J. D., "Bending Moments in Long Rectangular Tanks on Elastic Foundations," Concrete and Constructional Engineering, Vol. 56, No. 10, October, 1961, pp. 335-338.

13. Davies, J. D., "Influence of Support Conditions on the Behavior of Long Rectangular Tanks," Journal, American Concrete Institute, Vol. 59, April, 1962, pp. 601-608.

14. Davies, J. D., "Bending Moments in Edge Supported Square Concrete Tanks," The Structural Engineer, Vol. 40, May, 1962.

15. Davies, J. D., "Bending Moments in Square Concrete Tanks Resting on Flat Rigid Supports," The Structural Engineer, Vol. 41, December, 1963, pp. 407-410.

16. Davies, J. D., "The Influence of Support Conditions on the Behavior of Square Concrete Tanks," Magazine of Concrete Research, Vol. 16, No. 48, Sept, 1964, pp. 153-166.

17. Davies, J. D., "Analysis of Long Rectangular Tanks Resting on Flat Rigid Supports," Journal, American Concrete Institute, Vol. 60, April, 1963, pp. 407-499.

18. Cheung, Y. K., and Zienkiewicz, O. C., "Plates and Tanks on Elastic Foundations - An Applications of Finite Element Method," International Journal of Solids and Structures, Vol. 1, 1965, pp. 451-461.

19. Cheung, Y. K., and J. D. Davies, "Analysis of Rectangular Tanks-Use of Finite Element Technique," Concrete, Vol. 1, May, 1967, pp. 169-174.

20. Davies, J. D., Y. K. Cheung, "Bending Moments on Long Walled Tanks," Journal, American Concrete Institute, Vol. 64, October, 1967.

21. Davies, J. D., Cheung, Y. K., and Gorecki, A., "Analysis of Long Rectangular Tanks," Water and Water Engineering, Dec, 1970, pp. 510-516.

22. Davies, J. D., and Long, J. E., "Behavior of Square Tanks on Elastic Foundations," Journal of the Engineering Mechanics Division, ASCE, Vol. 94, No. EM3, Proc. Paper 5985, June, 1968, pp. 733-772.

23. Jofriet, Jan. C., "Design of Rectangular Concrete Tank Walls," Journal, American Concrete Institute, Vol. 72, July, 1975.

24. Beck, R. L., "Analysis of Short-Walled Rectangular Concrete Tanks," Master's Thesis presented at Virginia Polytechnic Institute, Blacksburg, VA, 1972.

25. Fitzpatrick, D. G., "Analysis of Rectangular Concrete Tanks Considering Interaction of Plate Bending," Master's Thesis presented at Virginia Polytechnic Institute, Blacksburg, VA, 1982.

26. Reddy, J. N., An Introduction to the Finite Element Method, McGraw-Hill Publishing Co., London, 1984.

27. Brenneman, James, "Analysis of Structures Idealized as Rectangular Elements in Combined Flexure and Extension," Master's Thesis presented at Virginia Polytechnic Institute, Blacksburg, VA, 1969.

28. Wilby, C. A., "Structural Analysis of Reinforced Concrete Tanks," Journal of the Structural Division, ASCE, Vol. 103, May, 1977, pp. 989-1004.

29. Bathe, K-J, Finite Element Procedures in Engineering Analysis, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.

30. Pryor, C. W., "Finite Element Analysis of Laminated Anisotropic Plates Including Transverse Shear Deformations," Dissertation presented to Virginia Polytechnic Institute, Blacksburg, VA, 1969.

31. Hughes, T. J., I. Levit, and J. Winget, "An Element-by-Element Solution Algorithm For Problems of Structural and Solid Mechanics," Computer Methods in Applied Mechanics and Engineering, Vol. 36, No. 2, 1983, pp. 241-254.

32. Manual of Steel Construction, American Institute of Steel Construction, Inc., 8th Edition, 1980.

X. APPENDICES

Appendix 1

```
//B0038DGF JOB 42605,DEVENS,REGION=5000K,TIME=(4,0)
/*LONGKEY       SAM
/*PRIORITY      IDLE
/*JOBPARM       LINES=10
//STEP1  EXEC   FORTVCG
//FORT.SYSIN    DD  *
C***********************************************************************
C                   FINIT ELEMENT PROGRAM DOCUMENTATION                *
C***********************************************************************
C
C          THIS PROGRAM WAS DESIGNED TO ANALYZE ONE QUARTER OF A
C     RECTANGULAR CONCRETE TANK UTILIZING SYMMETRY TO REDUCE THE NUMBER
C     OF DEGREES OF FREEDOM OF THE SYSTEM.  HOWEVER, ONE OR TWO PLATE
C     PROBLEMS CAN BE ANALYZED.  THE INPUT DATA HAS BEEN MINIMIZED TO
C     PERMIT SOMEONE UNFAMILIAR WITH THE FINIT ELEMENT METHOD TO USE
C     THE PROGRAM.  SOME OF THE PROGRAM FEATURES INCLUDE:
C
C        ==> AUTOMATIC GENERATION OF THE NODE NUMBERS FOR 1,2, OR 3
C               PLATES GIVEN THE NUMBER OF ELEMENTS IN EACH DIRECTION
C        ==> ALLOWANCE FOR TAPERED WALLS
C        ==> AUTOMATIC GENERATION OF ELEMENT NUMBERS
C        ==> AUTOMATIC ELIMINATION OF SOME SYMMETRY BOUNDARY CONDITIONS
C               FOR 1,2, OR 3 PLATES
C        ==> ALLOWANCE FOR ADDITIONAL BOUNDARY CONDITIONS TO BE
C               PRESCRIBED TO ZERO
C        ==> INCLUSION OF BOTH TRIANGULAR AND UNIFORM LOADINGS
C        ==> INCLUSION OF A MODIFIED STRIP LOADING ON THE FLOOR
C                SLAB TO ACCOUNT FOR DISTRIBUTION OF SHEAR FROM
C                THE WALLS THROUGH THE FLOOR
C        ==> LOADINGS TO BE INTERNAL OR EXTERNAL
C        ==> LOADINGS TO BE AT ARBITRARY HEIGHTS FROM BASE
C        ==> INCLUSION OF SOME TRIGGER CARDS TO PREVENT EXECUTION
C                WITH IMPROPER DATA
C        ==> ALLOWANCE FOR A WINKLER FOUNDATION
C        ==> INCLUSION OF DEAD LOAD FOR WALLS
C
```

Appendix 2

```
C***********************************************************************
C     THE SIMPLIFICATIONS AND ASSUMPTIONS INHERENT IN THIS PROGRAM     *
C***********************************************************************
C
C        ==> RESISTANCE TO ROTATION IN THE NORMAL DIRECTION OF THE
C               PLATE IS ASSUMED TO BE INFINITE AND IS SUBSEQUENTLY
C               ELIMINATED AS A BOUNDARY CONDITION
C        ==> PLATES MUST BE ORTHOGONAL
C        ==> GLOBAL AXES MUST COINCIDE WITH THE PLATES
C        ==> TRIANGULAR LOADS ARE ONLY PERMITTED ON THE WALLS
C        ==> LOADING CONDITIONS ARE APPROXIMATED AS POINT LOADS AT
C               THE NODES
C        ==> UNIFORM LOAD MUST COVER THE FULL WIDTH OF PLATE 3
C        ==> LOADS MUST BE THE FULL LENGTH OF THE WALL
C        ==> THE THICKNESS OF A TAPERED ELEMENT IS APPROXIMATED BY
C               IT'S AVERAGE THICKNESS
C        ==> THE MODULUS OF ELASTICITY AND POISSON'S RATIO ARE THE
C               SAME FOR ALL THREE PLATES
C
```

