Lawrence Berkeley National Laboratory

Recent Work

Title
SLOSHING OF WATER IN TORUS PRESSURE-SUPPRESSION POOL OF BOILING WATER REACTORS UNDER EARTHQUAKE GROUND MOTIONS

Permalink
https://escholarship.org/uc/item/59v5j56g

Author
Aslam, M.

Publication Date
1978-08-01
SLOSHING OF WATER IN TORUS
PRESSURE-SUPPRESSION POOL OF BOILING WATER REACTORS
UNDER EARTHQUAKE GROUND MOTIONS

M. Aslam, W. G. Godden, and D. T. Scalise

August 1978

Prepared for the U. S. Department of Energy
under Contract W-7405-ENG-48
DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.
SLOSHING OF WATER IN TORUS
PRESSURE-SUPPRESSION POOL OF BOILING WATER REACTORS
UNDER EARTHQUAKE GROUND MOTIONS

A Report of an Analytical and Experimental Study
of Sloshing Response in Axisymmetric Tanks
Under Earthquake Ground Motions

by

M. Aslam
Associate Development Engineer
Department of Civil Engineering, University of California
Berkeley, California 94720

W. G. Godden
Professor of Civil Engineering
University of California
Berkeley, California 94720

and

D. T. Scalise
Engineering Sciences Department Head
Lawrence Berkeley Laboratory, University of California
Berkeley, California 94720

August 1978
CONTENTS

LIST OF SYMBOLS. ......................................................... v

ABSTRACT ........................................................................ vi

1. INTRODUCTION. ......................................................... 1-1
   1.1 Objective. ................................................................. 1-1
   1.2 Scope of the Investigation .................................. 1-2

2. TORUS TANK MODEL TESTS AND COMPARISON WITH APPROXIMATE ANALYSIS ................................. 2-1
   2.1 Introduction ............................................................... 2-1
   2.2 Model Description and Instrumentation ..................... 2-2
   2.3 Test Procedure and Experimental Data ........................ 2-2
      2.3.1 Tests on small shaking table ............................... 2-2
      2.3.2 Tests under simulated earthquakes ..................... 2-3
   2.4 Comparison of Test Results with Approximate Theory .... 2-5
      2.4.1 Comparison of natural sloshing frequencies .......... 2-5
      2.4.2 Comparison of sloshing displacements under steady state harmonic ground motion .. 2-6
      2.4.3 Comparison of sloshing displacements under simulated earthquake ground motions .. 2-6
   2.5 Dynamic Pressures ....................................................... 2-7
   2.6 Linearity Range .......................................................... 2-8
   2.7 Summary of Important Observations on Sloshing in Torus Tank Model ............................................. 2-8

3. FINITE ELEMENT ANALYSIS OF EARTHQUAKE INDUCED SLOSHING IN AXISYMMETRIC TANKS ...................... 3-1
   3.1 Introduction ............................................................... 3-1
   3.2 Mathematical Formulation ....................................... 3-2
      3.2.1 Equations of motion ........................................... 3-2
      3.2.2 Boundary conditions ......................................... 3-3
   3.3 Finite Element Formulation ...................................... 3-4
      3.3.1 Derivation of finite element equations ............... 3-4
      3.3.2 Isoparametric formulation for axisymmetric tank under arbitrary horizontal ground motion ...................................................... 3-7
      3.3.3 Free surface (mass) matrix for axisymmetric element .......................................................... 3-12
      3.3.4 Evaluation of fluid (stiffness) matrix for axisymmetric element ............................................. 3-13
      3.3.5 Load vector for axisymmetric element ................... 3-14
-v-

LIST OF SYMBOLS

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Outer radius of tank</td>
</tr>
<tr>
<td>b</td>
<td>Inner radius of tank</td>
</tr>
<tr>
<td>B1</td>
<td>Liquid-solid interface boundary</td>
</tr>
<tr>
<td>B2</td>
<td>Free surface boundary</td>
</tr>
<tr>
<td>EB1</td>
<td>Element boundary B1</td>
</tr>
<tr>
<td>EB2</td>
<td>Element boundary B2</td>
</tr>
<tr>
<td>EV</td>
<td>Element Volume</td>
</tr>
<tr>
<td>F</td>
<td>Load vector (Solid-liquid interface matrix)</td>
</tr>
<tr>
<td>g</td>
<td>Acceleration of gravity</td>
</tr>
<tr>
<td>h</td>
<td>Water depth</td>
</tr>
<tr>
<td>J</td>
<td>Jacobian matrix</td>
</tr>
<tr>
<td>K</td>
<td>Fluid matrix (equivalent to stiffness matrix)</td>
</tr>
<tr>
<td>M</td>
<td>Free surface matrix (equivalent to mass matrix)</td>
</tr>
<tr>
<td>N</td>
<td>Shape functions or a number</td>
</tr>
<tr>
<td>p</td>
<td>Pressure</td>
</tr>
<tr>
<td>r</td>
<td>Local coordinate</td>
</tr>
<tr>
<td>s</td>
<td>Local coordinate or surface</td>
</tr>
<tr>
<td>t</td>
<td>Time</td>
</tr>
<tr>
<td>v_n</td>
<td>Normal velocity</td>
</tr>
<tr>
<td>X,Y,Z</td>
<td>Rectangular coordinates</td>
</tr>
<tr>
<td>φ</td>
<td>Velocity potential</td>
</tr>
<tr>
<td>\dot{\phi}</td>
<td>d\phi/dt</td>
</tr>
<tr>
<td>Δt</td>
<td>Time step</td>
</tr>
<tr>
<td>ρ</td>
<td>Mass density</td>
</tr>
<tr>
<td>δ</td>
<td>Free surface displacement or a parameter</td>
</tr>
</tbody>
</table>
ABSTRACT

This report presents an analytical and experimental investigation into the sloshing of water in torus tanks under horizontal earthquake ground motions. This study was motivated because of the use of torus tanks for pressure-suppression pools in Boiling Water Reactors. Such a pressure-suppression pool would typically have 80 ft and 140 ft inside and outside diameters, a 30 ft diameter section, and a water depth of 15 ft.

A general finite element analysis was developed for all axisymmetric tanks and a computer program was written to obtain time-history plots of sloshing displacements of water and dynamic pressures. Tests were carried out on a 1/60th scale model under sinusoidal as well as simulated earthquake ground motions. Tests and analytical results regarding natural frequencies, surface water displacements, and dynamic pressures were compared and a good agreement was found within the range of displacements studied. The computer program gave satisfactory results within a maximum range of sloshing displacements in the full-size prototype of 30 in. which is greater than the value obtained under the full intensity of the El Centro earthquake (N-S component 1940). The range of linear behavior was studied experimentally by subjecting the torus model to increasing intensities of the El Centro earthquake. The general computer program was also used for comparison with a previous study on the sloshing of water in annular tanks, and the previous annular tank solution was also used as an approximate solution in the torus tank
problem showing that sloshing response is not very sensitive to the precise cross-section geometry of an axisymmetric tank.

Tests were also conducted to study the effect of vertical ground motions on the dynamic response of the fluid. These showed a negligible effect on sloshing displacements.

KEY WORDS

1. INTRODUCTION

1.1 Objective

This report presents the results of a study into the sloshing response of water in torus tanks under the action of earthquake-induced ground motions, and is a continuation of a previous investigation into the sloshing response of water in annular tanks [1].

Torus tanks are used as pressure-suppression pools in certain designs of boiling water reactors (BWR) (e.g., General Electric Mark I), and a knowledge of the dynamic response of the water and particularly of the resulting water surface elevations is important in evaluating the effectiveness of the system under earthquake conditions. A typical torus suppression pool as used in the GE Mark I reactor has an outside diameter of 140 ft and a section height of 30 ft. A 1/60th scale idealized model of such a pool is shown in Fig. 2-1a.