Appendix 3

```
C***********************************************************************
C     DESCRIPTION ON SOME OF THE VARIABLE NAMES USED IN THE PROGRAM    *
C***********************************************************************
C
C      A,B,C ------ ELEMENT DIMENSIONS IN THE GLOBAL 2,3, AND 1
C                   DIRECTIONS, RESPECTIVELY
C
C      C1 --------- MATERIAL STIFFNESS COEFFICIENTS OF THE ORTHOTROPIC
C                   ELASTICITY
C
C      CONST ------ WEIGHT VALUE * WEIGHT VALUE * DETERMINANT OF
C                   JACOBIAN MATRIX
C
C      D ---------- ORTHOTROPIC PLATE STIFFNESS COEFFICIENTS
C
C      DET -------- DETERMINANT OF JACOBIAN MATRIX
C
C      DSF(I,J) --- DERIVATIVE OF SF(J) WITH RESPECT TO XI IF I=1 AND
C                   ETA IF I=2(XI & ETA ARE LOCAL X & Y AXIS)
C
C      DXW -------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO X
C                   TIMES W DISPLACEMENT
C
C      DXX -------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO X
C                   TIMES SX DISPLACEMENT
C
C      DXY -------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO X
C                   TIMES SY DISPLACEMENT
C
C      DYW -------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO Y
C                   TIMES W DISPLACEMENT
C
C      DYX -------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO Y
C                   TIMES SX DISPLACEMENT
C
C      DYY -------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO Y
C                   TIMES SY DISPLACEMENT
C
```

```
C      E ---------- MODULUS OF ELASTICITY (KSI)- E1,E2
C
C      EF --------- ELEMENT FORCE VECTOR IN TERMS OF LOCAL COORDINATES
C
C      ELSTIFF ---- ELEMENT STIFFNESS MATRIX IN LOCAL COORDINATE SYSTEM
C
C      ELXY ------- STORES X & Y COORDINATES FOR EACH OF NINE NODES
C                   ASSOCIATED WITH ELEMENT BEING ADDRESSED
C                   ELXY(I,1) = X(I) ; ELXY(I,2) = Y(I)
C
C      ETA -------- NETA ON LOCAL COORDINATE AXIS (Y)
C
C      F ---------- INTENSITY OF THE APPLIED LOADS IN APPROPRIATE
C                   DIRECTION
C
C      G12 -------- SHEAR MODULI OF ELASTICITY (KSI)- G13,G23
C
C      GAUSS ------ 3 X 3 GAUSS POINTS
C
C      GAUSS1 ----- 2 X 2 GAUSS POINTS
C
C      GDSF(I,J) -- DERIVATIVE OD SF(J) WITH RESPECT X IF I=1 AND
C                   Y IF I=2
C
C      GF --------- REPRESENTS THE LOAD VECTOR BEFORE SUBROUTINE SOLVE
C                   AND THE DISPLACEMENTS AFTER SUBROUTINE SOLVE
C
C      GJINV ------ INVERSE OF ELEMENT JACOBIAN MATRIX
C
C      GJ --------- JACOBIAN MATRIX FOR EACH ELEMENT
C
C      H ---------- HEIGHT OF APPLIED LOADS (SEE DATA INPUT)
C
C      IHBW ------- HALF BAND WIDTH OF GLOBAL SYSTEM STIFFNESS MATRIX
C
C      IOP -------- A MATRIX OF RANK THREE CONTAINING THE ELEMENT
C                   NUMBERS FOR EACH PLATE
C
```

```
C       JCODE ------ CONTAINS THE NUMBERS OF THE DEGREES OF FREEDOM AT
C                    EACH NODE IN GLOBAL COORDINATES
C
C       LDIR ------- DIRECTION OF APPLIED LOADS (SEE DATA INPUT)
C
C       LTYPE ------ INDICATES LOAD TYPE (SEE DATA INPUT)
C
C       MCODE ------ CONTAINS THE DEGREES OF FREEDOM FOR EACH ELEMENT
C
C       NDOF ------- NUMBER OF DEGREES OF FREEDOM
C
C       NELEM ------ NUMBER OF ELEMENTS
C
C       NLOAD ------ NUMBER OF APPLIED LOADS / NOT COUNTING GRAVITATIONAL
C                    OR STRIP AUTOMATICALLY IMPOSED LOADS
C
C       NOD -------- A MATRIX OF RANK TWO CONTAINING THE NODE NUMBERS
C                    FOR EACH ELEMENT
C
C       NOP -------- A MATRIX OF RANK THREE CONTAINING THE NODE
C                    NUMBERING SCHEME FOR EACH PLATE
C
C       NNODES ----- NUMBER OF NODES
C
C       NP --------- NUMBERING OF NODES 1 TO 9 IN LOCAL FRAME OF
C                    REFERENCE
C
C       NPL -------- PLATE NUMBER LOAD APPLIED ON
C
C       NPLTS ------ FLAG INDICATING THE NUMBER OF PLATES BEING
C                    ANALYZED
C
C       NX,NY,NZ --- NUMBER OF ELEMENTS IN THE GLOBAL 1,2, AND 3
C                    DIRECTIONS, RESPECTIVELY
```

```
C
C      SF(I) ------ INTERPOLATION FUNCTION FOR NODE I OF ELEMENT
C
C      SOIL ------- EQUIVALENT SPRING STIFFNESS OF THE FOUNDATION
C                   AT AN INTERNAL NODE (KIPS/IN)
C
C      SS(1) ------ NORMAL IN-PLANE (NX) RESULTANT - KIPS/INCH
C
C      SS(2) ------ NORMAL IN-PLANE (NY) RESULTANT - KIPS/INCH
C
C      SS(3) ------ IN-PLANE SHEARING (NXY & NYX) RESULTANT - KIPS/INCH
C
C      SS(4) ------ BENDING MOMENT (MX) RESULTANT - KIP-INCH
C
C      SS(5) ------ BENDING MOMENT (MY) RESULTANT - KIP-INCH
C
C      SS(6) ------ TWISTING MOMENT (MXY) RESULTANT - KIP-INCH
C
C      SS(7) ------ BENDING SHEAR (QX) RESULTANT - KIPS/INCH
C
C      SS(8) ------ BENDING SHEAR (QY) RESULTANT - KIPS/INCH
C
C      SST -------- CONTAINS THE GLOBAL STIFFNESS MATRIX STORED IN
C                   HALF-BANDED FORM THAT CAN BE USED BY THE LINPACK
C                   EQUATION SOLVER
C
C      SXI -------- INTERPOLATION FUNCTION TIMES SX DISPLACEMENT
C
C      SYI -------- INTERPOLATION FUNCTION TIMES SY DISPLACEMENT
C
C      THK -------- CONTAINS THE STEPPED THICKNESSES FOR EACH PLATE
C
C      THKF ------- THICKNESS OF THE FLOOR
C
C      THKSB,THKST- THICKNESS OF PLATE 1, BOTTOM AND TOP RESPECTIVELY
C
C      THKLB,THKLT- THICKNESS OF PLATE 2, BOTTOM AND TOP RESPECTIVELY
C
```

```
C     UX --------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO X
C                  TIMES U DISPLACEMENT
C
C     UY --------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO Y
C                  TIMES U DISPLACEMENT
C
C     VNU -------- POISSON'S RATIO
C
C     VX --------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO X
C                  TIMES V DISPLACEMENT
C
C     VY --------- PARTIAL OF INTERPOLATION FUNCTION WITH RESPECT TO Y
C                  TIMES V DISPLACEMENT
C
C     W ---------- WEIGHT OF LOAD APPLIED TO PLATE (SEE DATA INPUT)
C
C     WC --------- SPECIFIC WEIGHT OF CONCRETE
C
C     WT --------- WEIGHT VALUES ASSOCIATED WITH 3 X 3 GAUSS POINTS
C
C     WT1 -------- WEIGHT VALUES ASSOCIATED WITH 2 X 2 GAUSS POINTS
C
C     X,Y,Z ------ DIMENSIONS OF THE SINGLE PLATE PROBLEM BEING
C                  ANALYZED DETERMINED BY THE BOUNDARY CONDITIONS
C                  BEING USED OR THE DIMENSIONS OF THE SYMMETRIC
C                  PORTION OF THE TWO OR THREE PLATE SYSTEM
C
C     X(I) ------- GLOBAL COORDINATE NUMBER ALONG X AXIS (PLATES 1 & 2)
C
C     X3(I) ------ GLOBAL COORDINATE NUMBER ALONG X AXIS (PLATE 3)
C
C     XI --------- ZETA ON LOCAL COORDINATE AXIS (X)
C
C     XNODE ------ LOCAL COORDINATES OF ELEMENTS (-1 TO 1)
C
C     Y(I) ------- GLOBAL COORDINATE NUMBER ALONG Y AXIS (PLATES 1 & 2)
C
C     Z3(I) ------ GLOBAL COORDINATE NUMBER ALONG Z AXIS (PLATE 3)
C
```

Appendix 4

```
C***********************************************************************
C                    INPUT OF DATA - UNFORMATTED                       *
C***********************************************************************
C
C  CARD 1 --------- NPLTS                                        (1I)
C
C            ENTER 1,2,3 FOR THE NUMBER OF PLATES TO BE ANALYZED
C
C  CARD 2 --------- X,Y,Z,NX,NY,NZ                               (3R,3I)
C
C            X - DIMENSION IN THE GLOBAL 1-DIRECTION (INCHES)
C            Y - DIMENSION IN THE GLOBAL 2-DIRECTION (INCHES)
C            Z - DIMENSION IN THE GLOBAL 3-DIRECTION (INCHES)
C              *** NOTE: ENTER THE DIMENSIONS THAT CORRESPOND TO THE
C                        BOUNDARY CONDITIONS THAT ARE APPLIED TO THE
C                        PLATE OR PLATES
C                 MUST ENTER Z=0. IF ONLY DOING A SINGLE PLATE PROBLEM
C            NX - NUMBER OF ELEMENTS IN THE GLOBAL 1-DIRECTION
C            NY - NUMBER OF ELEMENTS IN THE GLOBAL 2-DIRECTION
C            NZ - NUMBER OF ELEMENTS IN THE GLOBAL 3-DIRECTION
C                  (MAXIMUM NUMBER OF ELEMENTS IN ANY DIRECTION
C                   IS EIGHT)
C
C  CARD 3 --------- E,VNU,WC,SOIL                                (4R)
C
C            E - MODULUS OF ELASTICITY OF CONCRETE (KSI)
C            VNU - POISSON'S RATIO OF CONCRETE
C            WC - SPECIFIC WEIGHT OF CONCRETE (PCF)
C            SOIL - ESTIMATED SPRING STIFFNESS OF THE SOIL (K/IN)
C
C  CARD 4 -------- THKST,THKSB,THKLT,THKLB,THKF                  (5R)
C
C            THKST - THICKNESS AT THE TOP OF PLATE 1 IN THE GLOBAL
C                      1-2 PLANE (INCHES)
C            THKSB - THICKNESS AT THE BOTTOM OF PLATE 1 IN THE
C                      GLOBAL 1-2 PLANE (INCHES)
```

```
C              THKLT - THICKNESS AT THE TOP OF PLATE 2 IN THE GLOBAL
C                        2-3 PLANE (INCHES)
C                           ENTER 0.0 FOR A SINGLE PLATE PROBLEM
C              THKLB - THICKNESS AT THE BOTTOM OF PLATE 2 IN THE
C                        GLOBAL 2-3 PLANE (INCHES)
C                           ENTER 0.0 FOR A SINGLE PLATE PROBLEM
C              THKF - THICKNESS OF THE FLOOR SLAB (INCHES)
C                           ENTER 0.0 FOR 1 OR 2 PLATE PROBLEM
C
C  5TH GROUP OF -- NOD,NDIR                                   (2I)
C    CARDS
C
C              NOD - NODE NUMBER AT WHICH A CONSTRAINT EXISTS
C              NDIR - GLOBAL DIRECTION OF THE CONSTRAINT, BOTH
C                       DISPLACEMENT AND ROTATION CONSTRAINTS
C                       ARE POSSIBLE
C          *** MUST ENTER  0 0  AS A TRIP CARD TO SIGNIFY THE END
C                     OF THE JOINT CONSTRAINTS ***
C
C  6TH GROUP OF -- LTYPE,NPL,LDIR,W,H                         (3I,2R)
C    CARDS
C
C              LTYPE - INDICATES THE TYPE OF LOADING
C                        ENTER 1 FOR TRIANGULAR LOAD
C                        ENTER 2 FOR UNIFORM LOAD
C              NPL - NUMBER OF THE PLATE TO WHICH THE LOAD IS APPLIED
C                        ENTER 0 FOR LTYPE=3
C                        ENTER 1 FOR PLATE IN THE 1-2 PLANE(PLATE 1)
C                        ENTER 2 FOR PLATE IN THE 2-3 PLANE(PLATE 2)
C                        ENTER 3 FOR THE FLOOR SLAB(PLATE 3)
C              LDIR - INDICATES THE LOCAL DIRECTION THAT THE LOAD IS
C                        APPLIED.  A NEGATIVE SIGN SHOULD BE ENTERED
C                        WITH THIS VALUE ONLY TO INDICATE THAT THE
C                        LOAD IS APPLIED OPPOSITE TO THE POSITIVE SENSE
C                        OF THE LOCAL DIRECTIONS.
```

```
C           W - MAGNITUDE OF THE LOADING, ALWAYS POSITIVE
C                     IF LTYPE=1, W REPRESENTS THE SPECIFIC WEIGHT OF
C                         THE LOADING (PCF)
C                     IF LTYPE=2, W REPRESENTS THE PRESSURE ON THE
C                         PLATE (PSF)
C                        *** NOTE: IF A SINGLE PLATE PROBLEM IS BEING
C                              ANALYZED TO DETERMINE THE FIXED END
C                              MOMENTS ON THE FLOOR SLAB, W REPRESENTS
C                              THE WEIGHT OF THE WALLS.
C           H - THE HEIGHT OF THE LOADING ON THE WALL.  THIS VALUE
C                     WILL GENERALLY BE THE FULL HEIGHT OF THE WALL
C                     BUT CAN INCLUDE ANY ARBITRARY HEIGHT.  FOR THE
C                     FLOOR, THE FULL DIMENSION (IN THE X-DIRECTION)
C                     MUST BE ENTERED. (INCHES)
C      *** MUST ENTER  0 0 0 0. 0.  AS A TRIP CARD TO INDICATE
C             THE END OF THE PLATE LOADINGS ***
C
```