Rather than deriving a particular analytical solution to the torus tank problem, it was decided at the outset to undertake a general study into the sloshing of water in axisymmetric tanks; this would be applicable to the torus tank, the annular tank previously studied, and to all tanks with a constant section of revolution. References [1] through [16] are some of the previous studies related to the sloshing of fluids in tanks.
1.2 Scope of the Investigation

This study includes the testing of a 1/60th scale model of GE Mark 1 torus, the development of a general finite element theory for axisymmetric tanks, and a computer code to implement the finite element theory. A comparison of test results with an approximate analysis based on the annular tank theory is also given in Chapter 2.

The test model was constructed by cementing together wedge-shaped lengths of 6 in. lucite tubing as shown in Fig. 2-1. This was tested under harmonic and simulated earthquake ground motions. The quantities measured included the sloshing frequencies and free surface displacements.

The finite element equations (Chapter 3) were derived using the Galerkin principle and a linearized small displacement theory was used. The velocity potential $\phi$ was taken as the field variable and the sloshing displacements as well as impulsive pressures were derived from $\phi$. The finite element equations were first derived for a general three dimensional sloshing problem under arbitrary ground motions, and then it was specialized to an axisymmetric tank subjected to horizontal earthquake ground motions only.

A computer code named 'SLOSH2' was developed to implement the finite element theory, and a comparison of sloshing displacements and pressures as predicted by the computer program was made with the test results from annular [1] and torus tanks. These comparisons show that the finite element program can successfully predict the sloshing displacements as well as impulsive pressures in an axisymmetric tank under horizontal ground motions within the range of linear behavior.
2. TORUS TANK MODEL TESTS AND COMPARISON WITH APPROXIMATE ANALYSIS

2.1 Introduction

The objective of the tests on the torus tank model was to obtain experimental data to compare with the results obtained from approximate analysis based on a previous study on annular tanks [1], and also to check the accuracy of the finite element solution developed and described in Chapter 3.

Test data on sloshing displacements was obtained for harmonic as well as simulated seismic-type ground motions. Details of the test procedure, experimental data, and comparison with results of approximate analysis based on annular tank theory are discussed in this chapter.

2.2 Model Description and Instrumentation

A simplified 1/60th scale model of a Mark 1 suppression pool was constructed as shown in Figs. 2-1a and 2-1b. The model was fabricated from 16 short lengths of 6 in. diameter clear plastic tubing cemented together to approximate complete torus. The internal details of the prototype, including the 'Headers' and 'Downcomers', were not reproduced in the model and not taken into account in the analysis. However, for reference purposes, the Header and Downcomer configuration is shown in Fig. 2-1C for a 1/30th scaled model designed for proof-tests by the reactor manufacturer. The mean diameter of the 1/60th scale model was 22 in. The normal operating water depth was 3 in.; that is, the water
surface at the section diameter, though water surface elevations above and below this level were also studied. The tank was mounted on a plywood base which was in turn prestressed to the shaking table.

The model was instrumented with one displacement gage located at a distance of $3/8$ in. from the inside wall. This gage was of the same type as used in the annular tank tests [1].

### 2.3 Test Procedure and Experimental Data

#### 2.3.1 Tests on the small shaking table

Sloshing frequencies of the 1/60th scale model of the Mark 1 torus and the steady state sloshing response under sinusoidal table motions were measured using the $3 \times 4$ ft shaking table described in Ref. [1] Sec. 4.4. The test set-up is shown in Fig. 2-2 and is similar to that described in Ref. [1] Sec. 4.5, the only difference being that this time a spectrum analyzer was used to determine the sloshing frequencies.

The test procedure to determine the steady state sloshing response under sinusoidal motions was the same as described in Ref. [1] Sec. 4.5 and was measured at the gage location shown in Fig. 2-la. To determine the sloshing frequencies, the frequency of the table motion was continuously changed and the water displacement signal was fed to the spectrum analyzer which was arranged to produce an averaged frequency spectrum. In this way the sloshing frequencies were read directly.

Tests on the small shaking table were not only conducted at the normal depth of 3 in., but also at depths of 2, 2.5, 3.5 and 4 in.
Test data regarding the sloshing frequencies and the wave heights are presented in Tables 2-1 and 2-2 respectively. Table 2-1 shows the mode number, the depth of water, and the measured as well as the analytical sloshing frequencies as approximated by using annular tank theory. Table 2-2 gives the steady-state response for harmonic ground motion, and tabulates the frequency of table motion, depth of water in the tank, amplitude of the table acceleration, and measured as well as the predicted values of the amplitudes of wave heights at the gage location. Two approximate solutions are given based on annular tank theory and are described in Sec. 2.4.

2.3.2 Tests under simulated earthquakes

Tests under simulated earthquake motion using the El Centro (1940) record were made at the University of California's Earthquake Simulator Laboratory. The test set-up of the 1/60th scale model of the Mark 1 torus is shown in Fig. 2-3 where the model is shown mounted on the 20 x 20 ft shaking table. Details of the shaking table facility and data acquisition system are given in Ref. [1] Chapter 5. The testing procedure was similar to that described in Ref. [1] Sec. 5.6. The prototype of the El Centro earthquake record was reduced by a factor of $\sqrt{60}$ to meet the similitude requirements of a 1/60th scale model.

In each test the quantities that were recorded, digitized and stored on magnetic tape included the horizontal and vertical table accelerations and displacements, and the water surface displacements in the model at the gage location shown in Fig. 2-1. The peak values of these quantities together, with the test numbers are shown in Table 2-3. The maximum and minimum water displacements in this table represents the
peak upward and downward displacements respectively. These tests were carried out both with and without the vertical component of the recorded ground motion. They were made at increasing amplitudes of ground motion in order to determine the range of linear response. Some selected results are plotted in Figs. 2-4 through 2-8. The center graph in Figs. 2-4 through 2-7 also shows a comparison of the measured displacement with approximate results from annular tank theory under horizontal ground motion only. Figure 2-8 shows that the vertical ground motion alone does not produce sloshing displacements as could be expected.

It should be remembered that a shaking table acts as a low-pass filter and will not fully reproduce motions at frequencies above the natural frequency of the system. The small time scale \( T_r = \sqrt{60} = 7.75 \) used in this study resulted in the acceleration peaks associated with high frequency ground motion being filtered out by the table. Hence the intensity of the earthquake given in Table 2-3, and measured by peak table acceleration, cannot be compared directly with the intensity of the original El Centro record. However the sloshing response is produced mainly by the low frequency components of the ground motion and these are correctly reproduced. It is estimated that the simulated earthquake ground motion in Table 2-3 with a peak acceleration of 0.34 g is equivalent to approximately 2.0 times the actual intensity of the El Centro earthquake in the significant low frequency components.

(Note: The recorded peak acceleration in the actual 1940 El Centro earthquake is 0.32 g.)
2.4 Comparison of Test Results with Approximate Theory

Approximate analyses were carried out using the computer program 'SLOSH' which is based on the annular tank theory developed in Ref. [1]. For carrying out the analyses the measured table acceleration was used and the tank radii a and b were taken as the horizontal distances from the axis of symmetry to the outer and inner walls where the free water surfaces come in contact with the solid boundaries. Two approximations were then made for the water depth as follows:

1. In the first approximation the depth of the water (h) in the torus tank solution was taken as the actual depth from the bottom of the torus to the free surface. This is called the 'Annular Tank' approximation in Tables 2-1 and 2-2.

2. In the second approximation, the depth of water used in the annular solution was adjusted to make the volume of water in the annular tank (with the same values of a and b as the torus tank) equal to the actual volume in the torus. This is called the 'Equivalent Volume' approximation in Tables 2-1 and 2-2.

Using these two approximations, analyses were carried out using the SLOSH program and the comparison of test and computer results for the sloshing frequencies and displacements is given.

2.4.1 Comparison of natural sloshing frequencies

Table 2-1 shows the comparison between the approximate results based on annular tank theory and the test values. It can be seen that for the first four modes, and within the range of water depth considered, the two approximations give very similar results. This could be expected
as in an annular tank over this range the natural frequencies are not very sensitive to water depth. Also, the approximate solutions compare well with the measured torus values in the 2 in. to 4 in. depth range indicating that frequencies are not very sensitive to the actual shape of cross-section of the tank.