Appendix 5

```
C*********************************************************************
C                              MAIN PROGRAM                          *
C*********************************************************************
C     SUPERVISES THE SEQUENTIAL CALLING OF MAJOR SUBROUTINES.
C
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER D
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/SOLV/MAXID,LDA
C
      READ(5,*) NPLTS
      WRITE(6,1)
    1 FORMAT(' * TWO PLATES (4X4) / UNIF-NO DEADWT / 100,10,.2')
      WRITE(6,2)
    2 FORMAT('                    / BOT&TOP - SUPPORTED (NOT FIXED)'/)
      LDA=500.0
      CALL DATA(NPLTS)
      CALL GEN(NPLTS)
      CALL GLOBXY(NPLTS)
      CALL PROCES(NPLTS)
      CALL GSFORC(NPLTS)
      CALL GSSTIF(NPLTS)
      CALL SOLVE
      CALL TENFOR(NPLTS)
      STOP
      END
```

```
C***********************************************************************
C                            SUBROUTINE DATA                           *
C***********************************************************************
C     READ AND ECHO X,Y,Z,NX,NY,NZ,E,VNU,WC,SOIL WHILE PRINTING AN ERROR
C     MESSAGE IF THE MAXIMUM NUMBER OF ELEMENTS IS EXCEEDED.  COMPUTE A,
C     B, & C.  CALL SUBROUTINE THICK AND PRINT PLATE THICKNESSES AT
C     SPECIFIC POINTS.
C
      SUBROUTINE DATA(NPLTS)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/TC/THKST,THKSB,THKLT,THKLB,THKF
      COMMON/COEFF/SOIL
C
      READ(5,*) X,Y,Z,NX,NY,NZ
      IF(NX.LE.8.OR.NY.LE.8.OR.NZ.LE.8) GO TO 50
         WRITE(6,100)
         STOP
   50 A=Y/DFLOAT(NY)
      IF(NPLTS.NE.1) B=Z/DFLOAT(NZ)
      C=X/DFLOAT(NX)
      READ(5,*) E,VNU,WC,SOIL
      READ(5,*) THKST,THKSB,THKLT,THKLB,THKF
      E1=E
      E2=E
      G12=E/(2.0D0*(1.0D0+VNU))
      G13=G12
      G23=G12
      DO 10 I=1,NPLTS
         GOTO(11,12,13),I
   11       CALL THICK(THKST,THKSB,NY,I)
            GO TO 10
   12       CALL THICK(THKLT,THKLB,NY,I)
            GO TO 10
   13       CALL THICK(THKF,THKF,NX,I)
```

```
 10 CONTINUE
    WRITE(6,101) X,C,Y,A,Z,B
    WRITE(6,201) E,VNU,WC,SOIL
    DO 20 I=1,NPLTS
       WRITE(6,105)
       GOTO(21,21,22),I
 21       J=NY
          GO TO 25
 22       J=NX
 25    IF(NPLTS.EQ.1) WRITE(6,103) THKSB,THKF
 20 IF(NPLTS.NE.1) WRITE(6,102) I,(THK(K,I),K=1,J)
100 FORMAT(' YOU HAVE EXCEEDED THE MAXIMUM NUMBER OF ELEMENTS IN THE X
   *-,Y-,OR Z-DIRECTION.  THE MAXIMUM NUMBER OF ELEMENTS AVAILABLE IN
   *THIS PROGRAM IS 8.')
101 FORMAT(//'  WALL SIZE',6X,'ELEMENT SIZE'/
   *' X=',F9.2,6X,'C=',F9.2/' Y=',F9.2,6X,'A=',F9.2/
   *' Z=',F9.2,6X,'B=',F9.2/)
102 FORMAT(' STEPPED THICKNESS FOR PLATE',I2,2X,'(INCHES)'/8F8.1)
103 FORMAT(' THE THICKNESS OF THE WALLS =',F8.1,' INCHES'/
   *' THE THICKNESS OF THE FLOOR PLATE =',F8.1,' INCHES')
105 FORMAT(/)
201 FORMAT(' MODULUS OF ELASTICITY-------',F8.2,1X,'(KSI)'/
   *' POISSONS RATIO--------------',F8.2/
   *' SPECIFIC WEIGHT OF CONCRETE-',F8.2,1X,'(PCF)'/
   *' SOIL STIFFNESS--------------',F8.2,1X,'(KIPS/INCH)'/)
    RETURN
    END
```

```
C***********************************************************************
C                              SUBROUTINE THICK                        *
C***********************************************************************
C     COMPUTE AVERAGE ELEMENT PLATE THICKNESSES.
C
      SUBROUTINE THICK(T,B1,NR,J)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
C
      IF(T.EQ.B1) GO TO 20
      SLOPE=(B1-T)/DFLOAT(NR)/2.0D0
      DO 10 I=1,NR
         M=2.0D0*DFLOAT(I)-1.0D0
   10    THK(I,J)=T+M*SLOPE
      RETURN
   20 DO 30 I=1,NR
   30    THK(I,J)=T
      RETURN
      END
```

```
C***********************************************************************
C                              SUBROUTINE GEN                          *
C***********************************************************************
C     GENERATE AND PRINT BOTH ELEMENT AND NODAL NUMBERS FOR 1 TO 3
C     PLATES.  CALCULATE NUMBER OF NODES.
C
      SUBROUTINE GEN(NPLTS)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
C
      N=0
      NX1=2*NX+1
      NY1=2*NY+1
      NZ1=2*NZ+1
      DO 10 I=1,NY1
         DO 20 J=1,NX1
            N=N+1
   20       NOP(I,J,1)=N
            IF(NPLTS.EQ.1) GO TO 10
               N=N-1
               DO 30 K=1,NZ1
                  N=N+1
   30             NOP(I,K,2)=N
   10 CONTINUE
      NYY=2*NY
      IF(NPLTS.NE.3) GO TO 90
         N=NOP(NYY,NZ1,2)
         DO 60 I=1,NX1
         DO 60 J=1,NZ1
            N=N+1
   60       NOP(I,J,3)=N
         DO 40 I=1,NX1
   40       NOP(NY1,I,1)=NOP(I,1,3)
         DO 50 I=1,NZ1
   50       NOP(NY1,I,2)=NOP(NX1,I,3)
   90 NNODES=N
```

```
      M=0
      DO 210 I=1,NY
         DO 220 J=1,NX
            M=M+1
220         IOP(I,J,1)=M
         IF(NPLTS.EQ.1) GO TO 210
            DO 230 K=1,NZ
               M=M+1
230            IOP(I,K,2)=M
210   CONTINUE
      IF(NPLTS.NE.3) GO TO 150
         DO 240 I=1,NX
         DO 240 J=1,NZ
            M=M+1
240         IOP(I,J,3)=M
150   NELEM=M
      WRITE(6,111)
      WRITE(6,300)
      NNN=1
      WRITE(6,400) NNN
      DO 100 I=1,NYY
         WRITE(6,110) (NOP(I,J,1),J=1,NX1)
         IF(I.EQ.2)GO TO 91
         IF(I.EQ.4)GO TO 91
         IF(I.EQ.6)GO TO 91
         IF(I.EQ.8)GO TO 91
         IF(I.EQ.10)GO TO 91
         IF(I.EQ.12)GO TO 91
         IF(I.EQ.14)GO TO 91
         IF(I.EQ.16)GO TO 91
         GO TO 100
 91      NYX=I/2
         WRITE(6,310) (IOP(NYX,J,1),J=1,NX)
100   CONTINUE
      WRITE(6,110) (NOP(NY1,J,1),J=1,NX1)
      IF(NPLTS.EQ.1) RETURN
         WRITE(6,111)
         NNN=2
```

```
      WRITE(6,400) NNN
      DO 101 I=1,NYY
         WRITE(6,110) (NOP(I,J,2),J=1,NZ1)
         IF(I.EQ.2)GO TO 92
         IF(I.EQ.4)GO TO 92
         IF(I.EQ.6)GO TO 92
         IF(I.EQ.8)GO TO 92
         IF(I.EQ.10)GO TO 92
         IF(I.EQ.12)GO TO 92
         IF(I.EQ.14)GO TO 92
         IF(I.EQ.16)GO TO 92
         GO TO 101
92       NYX=I/2
         WRITE(6,310) (IOP(NYX,J,2),J=1,NZ)
101   CONTINUE
      WRITE(6,110) (NOP(NY1,J,2),J=1,NZ1)
      IF(NPLTS.EQ.2) RETURN
         NNN=3
         NXX=2*NX
         WRITE(6,400) NNN
         DO 102 I=1,NXX
            WRITE(6,110) (NOP(I,J,3),J=1,NZ1)
            IF(I.EQ.2)GO TO 93
            IF(I.EQ.4)GO TO 93
            IF(I.EQ.6)GO TO 93
            IF(I.EQ.8)GO TO 93
            IF(I.EQ.10)GO TO 93
            IF(I.EQ.12)GO TO 93
```

```
            IF(I.EQ.14)GO TO 93
            IF(I.EQ.16)GO TO 93
            GO TO 102
 93         NXY=I/2
            WRITE(6,310) (IOP(NXY,J,3),J=1,NZ)
102       CONTINUE
          WRITE(6,110) (NOP(NX1,J,3),J=1,NZ1)
111 FORMAT(/)
110 FORMAT(17I5)
310 FORMAT(T90,8I4)
300 FORMAT(' NODE NUMBERS',T90,'ELEMENT NUMBERS'/)
400 FORMAT('  PLATE',I2/)
    RETURN
    END
```

```
C*********************************************************************
C                           SUBROUTINE GLOBXY                        *
C*********************************************************************
C     GENERATE RELATIONSHIPS BETWEEN ELEMENTS AND THEIR RESPECTIVE NODES
C     WHILE IDENTIFYING NODES TO GLOBAL COORDINATES.
C
      SUBROUTINE GLOBXY(NPLTS)
      IMPLICIT REAL*8 (A-H,O-Z)
      INTEGER D
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/MSH/X1(250),Y1(250),X3(250),Z3(250),ELXY(9,2),NOD(48,9)
      COMMON/N/NOD1(48,9)
C
      NX1=2*NX+1
      NY1=2*NY+1
      NZ1=2*NZ+1
C
C     PLATE 1 ELEMENT TO NODE RELATIONSHIP
C
      D=NX
      IF(NPLTS .NE. 1)D=NX+NZ
      NOD(1,1)=4*D+5
      NOD(1,2)=4*D+3
      NOD(1,3)=1
      NOD(1,4)=3
      NOD(1,5)=4*D+4
      NOD(1,6)=2*D+2
      NOD(1,7)=2
      NOD(1,8)=2*D+4
      NOD(1,9)=2*D+3
      IF(NY .EQ. 1)GO TO 35
         M=1
         DO 30 K=2,NY
            L=(K-1)*D+1
            DO 20 I=1,9
 20            NOD(L,I)=NOD(M,I)+4*D+2
 30      M=L
 35      DO 60 NI=2,NX
```

```
            DO 40 I=1,9
 40            NOD(NI,I)=NOD(NI-1,I)+2
            M=NI
            DO 60 NJ=2,NY
               L=(NJ-1)*D+NI
               DO 50 J=1,9
 50               NOD(L,J)=NOD(M,J)+4*D+2
 60      M=L
      IF(NPLTS .EQ. 1) GO TO 67
C
C     PLATE 2 ELEMENT TO NODE RELATIONSHIP
C
      EE=NX+1
      NOD(EE,1)=NX1+4*D+2
      NOD(EE,2)=NX1
      NOD(EE,3)=NX1+2
      NOD(EE,4)=NX1+4*D+4
      NOD(EE,5)=NX1+2*D+1
      NOD(EE,6)=NX1+1
      NOD(EE,7)=NX1+2*D+3
      NOD(EE,8)=NX1+4*D+3
      NOD(EE,9)=NX1+2*D+2
      IF(NY .EQ. 1) GO TO 63
      M=EE
      DO 62 K=2,NY
         L=(K-1)*D+NX+1
         DO 61 I=1,9
 61         NOD(L,I)=NOD(M,I)+4*D+2
 62   M=L
 63   DO 66 NI=2,NZ
         NK=NI+NX
         DO 64 I=1,9
 64         NOD(NK,I)=NOD(NK-1,I)+2
         M=NK
         DO 66 NJ=2,NY
            L=(NJ-1)*D+NK
            DO 65 J=1,9
 65            NOD(L,J)=NOD(M,J)+4*D+2
 66   M=L
```

```
C
C     PLATE 1 & 2 NODE TO GLOBAL COORDINATE RELATIONSHIPS
C
   67 D=NX1
      IF(NPLTS .NE. 1)D=NX1+NZ1-1
      YC=0.0D0
      DO 80 NI=1,NY1
         J=NY1*D
         IF(NPLTS .NE. 1)J=NY1*D-(NZ1-1)
         I=J-D*(NI-1)+1
         XC=0.0D0
         DO 70 NJ=1,NX1
            I=I-1
            X1(I)=XC
            Y1(I)=YC
   70    XC=XC+C/2.0D0
   80 YC=YC+A/2.0D0
      IF(NPLTS .EQ. 1)GO TO 230
      YC=0.0D0
      DO 100 NI=1,NY1
         I=J-(NX1+NZ1-1)*(NI-1)-1
         XC=0.0D0
         DO 90 NJ=1,NZ1
            I=I+1
            X1(I)=XC
            Y1(I)=YC
   90 XC=XC+B/2.0D0
  100 YC=YC+A/2.0D0
      IF(NPLTS .EQ. 2)GO TO 230
C
C     PLATE 3 ELEMENT TO NODE RELATIONSHIP
C
      EE=(NX+NZ)*NY
      EN=NOD(EE,7)
      E=EE+1
      NOD1(E,1)=EN+4*NZ+3
      NOD1(E,2)=EN+4*NZ+5
      NOD1(E,3)=EN+3
```

```
      NOD1(E,4)=EN+1
      NOD1(E,5)=EN+4*NZ+4
      NOD1(E,6)=EN+2*NZ+4
      NOD1(E,7)=EN+2
      NOD1(E,8)=EN+2*NZ+2
      NOD1(E,9)=EN+2*NZ+3
      IF(NX .EQ. 1)GO TO 130
      M=E
      DO 120 K=2,NX
         L=(K-1)*NZ+E
         DO 110 I=1,9
110         NOD1(L,I)=NOD1(M,I)+2*NZ1
120   M=L
130   IF(NZ .EQ. 1)GO TO 170
      DO 160 NI=2,NZ
         NK=NI+E-1
         DO 140 I=1,9
140         NOD1(NK,I)=NOD1(NK-1,I)+2
         M=NK
         DO 160 NJ=2,NX
            L=(NJ-1)*NZ+NK
            DO 150 J=1,9
150            NOD1(L,J)=NOD1(M,J)+2*NZ1
160   M=L
170   BWE=NY*(NX+NZ)
      BFE=NY*(NX+NZ)+NX*NZ
      DO 190 I=1,NZ
         X1(NOD1(BFE,2))=X1(NOD(BWE,4))
         Y1(NOD1(BFE,2))=Y1(NOD(BWE,4))
         X1(NOD1(BFE,5))=X1(NOD(BWE,8))
         Y1(NOD1(BFE,5))=Y1(NOD(BWE,8))
         X1(NOD1(BFE,1))=X1(NOD(BWE,1))
         Y1(NOD1(BFE,1))=Y1(NOD(BWE,1))
```

```
         NOD(BWE,4)=NOD1(BFE,2)
         NOD(BWE,8)=NOD1(BFE,5)
         NOD(BWE,1)=NOD1(BFE,1)
         BWE=BWE-1
190      BFE=BFE-1
      BFE=BFE+1
      DO 180 I=1,NX
         X1(NOD1(BFE,1))=X1(NOD(BWE,1))
         Y1(NOD1(BFE,1))=Y1(NOD(BWE,1))
         X1(NOD1(BFE,8))=X1(NOD(BWE,5))
         Y1(NOD1(BFE,8))=Y1(NOD(BWE,5))
         X1(NOD1(BFE,4))=X1(NOD(BWE,2))
         Y1(NOD1(BFE,4))=Y1(NOD(BWE,2))
         NOD(BWE,1)=NOD1(BFE,1)
         NOD(BWE,5)=NOD1(BFE,8)
         NOD(BWE,2)=NOD1(BFE,4)
         BWE=BWE-1
180      BFE=BFE-NZ
191   DO 192 J=1,9
192      NOD(E,J)=NOD1(E,J)
      E=E+1
      IF(E.LE.NELEM)GO TO 191
```

```
C
C     PLATE 3 NODE TO GLOBAL COORDINATE RELATIONSHIP
C
      XC=0.0D0
      DO 220 NI=1,NX1
         J=(NY1-1)*(NX1+NZ1-1)+NZ1*(NX1-1)
         I=J-NZ1*(NI-1)
         ZC=0.0D0
         DO 210 NJ=1,NZ1
            I=I+1
            X3(I)=XC
            Z3(I)=ZC
 210     ZC=ZC+B/2.0D0
 220  XC=XC+C/2.0D0
 230  A13=0.0D0
C     WRITE(6,270)
C     DO 300 I=1,NELEM
C        DO 300 J=1,9
C 300       WRITE(6,280) I,J,NOD(I,J),X1(NOD(I,J)),Y1(NOD(I,J)),
C    *                   X3(NOD(I,J)),Z3(NOD(I,J))
C 270 FORMAT(' ELEMENT',5X,'ELEMENT NODE',5X,'NODE #',7X,'X',7X,'Y',
C    *7X,'X3',7X,'Z3'/)
C 280 FORMAT(3X,I4,10X,I4,10X,I4,5X,F6.1,2X,F6.1,2X,F6.1,2X,F6.1/)
      RETURN
      END
```

```
C***********************************************************************
C                                   SUBROUTINE PROCES                  *
C***********************************************************************
C     GENERATE AND PRINT THE JCODE WITH IMPOSED JOINT CONSTRAINTS
C     FOLLOWED BY THE CONSTRUCTION AND PRINTING OF THE SYSTEM MCODE.
C     COMPUTE THE HALF BAND WIDTH, NUMBER OF ELEMENTS, AND NUMBER OF
C     DEGREES OF FREEDOM.
C
      SUBROUTINE PROCES(NPLTS)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/CODE/JCODE(220,6)
      COMMON/SOLV/MAXID,LDA
      DIMENSION IN(9)
C
C     JOINT CODE CONSTRUCTION
C
      DO 10 I=1,NNODES
      DO 10 J=1,6
   10    JCODE(I,J)=1
C
C     JOINT CONSTRAINTS
C
      NXX=2*NX-1
      NYX=2*NY-1
      IF(NPLTS .GT. 1)NZX=2*NZ-1
      NX1=2*NX+1
      NY1=2*NY+1
      IF(NPLTS .GT. 1)NZ1=2*NZ+1
      GOTO(201,202,203),NPLTS
  203    DO 22 I=1,NX1
         DO 22 J=1,NZ1
            M=NOP(I,J,3)
```

```
C
C     PLATE 3 BOUNDARY (X, Z=1, NZ1) VERTICAL CONSTRAINT
C
         DO 28 J=1, NZ1
            M=NOP(NX1,J,3)
   28       JCODE(M,2)=0
C
C     PLATE 1 & 3 CORNER (Z, X=1, NX1) 90 DEGREE ANGLE
C
         DO 31 J=1, NX1
            M=NOP(J,1,3)
   31       JCODE(M,5)=0
C
C     PLATE 3 IN-PLANE SLOPE
C
   22       JCODE(M,4)=0
         DO 25 I=1,NX1
            M=NOP(I,NZ1,3)
C
C     PLATE 3 OUTSIDE EDGE (Z, X=1, NX1) IN-PLANE DISP & SLOPE
C
            JCODE(M,3)=0
   25       JCODE(M,5)=0
         DO 26 J=1,NZ1
            M=NOP(1,J,3)
C
C     PLATE 3 OUTSIDE EDGE (X, Z=1, NZ1) IN-PLANE DISP & SLOPE
C
            JCODE(M,1)=0
   26       JCODE(M,6)=0
C
C     PLATE 3 BOUNDARY (Z, X=1, NX1) VERTICAL CONSTRAINT
C
         DO 27 J=1, NX1
            M=NOP(J,1,3)
   27       JCODE(M,2)=0
```

```
C
C     PLATE 2 & 3 CORNER (X, Z=1, NZ1) 90 DEGREE ANGLE
C
      DO 32 J=1, NZ1
         M=NOP(NX1,J,3)
   32    JCODE(M,6)=0
  202 DO 21 I=1,NY1
      DO 21 J=1,NZ1
         M=NOP(I,J,2)
C
C     PLATE 2 IN-PLANE SLOPE
C
   21    JCODE(M,5)=0
      DO 23 I=1,NY1
         M=NOP(I,1,1)
C
C     PLATE 1 OUTSIDE EDGE IN-PLANE DISP & SLOPE
C
         JCODE(M,1)=0
   23    JCODE(M,4)=0
      DO 24 I=1,NY1
         M=NOP(I,NZ1,2)
C
C     PLATE 2 OUTSIDE EDGE IN-PLANE DISP & SLOPE
C
         JCODE(M,3)=0
   24    JCODE(M,4)=0
C
C     PLATE 1 & 2 CORNER (X, Y=1, NY1) 90 DEGREE ANGLE
C
      DO 29 J=1, NY1
         M=NOP(J,NX1,1)
   29    JCODE(M,4)=0
  201 DO 20 I=1,NY1
      DO 20 J=1,NX1
         M=NOP(I,J,1)
C
C     PLATE 1 IN-PLANE SLOPE
C
   20    JCODE(M,6)=0
```

```
C
C     PLATE 1 IN-PLANE SLOPE
C
   20       JCODE(M,6)=0
C
   30 READ(5,*) NOD,NDIR
      IF(NOD.EQ.0) GO TO 35
         JCODE(NOD,NDIR)=0
         GO TO 30
   35 NDOF=0
      DO 36 I=1,NNODES
      DO 36 J=1,6
         IF(JCODE(I,J).EQ.0) GO TO 36
            NDOF=NDOF+1
            JCODE(I,J)=NDOF
   36    CONTINUE
      I1=1
      DO 42 M=1,NYX,2
      J1=1
      DO 40 N=1,NXX,2
         I=NOP(M+2,N+2,1)
         J=NOP(M+2,N,1)
         K=NOP(M,N,1)
         L=NOP(M,N+2,1)
         M1=NOP(M+2,N+1,1)
         N1=NOP(M+1,N,1)
         O=NOP(M,N+1,1)
         P=NOP(M+1,N+2,1)
         Q=NOP(M+1,N+1,1)
         NN=IOP(I1,J1,1)
         DO 41 NM=1,6
```

```
                MCODE(NN,NM)=JCODE(I,NM)
                MCODE(NN,NM+6)=JCODE(J,NM)
                MCODE(NN,NM+12)=JCODE(K,NM)
                MCODE(NN,NM+18)=JCODE(L,NM)
                MCODE(NN,NM+24)=JCODE(M1,NM)
                MCODE(NN,NM+30)=JCODE(N1,NM)
                MCODE(NN,NM+36)=JCODE(O,NM)
                MCODE(NN,NM+42)=JCODE(P,NM)
41              MCODE(NN,NM+48)=JCODE(Q,NM)
      J1=J1+1
40    CONTINUE
      I1=I1+1
42    CONTINUE
      IF(NPLTS.EQ.1) GO TO 65
      I1=1
         DO 52 M=1,NYX,2
         J1=1
         DO 50 N=1,NZX,2
            IN(1)=NOP(M+2,N,2)
            IN(2)=NOP(M,N,2)
            IN(3)=NOP(M,N+2,2)
            IN(4)=NOP(M+2,N+2,2)
            IN(5)=NOP(M+1,N,2)
            IN(6)=NOP(M,N+1,2)
            IN(7)=NOP(M+1,N+2,2)
            IN(8)=NOP(M+2,N+1,2)
            IN(9)=NOP(M+1,N+1,2)
            NN=IOP(I1,J1,2)
            II=1
            DO 51 NM=1,9
               MCODE(NN,II)=JCODE(IN(NM),2)
               MCODE(NN,II+1)=JCODE(IN(NM),3)
               MCODE(NN,II+2)=JCODE(IN(NM),1)
               MCODE(NN,II+3)=JCODE(IN(NM),6)
               MCODE(NN,II+4)=JCODE(IN(NM),4)
               MCODE(NN,II+5)=JCODE(IN(NM),5)
51          II=6*NM+1
```

```
      J1=J1+1
 50   CONTINUE
    I1=I1+1
 52 CONTINUE
    IF(NPLTS.NE.3) GO TO 65
    I1=1
      DO 62 M=1,NXX,2
      J1=1
      DO 60 N=1,NZX,2
         IN(1)=NOP(M+2,N,3)
         IN(2)=NOP(M+2,N+2,3)
         IN(3)=NOP(M,N+2,3)
         IN(4)=NOP(M,N,3)
         IN(5)=NOP(M+2,N+1,3)
         IN(6)=NOP(M+1,N+2,3)
         IN(7)=NOP(M,N+1,3)
         IN(8)=NOP(M+1,N,3)
         IN(9)=NOP(M+1,N+1,3)
        NN=IOP(I1,J1,3)
        II=1
        DO 61 NM=1,9
           MCODE(NN,II)=JCODE(IN(NM).3)
           MCODE(NN,II+1)=JCODE(IN(NM),1)
           MCODE(NN,II+2)=JCODE(IN(NM),2)
           MCODE(NN,II+3)=JCODE(IN(NM),5)
           MCODE(NN,II+4)=JCODE(IN(NM),6)
           MCODE(NN,II+5)=JCODE(IN(NM),4)
 61     II=6*NM+1
      J1=J1+1
 60   CONTINUE
    I1=I1+1
 62 CONTINUE
 65 NELEM=NN
    MAXID=0
    DO 80 I=1,NN
         MAX=0
         MIN=4000
```

```
            DO 90 J=1,54
               M=MCODE(I,J)
               IF(M.EQ.0) GO TO 90
               IF(M.LT.MIN) MIN=M
               IF(M.GT.MAX) MAX=M
   90       CONTINUE
            ID=MAX-MIN
            IF(ID.GT.MAXID) MAXID=ID
   80    CONTINUE
      IHBW=MAXID+1
      WRITE(6,105)
      WRITE(6,99)
      WRITE(6,100)(I,(JCODE(I,J),J=1,6),I=1,NNODES)
      WRITE(6,105)
C     WRITE(6,109)
C     WRITE(6,110)(I,(MCODE(I,J),J=1,18),I=1,NELEM)
C     WRITE(6,105)
C     WRITE(6,110)(I,(MCODE(I,J),J=19,36),I=1,NELEM)
C     WRITE(6,105)
C     WRITE(6,110)(I,(MCODE(I,J),J=37,54),I=1,NELEM)
C     WRITE(6,105)
      WRITE(6,200) NNODES,NELEM,NDOF,IHBW
      WRITE(6,105)
   99 FORMAT(' JOINT',4X,T23,'JOINT CODE'/)
  100 FORMAT(I4,6X,6I5)
  105 FORMAT(/)
C 109 FORMAT(' ELEMENT',2X,T60,'MEMBER CODE'/)
C 110 FORMAT(I5,5X,6I5,5X,6I5,5X,6I5)
  200 FORMAT(' NUMBER OF NODES--------------',I4/
     *' NUMBER OF ELEMENTS-----------',I4/
     *' NUMBER OF DEGREES OF FREEDOM-',I4/
     *' THE HALF BAND WIDTH----------',I4/)
      RETURN
      END
```

```
C***********************************************************************
C                            SUBROUTINE GSFORC                         *
C***********************************************************************
C      GENERATE THE GLOBAL SYSTEM FORCE VECTOR FROM GIVEN LOADINGS IN
C      SUBROUTINE LOAD AND ELEMENT FORCE MATRICES DERIVED IN SUBROUTINE
C      EFORCE.
C
       SUBROUTINE GSFORC(NPLTS)
       IMPLICIT REAL*8 (A-H,O-Z)
       COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
      *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
      *       IHBW,NX,NY,NZ
       COMMON/LD/LTYPE(10),NPL(10),LDIR(10),W(10),H(10),F(6),NLOAD
       COMMON/FO/EF(54)
       COMMON/TC/THKST,THKSB,THKLT,THKLB,THKF
C
       DO 10 I=1,NDOF
   10     GF(I)=0.0
       I=1
       J=1
   15  IF(J.NE.1) I=I+1
       J=J+1
       READ(5,*) LTYPE(I),NPL(I),LDIR(I),W(I),H(I)
       IF(LTYPE(I).NE.0) GO TO 15
       NLOAD=I-1
       WRITE(6,170)
       DO 20 I=1,NLOAD
          I6=LTYPE(I)
          IF(I6.EQ.0) GO TO 23
          GO TO(21,22),I6
   21        WRITE(6,180) NPL(I),LDIR(I),W(I),H(I)
             GO TO 20
   22        WRITE(6,190) NPL(I),LDIR(I),W(I),H(I)
   20  CONTINUE
   23  IF(NPLTS.EQ.3) WRITE(6,210)
```

```
      NN=0
      IF(NPLTS.NE.3) GO TO 24
      TWX=Y*C*WC/1728000.0D0*(THKST+.5D0*(THKSB-THKST))
      THKS=2.5D0*THKF
      WX=(THKSB+THKS)*C
      PX=TWX/WX
      TWZ=Y*B*WC/1728000.0D0*(THKLT+.5D0*(THKLB-THKBT))
      WZ=(THKLB+THKS)*B
      PZ=TWZ/WZ
24    DO 130 I1=1,NPLTS
         GO TO (30,40,50),I1
30          NR=NY
            NC=NX
            N=1
            GO TO 54
40          NR=NY
            NC=NZ
            N=NX+1
            GO TO 54
50          NR=NX
            NC=NZ
            N=(NX+NZ)*NY+1
54       DO 120 I3=1,NR
            T=THK(I3,I1)
            DO 55 NF=1,6
55             F(NF)=0.0D0
            CALL LOAD(NPLTS,I1,T,I3,NN)
            DO 80 J3=1,NC
               F3=F(3)
               IF(I1 .EQ. 3) CALL STRIP(PX,PZ,THKS,NR,NC)
               CALL EFORCE(I1,N)
               DO 60 JM=1,54
                  J1=MCODE(N,JM)
                  IF(J1 .EQ. 0)GO TO 60
                     GF(J1)=GF(J1)+EF(JM)
60                CONTINUE
                  N=N+1
                  F(3)=F3
```

```
  80        CONTINUE
            GO TO (90,100,120),I1
  90           IF(NPLTS .EQ. 1)GO TO 120
               IF(NPLTS .NE. 1)N=N+NZ
               GO TO 120
 100           N=N+NX
 120     CONTINUE
 130  CONTINUE
C     WRITE(6,150)
C     DO 140 I=1,NDOF
C 140    WRITE(6,160) I,GF(I)
      RETURN
C 150 FORMAT(' GLOBAL FORCE VECTOR',//,' NDOF',5X,'FORCE'/)
C 160 FORMAT(I4,3X,D13.6/)
 170  FORMAT(' PLATE',T11,'TYPE LOAD',T25,'DIRECTION',T41,'MAGNITUDE',
     *      T63,'HEIGHT'/)
 180  FORMAT(3X,I1,7X,'TRIANG',8X,I4,9X,D10.4,1X,'PCF',5X,D10.4,1X,
     *      'INCHES')
 190  FORMAT(3X,I1,7X,' UNIF ',8X,I4,9X,D10.4,1X,'PSF',5X,D10.4,1X,
     *      'INCHES')
 210  FORMAT(3X,'1',7X,'WALL DEAD WEIGHT ADDRESSED AS SURFACE TRACTION',
     */,3X,'2',7X,'WALL DEAD WEIGHT ADDRESSED AS SURFACE TRACTION',/,
     *3X,'3',7X,'STRIP LOAD APPROX UTILIZED ON FLOOR SLAB'/)
      END
```

```
C***********************************************************************
C                               SUBROUTINE EFORCE                      *
C***********************************************************************
C     GENERATES ELEMENT FORCE VECTORS THROUGHT THE ASSISTANCE OF SUB-
C     ROUTINE SHAPE BY USING GAUSS NUMERICAL INTEGRATION.
C
      SUBROUTINE EFORCE(I1,N)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION SFX(9)
      COMMON/FO/EF(54)
      COMMON/LD/LTYPE(10),NPL(10),LDIR(10),W(10),H(10),F(6),NLOAD
      COMMON/MSH/X1(250),Y1(250),X3(250),Z3(250),ELXY(9,2),NOD(48,9)
      COMMON/SH/SF(9),GDSF(2,9),GAUSS(3),WT(3),GAUSS1(2),WT1(2)
C
      GAUSS(1)=-.77459667D0
      GAUSS(2)=0.0D0
      GAUSS(3)=.77459667D0
      GAUSS1(1)=-.5773502692D0
      GAUSS1(2)= .5773502692D0
      WT(1)=.55555555D0
      WT(2)=.88888888D0
      WT(3)=.55555555D0
      WT1(1)=1.0D0
      WT1(2)=1.0D0
C
      GO TO (1,3,5),I1
    1   DO 2 I=1,9
          NI=NOD(N,I)
          ELXY(I,1)=X1(NI)
    2     ELXY(I,2)=Y1(NI)
          GO TO 7
    3   DO 4 I=1,9
          NI=NOD(N,I)
          ELXY(I,1)=Y1(NI)
    4     ELXY(I,2)=X1(NI)
          GO TO 7
```

```
 5   DO 6 I=1,9
        NI=NOD(N,I)
        ELXY(I,1)=Z3(NI)
 6      ELXY(I,2)=X3(NI)
C
 7  DO 10 JI=1,9
10     SFX(JI)=0.0D0
    DO 30 NI=1,3
       DO 30 NJ=1,3
          XI=GAUSS(NI)
          ETA=GAUSS(NJ)
          CALL SHAPE(XI,ETA,DET)
          CONST=DET*WT(NI)*WT(NJ)
          DO 20 JI=1,9
20           SFX(JI)=SFX(JI)+CONST*SF(JI)
30  CONTINUE
    DO 40 JI=1,54
40     EF(JI)=0.0D0
C   WRITE(6,70)
    II=1
    DO 60 JI=1,9
      DO 50 KI=1,6
          EF(II)=F(KI)*SFX(JI)
C         WRITE(6,80) II,EF(II)
50        II=II+1
60    II=6*JI+1
    RETURN
C 70 FORMAT(' LOCAL ELEMENT FORCE VECTOR',//,'NODE',5X,'FORCE'/)
C 80 FORMAT(I4,3X,D13.5/)
    END
```

```
C***********************************************************************
C                          SUBROUTINE SHAPE                            *
C***********************************************************************
C     EVALUATES THE INTERPOLATION FUNCTIONS AND THEIR DERIVATIVES WITH
C     REGARD TO NATURAL COORDINATES FOR A NINE-NODE QUADRILATERAL
C     ELEMENT.
C
      SUBROUTINE SHAPE(XI,ETA,DET)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/MSH/X1(250),Y1(250),X3(250),Z3(250),ELXY(9,2),NOD(48,9)
      COMMON/SH/SF(9),GDSF(2,9),GAUSS(3),WT(3),GAUSS1(2),WT1(2)
C
      DIMENSION XNODE(9,2),NP(9),DSF(2,9),GJ(2,2),GJINV(2,2)
C
      DATA XNODE/-1.0D0,2*1.0D0,-1.0D0,0.0D0,1.0D0,0.0D0,-1.0D0,0.0D0,
     *          2*-1.0D0,2*1.0D0,-1.0D0,0.0D0,1.0D0,2*0.0D0/
      DATA NP/1,2,3,4,5,7,6,8,9/
C
C     QUADRATIC INTERPOLATION FUNCTIONS FOR NINE-NODE ELEMENT
C
      FNC(A1,B1)=A1*B1
      DO 100 I2=1,9
         NI=NP(I2)
         XP=XNODE(NI,1)
         YP=XNODE(NI,2)
         XI0=1.0D0+XI*XP
         ETA0=1.0D0+ETA*YP
         XI1=1.0D0-XI*XI
         ETA1=1.0D0-ETA*ETA
         XI2=XP*XI
         ETA2=YP*ETA
         IF(I2 .GT. 4)GO TO 10
            SF(NI)=0.25*FNC(XI0,ETA0)*XI2*ETA2
            DSF(1,NI)=0.25*XP*FNC(ETA2,ETA0)*(1.0+2.0*XI2)
            DSF(2,NI)=0.25*YP*FNC(XI2,XI0)*(1.0+2.0*ETA2)
            GO TO 100
```

```
 10     IF(I2 .GT. 6)GO TO 20
           SF(NI)=0.5*FNC(XI1,ETA0)*ETA2
           DSF(1,NI)=-XI*FNC(ETA2,ETA0)
           DSF(2,NI)=0.5*FNC(XI1,YP)*(1.0+2.0*ETA2)
           GO TO 100
 20     IF(I2 .GT. 8)GO TO 30
           SF(NI)=0.5*FNC(ETA1,XI0)*XI2
           DSF(1,NI)=0.5*FNC(ETA1,XP)*(1.0+2.0*XI2)
           DSF(2,NI)=-ETA*FNC(XI2,XI0)
           GO TO 100
 30     SF(NI)=FNC(XI1,ETA1)
        DSF(1,NI)=-2.0*XI*ETA1
        DSF(2,NI)=-2.0*ETA*XI1
100  CONTINUE
C
     DO 110 I2=1,2
        DO 110 J2=1,2
           GJ(I2,J2)=0.0D0
           DO 110 K2=1,9
110           GJ(I2,J2)=GJ(I2,J2)+DSF(I2,K2)*ELXY(K2,J2)
     DET=GJ(1,1)*GJ(2,2)-GJ(1,2)*GJ(2,1)
C
     GJINV(1,1)=GJ(2,2)/DET
     GJINV(2,2)=GJ(1,1)/DET
     GJINV(1,2)=-GJ(1,2)/DET
     GJINV(2,1)=-GJ(2,1)/DET
C
     DO 120 I2=1,2
        DO 120 J2=1,9
           GDSF(I2,J2)=0.0D0
           DO 120 K2=1,2
120           GDSF(I2,J2)=GDSF(I2,J2)+GJINV(I2,K2)*DSF(K2,J2)
     RETURN
     END
```

```
C***********************************************************************
C                                   SUBROUTINE LOAD                    *
C***********************************************************************
C     COORDINATES THE GENERATION OF FORCES IMPOSED UPON ELEMENTS FROM
C     DEAD WEIGHT, TRIANGULAR, UNIFORM, AND STRIP LOADINGS REPRESENTED
C     BY THEIR RESPECTIVE SUBROUTINE.
C
      SUBROUTINE LOAD(NPLTS,I1,T,I3,NN)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/LD/LTYPE(10),NPL(10),LDIR(10),W(10),H(10),F(6),NLOAD
C
C     IF(NPLTS.NE.1.AND.I1.NE.3) CALL DEADWT(I1,T)
C
      DO 30 NJ=1,NLOAD
         IF(NPL(NJ) .NE. I1) GO TO 30
            L1=LTYPE(NJ)
   99       GOTO(10,20,30),L1
   10          IF(NPL(NJ).EQ.3)GO TO 15
               CALL TRIANG(I3,NJ,NN)
               GO TO 30
   15          WRITE(6,100)
               GO TO 30
   20          CALL UNIF(I3,NJ)
   30 CONTINUE
      RETURN
  100 FORMAT(' THIS PROGRAM DOES NOT ALLOW FOR TRIANGULAR BASE LOADS'/)
      END
```

```
C***********************************************************************
C                             SUBROUTINE TRIANG                        *
C***********************************************************************
C     CALCULATES THE AVERAGE FORCE PRODUCED ON A VERTICAL PLATE ELEMENT
C     FROM A TRIANGULAR LOAD.
C
      SUBROUTINE TRIANG(I3,NJ,NN)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/LD/LTYPE(10),NPL(10),LDIR(10),W(10),H(10),F(6),NLOAD
C
      P2=0.0D0
      D=Y-H(NJ)
      W1=W(NJ)/1728000.0D0
      DO 10 KI=1,I3
         DY=DFLOAT(KI)*A
         IF(DY.LT.D) GO TO 10
            P1=P2
            P2=(DY-D)*W1
            NN=NN+1
            DH=A
            IF(NN.EQ.1) DH=DY-D
            P=(P1+P2)/2.0D0
            IF(LDIR(NJ) .LT. 0) P=-P
            FT=P*DH/A
   10 CONTINUE
C     WRITE(6,20) FT
C  20 FORMAT(' MAGNITUDE OF TRIANG=',D13.5/)
      F(3)=F(3)+FT
      RETURN
      END
```

```
C*********************************************************************
C                              SUBROUTINE UNIF                       *
C*********************************************************************
C     GENERATES THE FORCE PRODUCED ON AN ELEMENT FROM A UNIFORM LOAD.
C
      SUBROUTINE UNIF(I3,NJ)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/LD/LTYPE(10),NPL(10),LDIR(10),W(10),H(10),F(6),NLOAD
C
      IF(NPL(NJ) .EQ. 3 .AND. H(NJ) .NE. X) GO TO 90
         L2=NPL(NJ)
         GOTO(10,20,30),L2
   10       F1=A
            DL=C
            V=Y
            GO TO 40
   20       F1=A
            DL=B
            V=Y
            GO TO 40
   30       F1=C
            DL=B
            V=X
   40    P=W(NJ)/144000.0D0
         D=V-H(NJ)
         IF(LDIR(NJ) .LT. 0) P=-P
         DY=DFLOAT(I3)*F1
         IF(DY.LT.D)GO TO 50
         DY1=(DFLOAT(I3)-1.0D0)*F1
         DH=F1
         IF(DY1.LT.D)DH=DY-D
         P5=P*DH/F1
C         WRITE(6,48) P5
C  48     FORMAT(' MAGNITUDE OF UNIF=',D13.5/)
```

```
      F(3)=F(3)+P5
 50 RETURN
 90 WRITE(6,102)
102 FORMAT(' THE UNIFORM LOAD ON THE FLOOR OF THE TANK MUST BE THE FUL
   *L WIDTH OF THE TANK.'/T5,'H MUST EQUAL X AND W MUST BE ADJUSTED SO
   * THAT THE MULTIPLICATION OF W AND H'/T5,'PROVIDE THE APPROPRIATE P
   *RESSURE ACROSS THE BOTTOM OF THE TANK'/)
    STOP
    END
```

```
C***********************************************************************
C                                     SUBROUTINE DEADWT                *
C***********************************************************************
C     GENERATES THE GRAVITATIONAL FORCES IMPOSED ALONG VERTICAL PLATE
C     ELEMENTS DUE TO THEIR WEIGHT.
C
      SUBROUTINE DEADWT(I1,T)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/LD/LTYPE(10),NPL(10),LDIR(10),W(10),H(10),F(6),NLOAD
C
      W1=WC/1728000.0D0
      GO TO (10,20),I1
   10    DL=C
         GO TO 30
   20    DL=B
   30 W2=W1*T
      WRITE(6,31) W2
   31 FORMAT(' MAGNITUDE OF DEADWT=',D13.5/)
      IF(I1. EQ. 1) F(2)=F(2)-W2
      IF(I1. EQ. 2) F(1)=F(1)-W2
      RETURN
      END
```

```
C*****************************************************************************
C                                 SUBROUTINE STRIP                           *
C*****************************************************************************
C     CALCULATES THE FORCES IMPOSED UPON THE BASE SLAB DUE WALL WEIGHT
C     BEING DITRIBUTED THROUGH THE BASE THICKNESS AT A 2.5 RATIO.
C
      SUBROUTINE STRIP(PX,PZ,THKS,NR,NC)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/LD/LTYPE(10),NPL(10),LDIR(10),W(10),H(10),F(6),NLOAD
C
      DZ=NC*B
      DZ1=(NC-1)*B
      IF(DZ.GE.THKS .AND. DZ1.LT.THKS) F(3)=F(3)+PX*(THKS-DZ1)/B
      IF(DZ.LE.THKS) F(3)=F(3)+PX
C     WRITE(6,10) PX
C  10 FORMAT(' MAGNITUDE OF PX STRIP LOAD= ',D13.5/)
C
      DX=NR*C
      DX1=(NR-1)*C
      THKS1=X-THKS
      IF(DX.GT.THKS1 .AND. DX1.LT.THKS1) F(3)=F(3)+PZ*(DX-THKS1)/C
      IF(DX.GT.THKS1 .AND. DX1.GE.THKS1) F(3)=F(3)+PZ
C     WRITE(6,20) PZ
C  20 FORMAT(' MAGNITUDE OF PZ STRIP LOAD=',D13.5/)
      RETURN
      END
```

```
C***********************************************************************
C                               SUBROUTINE GSSTIF                      *
C***********************************************************************
C     GENERATE THE GLOBAL SYSTEM STIFFNESS MATRIX FROM ALL LOCAL
C     ELEMENT STIFFNESS MATRICES DERIVED IN SUBROUTINE ESTIF.
C
      SUBROUTINE GSSTIF(NPLTS)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/SSM/SST(500,1000)
      COMMON/COEFF/SOIL
      COMMON/ST/ESTIFF(54,54)
C
      DO 81 I=1,IHBW
      DO 81 J=1,NDOF
   81    SST(I,J)=0.0D0
C
      DO 130 I1=1,NPLTS
         GO TO (10,20,30),I1
   10       NR=NY
            NC=NX
            N=1
            GO TO 40
   20       NR=NY
            NC=NZ
            N=NX+1
            GO TO 40
   30       NR=NX
            NC=NZ
            N=(NX+NZ)*NY+1
   40    DO 120 I3=1,NR
            T=THK(I3,I1)
            DO 80 J3=1,NC
               CALL ESTIF(I1,T,N)
```

```
      DO 60 JM=1,54
         J1=MCODE(N,JM)
         IF(J1.EQ.0) GO TO 60
            DO 50 KM=1,54
               K=MCODE(N,KM)
               IF(K.EQ.0) GO TO 50
                  KB=J1-K+IHBW
                  SST(KB,K)=SST(KB,K)+ESTIFF(JM.KM)
50             CONTINUE
60       CONTINUE
         N=N+1
80    CONTINUE
      GO TO (90,100,120),I1
90       IF(NPLTS .EQ. 1)GO TO 120
         IF(NPLTS .NE. 1)N=N+NZ
         GO TO 120
100      N=N+NX
120   CONTINUE
130 CONTINUE
    IF(NPLTS.EQ.3.AND.SOIL.GT.0.) CALL MODIFY
    RETURN
    END
```

```
C***********************************************************************
C                              SUBROUTINE MODIFY                       *
C***********************************************************************
C     MODIFIES THE HORIZONTAL PLATE STIFFNESS FOR A 3 PLATE SYSTEM TO
C     ACCOMMODATE A WINKLER FOUNDATION.
C
      SUBROUTINE MODIFY
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/CODE/JCODE(220,6)
      COMMON/SSM/SST(500,1000)
      COMMON/COEFF/SOIL
C
      NZ1=2*NZ+1
      NX1=2*NX+1
      DO 10 I=1,NX1
      DO 10 J=1,NZ1
         K=NOP(I,J,3)
         L=JCODE(K,2)
         IF(L.EQ.0) GO TO 10
            COEF=SOIL/4.0D0
            IF(I.EQ.1.OR.I.EQ.NX1) COEF=COEF/2.0D0
            IF(J.EQ.1.OR.J.EQ.NZ1) COEF=COEF/2.0D0
            SST(IHBW,L)=SST(IHBW,L)+COEF
   10 CONTINUE
      RETURN
      END
```

```
C**********************************************************************
C                              SUBROUTINE ESTIF                       *
C**********************************************************************
C     GENERATES ELEMENT STIFFNESS MATRICES THROUGH THE ASSISTANCE OF
C     SUBROUTINE SHAPE BY USING GAUSS POINTS AND NUMERICAL INTEGRATION.
C     (FULL INTEGRATION IS USED FOR IN-PLANE & BENDING TERMS WHILE RE-
C     DUCED INTEGRATION IS USED ON SHEAR TERMS.)
C
      SUBROUTINE ESTIF(I1,T,N)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/ST/ESTIFF(54,54)
      COMMON/SH/SF(9),GDSF(2,9),GAUSS(3),WT(3),GAUSS1(2),WT1(2)
      COMMON/MSH/X1(250),Y1(250),X3(250),Z3(250),ELXY(9,2),NOD(48,9)
      COMMON/CON/C1(3,3),D(5,5)
C
      DIMENSION SXY(9,9),SX1Y(9,9),SXY1(9,9),SX1X1(9,9),
     *          SX1Y1(9,9),SY1Y1(9,9)
C
C     COMPUTE ELEMENT MODULI OF ELASTICITY
C
      V21=VNU*E2/E1
      DENOM=1.0D0-VNU*V21
      C1(1,1)=E1*T/DENOM
      C1(1,2)=V21*C1(1,1)
      C1(2,2)=E2*C1(1,1)/E1
      C1(3,3)=T*G12
```

```
C
C     COMPUTE ELEMENT PLATE STIFFNESSES
C
      SC=5.0D0/6.0D0
      TC=T**2/12.0D0
      D(1,1)=TC*C1(1,1)
      D(1,2)=TC*C1(1,2)
      D(2,2)=TC*C1(2,2)
      D(3,3)=TC*C1(3,3)
      D(4,4)=T*G13*SC
      D(5,5)=T*G23*SC
C     WRITE(6,8) T,C1(1,1),C1(1,2),C1(2,2),C1(3,3)
C     WRITE(6,9) D(1,1),D(1,2),D(2,2),D(3,3),D(4,4),D(5,5)
C   8 FORMAT(5D13.5/)
C   9 FORMAT(6D13.5/)
C
      GO TO (1,3,5),I1
    1    DO 2 I=1,9
            NI=NOD(N,I)
            ELXY(I,1)=X1(NI)
    2       ELXY(I,2)=Y1(NI)
         GO TO 7
    3    DO 4 I=1,9
            NI=NOD(N,I)
            ELXY(I,1)=Y1(NI)
    4       ELXY(I,2)=X1(NI)
         GO TO 7
    5    DO 6 I=1,9
            NI=NOD(N,I)
            ELXY(I,1)=Z3(NI)
    6       ELXY(I,2)=X3(NI)
C
C     INITIALIZE THE ARRAYS
C
    7 DO 20 I=1,9
         DO 20 J=1,9
```

```
                SX1X1(I,J)=0.0D0
                SX1Y1(I,J)=0.0D0
   20           SY1Y1(I,J)=0.0D0
C
C     DO-LOOPS ON NUMERICAL (GAUSS) QUADRATURE BEGIN HERE
C
      DO 100 NI=1,3
         DO 100 NJ=1,3
            XI=GAUSS(NI)
            ETA=GAUSS(NJ)
            CALL SHAPE(XI,ETA,DET)
            CONST=DET*WT(NI)*WT(NJ)
            DO 80 I=1,9
               DO 80 J=1,9
                  SX1X1(I,J)=SX1X1(I,J)+CONST*GDSF(1,I)*GDSF(1,J)
                  SX1Y1(I,J)=SX1Y1(I,J)+CONST*GDSF(1,I)*GDSF(2,J)
   80             SY1Y1(I,J)=SY1Y1(I,J)+CONST*GDSF(2,I)*GDSF(2,J)
  100 CONTINUE
      DO 110 I=1,54
         DO 110 J=1,54
  110       ESTIFF(I,J)=0.0D0
      II=1
      DO 130 I=1,9
         JJ=1
         DO 120 J=1,9
C
C     IN-PLANE STIFFNESSES
C
            ESTIFF(II,JJ)=C1(1,1)*SX1X1(I,J)+C1(3,3)*SY1Y1(I,J)
            ESTIFF(II+1,JJ+1)=C1(3,3)*SX1X1(I,J)+C1(2,2)*SY1Y1(I,J)
            ESTIFF(II,JJ+1)=C1(1,2)*SX1Y1(I,J)+C1(3,3)*SX1Y1(J,I)
            ESTIFF(II+1,JJ)=C1(1,2)*SX1Y1(J,I)+C1(3,3)*SX1Y1(I,J)
```

```
C
C     PLATE BENDING STIFFNESSES
C
            ESTIFF(II+3,JJ+3)=D(1,1)*SX1X1(I,J)+D(3,3)*SY1Y1(I,J)
            ESTIFF(II+3,JJ+4)=D(1,2)*SX1Y1(I,J)+D(3,3)*SX1Y1(J,I)
            ESTIFF(II+4,JJ+3)=D(1,2)*SX1Y1(J,I)+D(3,3)*SX1Y1(I,J)
            ESTIFF(II+4,JJ+4)=D(3,3)*SX1X1(I,J)+D(2,2)*SY1Y1(I,J)
  120     JJ=6*J+1
  130 II=6*I+1
      DO 131 I=1,9
         DO 131 J=1,9
            SXY(I,J)=0.0D0
            SX1Y(I,J)=0.0D0
            SXY1(I,J)=0.0D0
            SX1X1(I,J)=0.0D0
  131       SY1Y1(I,J)=0.0D0
      DO 150 NI=1,2
         DO 150 NJ=1,2
            XI=GAUSS1(NI)
            ETA=GAUSS1(NJ)
            CALL SHAPE(XI,ETA,DET)
            CONST=DET*WT1(NI)*WT1(NJ)
            DO 140 I=1,9
               DO 140 J=1,9
                  SXY(I,J)=SXY(I,J)+CONST*SF(I)*SF(J)
                  SX1Y(I,J)=SX1Y(I,J)+CONST*GDSF(1,I)*SF(J)
                  SXY1(I,J)=SXY1(I,J)+CONST*SF(I)*GDSF(2,J)
                  SX1X1(I,J)=SX1X1(I,J)+CONST*GDSF(1,I)*GDSF(1,J)
  140             SY1Y1(I,J)=SY1Y1(I,J)+CONST*GDSF(2,I)*GDSF(2,J)
  150 CONTINUE
      II=1
      DO 170 I=1,9
         JJ=1
         DO 160 J=1,9
```

```
C
C     PLATE SHEAR STIFFNESSES
C
            ESTIFF(II+2,JJ+2)=D(4,4)*SX1X1(I,J)+D(5,5)*SY1Y1(I,J)
            ESTIFF(II+3,JJ+2)=D(4,4)*SX1Y(J,I)
            ESTIFF(II+2,JJ+4)=D(5,5)*SXY1(J,I)
            ESTIFF(II+2,JJ+3)=D(4,4)*SX1Y(I,J)
            ESTIFF(II+3,JJ+3)=ESTIFF(II+3,JJ+3)+D(4,4)*SXY(I,J)
            ESTIFF(II+4,JJ+2)=D(5,5)*SXY1(I,J)
            ESTIFF(II+4,JJ+4)=ESTIFF(II+4,JJ+4)+D(5,5)*SXY(I,J)
            ESTIFF(II+5,JJ+5)=9.0D09
  160    JJ=6*J+1
  170 II=6*I+1
C     WRITE(6,180)
C     WRITE(6,190) N
C     WRITE(6,200) ((ESTIFF(I,K),K=1,9),I=1,54)
C     WRITE(6,180)
C     WRITE(6,200) ((ESTIFF(I,K),K=10,18),I=1,54)
C     WRITE(6,180)
C     WRITE(6,200) ((ESTIFF(I,K),K=19,27),I=1,54)
C     WRITE(6,180)
C     WRITE(6,200) ((ESTIFF(I,K),K=28,36),I=1,54)
C     WRITE(6,180)
C     WRITE(6,200) ((ESTIFF(I,K),K=37,45),I=1,54)
C     WRITE(6,180)
C     WRITE(6,200) ((ESTIFF(I,K),K=46,54),I=1,54)
      RETURN
C 180 FORMAT(/)
C 190 FORMAT(' ELEMENT',2X,I4,2X,'LOCAL STIFFNESS MATRIX'/)
C 200 FORMAT(5X,9D13.5/)
      END
```

```fortran
C***********************************************************************
C                              SUBROUTINE SOLVE                        *
C***********************************************************************
C     SOLVES THE "K*Q=GF" EQUATION FOR Q (DISPLACEMENTS) BY A LINPACK
C     EQUATION SOLVER.  THE PROCESS USES BACKWARD AND THEN FORWARD
C     SUBSTITUTION.("Q" WILL REPLACE THE "GF" VALUES.)
C
      SUBROUTINE SOLVE
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/SSM/SST(500,1000)
      COMMON/SOLV/MAXID,LDA
C
C     REDUCE STIFFNESS MATRIX USING THE LINPACK EQUATION SOLVER
C
      CALL SPBFA(INFO)
      IF(INFO.EQ.0) GO TO 90
         WRITE(6,100) INFO
  100    FORMAT(///' *** STOP ***  THE LEADING MINOR OF ORDER',I5,2X,
     *          'IS NOT POSITIVE DEFINITE'/)
         STOP
C
C     REDUCE FORCE VECTOR AND BACK SOLVE FOR DISPLACEMENTS
C
   90 CALL SPBSL
C     WRITE(6,11)
C     WRITE(6,10) (I,GF(I),I=1,NDOF)
C  10 FORMAT(6(' Q(',I3,')=',D10.4,3X))
C  11 FORMAT(/' GENERALIZED DISPLACEMENTS'/)
      RETURN
      END
```

```
C***********************************************************************
C                                 SUBROUTINE SPBFA                     *
C***********************************************************************
      SUBROUTINE SPBFA (INFO)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/SSM/SST(500,1000)
      COMMON/SOLV/MAXID,LDA
C
      DO 30 J=1,NDOF
         INFO=J
         S=0.0D0
         IK=MAXID +1
         JK=MAX0 (J-MAXID,1)
         MU=MAX0 (MAXID+2-J,1)
         IF(MAXID.LT.MU) GO TO 20
            DO 10 K=MU,MAXID
               T=SST(K,J)-SDOT(K-MU,SST(IK,JK),1,SST(MU,J),1)
               T=T/SST(MAXID+1,JK)
               SST(K,J)=T
               S=S+T*T
               IK=IK-1
               JK=JK+1
   10       CONTINUE
   20    CONTINUE
         S=SST(MAXID+1,J)-S
         IF(S.LE.0.0D0) GO TO 40
         SST(MAXID+1,J)=SQRT(S)
   30 CONTINUE
      INFO=0
   40 CONTINUE
      RETURN
      END
```

```
C***********************************************************************
C                                     SUBROUTINE SPBSL                  *
C***********************************************************************
      SUBROUTINE SPBSL
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/SSM/SST(500,1000)
      COMMON/SOLV/MAXID,LDA
C
C     FORWARD REDUCTION OF CONSTANTS
C
      DO 10 K=1,NDOF
         LM=MINO (K-1,MAXID)
         LA=MAXID+1-LM
         LB=K-LM
         T=SDOT(LM,SST(LA,K),1,GF(LB),1)
         GF(K)=(GF(K)-T)/SST(MAXID+1,K)
   10 CONTINUE
C
C     BACKSUBSTITUTION
C
      DO 20 KB=1,NDOF
         K=NDOF +1-KB
         LM=MINO(K-1,MAXID)
         LA=MAXID+1-LM
         LB=K-LM
         GF(K)=GF(K)/SST(MAXID+1,K)
         T=-GF(K)
         CALL SAXPY(LM,T,SST(LA,K),1,GF(LB),1)
   20 CONTINUE
      RETURN
      END
```

```
C***********************************************************************
C                                  FUNCTION SDOT                       *
C***********************************************************************
      REAL FUNCTION SDOT*8(N,SX,INCX,SY,INCY)
      REAL*8 SX(*),SY(*),STEMP
C
      STEMP=0.0D0
      SDOT=0.0D0
      IF(N.LE.0) GO TO 70
        IF(INCX.EQ.1.AND.INCY.EQ.1) GO TO 20
          IX=1
          IY=1
          IF(INCX.LT.0) IX=(-N+1)*INCX+1
          IF(INCY.LT.0) IY=(-N+1)*INCY+1
          DO 10 I=1,N
            STEMP=STEMP+SX(IX)*SY(IY)
            IX=IX+INCX
            IY=IY+INCY
   10     CONTINUE
          SDOT=STEMP
          GO TO 70
   20   M=MOD(N,5)
        IF(M.EQ.0) GO TO 40
          DO 30 I=1,M
            STEMP=STEMP+SX(I)*SY(I)
   30     CONTINUE
          IF(N.LT.5) GO TO 60
   40       MP1=M+1
            DO 50 I=MP1,N,5
              STEMP=STEMP+SX(I)*SY(I)+SX(I+1)*SY(I+1)+SX(I+2)*
     *                  SY(I+2)+SX(I+3)*SY(I+3)+SX(I+4)*SY(I+4)
   50       CONTINUE
   60     SDOT=STEMP
   70 CONTINUE
      RETURN
      END
```

```
C***********************************************************************
C                                SUBROUTINE SAXPY                      *
C***********************************************************************
      SUBROUTINE SAXPY(N,SA,SX,INCX,SY,INCY)
      REAL*8 SX(*),SY(*),SA
C
      IF(N.LE.0) RETURN
      IF(SA.EQ.0.0D0) RETURN
      IF(INCX.EQ.1.AND.INCY.EQ.1) GO TO 20
         IX=1
         IY=1
         IF(INCX.LT.0)IX=(-N+1)*INCX+1
         IF(INCY.LT.0)IY=(-N+1)*INCY+1
         DO 10 I=1,N
            SY(IY)=SY(IY)+SA*SX(IX)
            IX=IX+INCX
            IY=IY+INCY
   10    CONTINUE
   20 M=MOD(N,4)
      IF(M.EQ.0) GO TO 40
         DO 30 I=1,M
            SY(I)=SY(I)+SA*SX(I)
   30    CONTINUE
         IF(N.LT.4) RETURN
   40 MP1=M+1
      DO 50 I=MP1,N,4
         SY(I)=SY(I)+SA*SX(I)
         SY(I+1)=SY(I+1) + SA*SX(I+1)
         SY(I+2)=SY(I+2) + SA*SX(I+2)
         SY(I+3)=SY(I+3) + SA*SX(I+3)
   50 CONTINUE
      RETURN
      END
```

```
C***********************************************************************
C                              SUBROUTINE TENFOR                        *
C***********************************************************************
C     PRINTS EACH NODE DISPLACEMENT (U,V,W,SX,SY, & SZ) BY ELEMENT.
C     CALCULATES AND PRINTS THE NORMAL IN-PLANE (NX & NY), IN-PLANE
C     SHEARING (NXY), BENDING SHEAR (QX & QY), BENDING MOMENT (MX &
C     MY), AND TWISTING MOMENT (MXY) FORCE FOR EACH ELEMENT NODE
C     BY ELEMENT AT THE GAUSS POINTS USING REDUCED INTEGRATION.
C
      SUBROUTINE TENFOR(NPLTS)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON GF(1000),THK(8,3),A,B,C,E1,E2,G12,G13,G23,VNU,WC,X,Y,Z,
     *       MCODE(48,54),NOP(9,9,3),IOP(4,4,3),NELEM,NNODES,NDOF,
     *       IHBW,NX,NY,NZ
      COMMON/MSH/X1(250),Y1(250),X3(250),Z3(250),ELXY(9,2),NOD(48,9)
      COMMON/SH/SF(9),GDSF(2,9),GAUSS(3),WT(3),GAUSS1(2),WT1(2)
      COMMON/SSM/SST(500,1000)
      COMMON/CON/C1(3,3),D(5,5)
      DIMENSION SS(32)
C
      DO 15 I=1,NELEM
         DO 10 J=1,54
            ND=MCODE(I,J)
            IF(ND .NE. 0) SST(J,I)=GF(ND)
            IF(ND .EQ. 0) SST(J,I)=0.0D0
   10    CONTINUE
   15 CONTINUE
C
      DO 150 I1=1,NPLTS
         WRITE(6,160)
         WRITE(6,170) I1
         GO TO (20,30,40),I1
   20       NR=NY
            NC=NX
            N=1
            GO TO 50
```

```
30        NR=NY
          NC=NZ
          N=NX+1
          GO TO 50
40        NR=NX
          NC=NZ
          N=NY*(NX+NZ)+1
50     DO 140 I=1,NR
          T=THK(I,I1)
          DO 110 J=1,NC
             WRITE(6,180)N
             WRITE(6,190)
             WRITE(6,200)
             GO TO(51,52,53),I1
51              WRITE(6,197)
                GO TO 54
52              WRITE(6,198)
                GO TO 54
53              WRITE(6,199)
54           WRITE(6,201) (SST(II,N),II=1,6)
             WRITE(6,202) (SST(II,N),II=7,12)
             WRITE(6,203) (SST(II,N),II=13,18)
             WRITE(6,204) (SST(II,N),II=19,24)
             WRITE(6,205) (SST(II,N),II=25,30)
             WRITE(6,206) (SST(II,N),II=31,36)
             WRITE(6,207) (SST(II,N),II=37,42)
             WRITE(6,208) (SST(II,N),II=43,48)
             WRITE(6,209) (SST(II,N),II=49,54)
             DO 80 NK=1,9
                NN=NOD(N,NK)
                GO TO(60,65,70),I1
60                 ELXY(NK,1)=X1(NN)
                   ELXY(NK,2)=Y1(NN)
                   GO TO 80
65                 ELXY(NK,1)=Y1(NN)
                   ELXY(NK,2)=X1(NN)
                   GO TO 80
```

```
70          ELXY(NK,1)=Z3(NN)
            ELXY(NK,2)=X3(NN)
80    CONTINUE
      NQ=1
      WRITE(6,280)
      GO TO(81,82,83),I1
81       WRITE(6,290)
         GO TO 84
82       WRITE(6,300)
         GO TO 84
83       WRITE(6,310)
84    DO 100 NI=1,2
         DO 95 NJ=1,2
            XI=GAUSS1(NI)
            ETA=GAUSS1(NJ)
            CALL SHAPE(XI,ETA,DET)
            UX=0.0D0
            UY=0.0D0
            VX=0.0D0
            VY=0.0D0
            SXI=0.0D0
            SYI=0.0D0
            DXW=0.0D0
            DYW=0.0D0
            DXX=0.0D0
            DXY=0.0D0
            DYX=0.0D0
            DYY=0.0D0
            DO 90 NK=1,9
               L=(NK-1)*6+1
               UX=UX+GDSF(1,NK)*SST(L,N)
               UY=UY+GDSF(2,NK)*SST(L,N)
               VX=VX+GDSF(1,NK)*SST(L+1,N)
               VY=VY+GDSF(2,NK)*SST(L+1,N)
               DXW=DXW+GDSF(1,NK)*SST(L+2,N)
               DYW=DYW+GDSF(2,NK)*SST(L+2,N)
               SXI=SXI+SF(NK)*SST(L+3,N)
               SYI=SYI+SF(NK)*SST(L+4,N)
```

```
                    DXX=DXX+GDSF(1,NK)*SST(L+3,N)
                    DYX=DYX+GDSF(2,NK)*SST(L+3,N)
                    DXY=DXY+GDSF(1,NK)*SST(L+4,N)
 90                 DYY=DYY+GDSF(2,NK)*SST(L+4,N)
                  SS(NQ)=(C1(1,1)*UX+C1(1,2)*VY)*T
                  SS(NQ+1)=(C1(1,2)*UX+C1(2,2)*VY)*T
                  SS(NQ+2)=(C1(3,3)*(UY+VX))*T
                  SS(NQ+3)=(D(1,1)*DXX+D(1,2)*DYY)
                  SS(NQ+4)=(D(1,2)*DXX+D(2,2)*DYY)
                  SS(NQ+5)=D(3,3)*(DXY+DYX)
                  SS(NQ+6)=D(4,4)*(DXW+SXI)
                  SS(NQ+7)=D(5,5)*(DYW+SYI)
                  NQ=NQ+8
 95             CONTINUE
100           CONTINUE
              WRITE(6,291) (SS(II),II=1,8)
              WRITE(6,292) (SS(II),II=17,24)
              WRITE(6,293) (SS(II),II=25,32)
              WRITE(6,294) (SS(II),II=9,16)
              N=N+1
110         CONTINUE
            GO TO(120,130,140),I1
120           IF(NPLTS .EQ. 1)GO TO 140
              IF(NPLTS .NE. 1)N=N+NZ
              GO TO 140
130           N=N+NX
140       CONTINUE
150     CONTINUE
        RETURN
160     FORMAT(/)
170     FORMAT(' PLATE',2X,I2/)
180     FORMAT(2X,'ELEMENT',2X,I4/)
190     FORMAT(3X,'NODAL DISPLACEMENTS (INCHES)'/)
197     FORMAT(4X,'GLOBAL',7X,'X',11X,'Y',11X,'Z',10X,'SX',10X,'SY',10X,
       *'SZ'/)
198     FORMAT(4X,'GLOBAL',7X,'Z',11X,'Y',11X,'X',10X,'SZ',10X,'SY',10X,
       *'SX'/)
```

```
199 FORMAT(4X,'GLOBAL',7X,'Z',11X,'X',11X,'Y',10X,'SZ',10X,'SX',10X,
   *'SY'/)
200 FORMAT(4X,'LOCAL',8X,'U',11X,'V',11X,'W',10X,'SU',10X,'SV',10X,
   *'SW'/)
201 FORMAT(' NODE 1',3X,6D12.4/)
202 FORMAT(' NODE 2',3X,6D12.4/)
203 FORMAT(' NODE 3',3X,6D12.4/)
204 FORMAT(' NODE 4',3X,6D12.4/)
205 FORMAT(' NODE 5',3X,6D12.4/)
206 FORMAT(' NODE 6',3X,6D12.4/)
207 FORMAT(' NODE 7',3X,6D12.4/)
208 FORMAT(' NODE 8',3X,6D12.4/)
209 FORMAT(' NODE 9',3X,6D12.4/)
280 FORMAT(3X,'RESULTANT FORCES (KIPS/INCH)'/)
290 FORMAT(16X,'NX',10X,'NY',10X,'NXY',9X,'MX',10X,'MY',10X,
   *        'MXY',9X,'QX',10X,'QY'/)
291 FORMAT(' NODE 1',3X,8D12.4/)
292 FORMAT(' NODE 2',3X,8D12.4/)
293 FORMAT(' NODE 3',3X,8D12.4/)
294 FORMAT(' NODE 4',3X,8D12.4/)
300 FORMAT(16X,'NY',10X,'NZ',10X,'NYZ',9X,'MY',10X,'MZ',10X,
   *        'MYZ',9X,'QY',10X,'QZ'/)
310 FORMAT(16X,'NZ',10X,'NX',10X,'NZX',9X,'MZ',10X,'MX',10X,
   *        'MZX',9X,'QZ',10X,'QX'/)
    END
```

The vita has been removed from
the scanned document

SHEAR ANALYSIS OF RECTANGULAR CONCRETE TANKS

CONSIDERING

INTERACTION OF PLATE ELEMENTS

by

Patrick Edward Devens

Committee Chairman: Dr. J. Herbert Moore

Civil Engineering

(ABSTRACT)

The program described in this paper is capable of determining horizontal and vertical shear resultants at any point within a one, two, or three plate system where plates are joined at ninety degree angles. Program options allow for tapered walls and combinations of both uniform and triangular loads for each type plate problem. The three plate solution process automatically imposes boundary conditions conducive to one quarter of a monolithically constructed rectangular tank system due to tank symmetry. The plate dead weight effects and their associated shear force upon the floor plate are also accounted for when analyzing the three plate problem. Difficulties are encountered in the refinement of plate finite element meshes due to program computer storage requirements. The result is that derived resultant values do correlate with published solutions for one or two plate problems. Three plate problems require at least an 8x8 plate mesh, 8.7 megabits computer storage, to provide acceptable resultant accuracy.