2.4.2 Comparison of sloshing displacements under steady state harmonic ground motion

Table 2-2 gives a comparison of the measured and computed results from approximate annular tank theory for water depths ranging from 2.5 in. to 3.5 in. under steady state sinusoidal table motions varying in frequency from 1.50 Hz to 2.55 Hz. It can be observed in Table 2-2 that the annular tank theory gives satisfactory results for this range of water depth and the 'Equivalent Volume' approximation gives better results in most cases. It was also found that when the torus tank is nearly empty or nearly full (water depths less than 2 in. or greater than 4 in. in the model) as could be expected the approximate theory does not give satisfactory results and should not be used.

2.4.3 Comparison of sloshing displacements under simulated earthquake ground motions

Figures 2-4 through 2-7 show the time-history plots of the water surface displacements at a distance of 3/8" from the inside wall under increasing intensities of the simulated El Centro earthquake motion (horizontal component only). The depth of water in the torus tank was 3 in. in all these tests and the comparison of measured and approximate analytical displacements is given in the middle plot in each case. In Fig. 2-4
the analysis was carried out using approximation (1) (i.e., depth of water in the annular solution was taken as 3 in.). It can be seen that the measured and computed results differ by about 30% in displacement amplitude, and the measured frequency of response oscillation is somewhat lower. In Fig. 2-5 the 'Equivalent Volume' approximation was used (\( h = 2.36 \) in.) for the same ground motion and it can be seen that the agreement between the test and approximate analysis is better as regards displacement amplitude.

Figures 2-6 and 2-7 show a comparison at higher intensities of the ground motion where the analysis in both cases was done using the 'Equivalent Volume' annular tank approximation. It can be seen that even at these relatively large displacements the approximate theory, where a modified depth based on an equivalent volume is used in the annular tank solution, gives reasonably satisfactory results. This approximate theory should however be used with caution especially when the water depth is outside the range of that used in these tests.

2.5 Dynamic Pressures

The dynamic pressures were not measured on account of their small values and therefore no comparison is available between measured and analytical results. It is however anticipated that the annular tank theory should not be expected to give satisfactory results for the dynamic impulsive pressures in a torus tank and should not be used for this purpose. If the pressures are required in the torus, the Finite Element Method described in the next chapter should be used.
2.6 Linearity Range

The range of linear behavior of sloshing response was tested experimentally by subjecting the model to increasing intensities of the simulated El Centro earthquake (horizontal component only), and measuring the maximum (upward) and minimum (downward) peak water surface displacements. The displacements associated with this ground motion are primarily in the first radial mode, and hence the following comments on linearity are primarily related to displacements in this mode. The results are plotted in Fig. 2-9 and are tabulated in Table 2-3, together with some tests which included the vertical component of ground motion.

For displacements less than 0.1 in. in the model, linearity holds within 1%. But as the amplitude of displacement increases, nonlinearity also increases and becomes approximately 10% at displacements in the order of 0.4 in. estimated on the basis of both upward and downward displacements. For practical purposes it may be assumed that the linear theory gives satisfactory results as long as the displacements are less than 0.5 in. in the model (or 30 in. in the prototype Mark 1 suppression pool). It may be seen in Fig. 2-6 that the modified annular tank theory (Equivalent Volume approximation) gives quite satisfactory results although the displacements are of the order of 0.5 in.

2.7 Summary of Important Observations on Sloshing in Torus Tank Model

1) The overall sloshing behavior of water in the torus tank model studied in this report was very similar to that in the annular tank described in Ref. [1].
2) The effect of vertical ground motion on the sloshing displacements is negligible in the linear range (compare Test No. 220378.2 with Test No. 220378.10 in Table 2-3) but becomes more significant in the non-linear range (compare Test No. 220378.6 with Test No. 220378.13 in Table 2-3).

3) Modified annular tank theory gives satisfactory results for sloshing displacements in the torus tank. However this approximate theory cannot be applied for the determination of dynamic pressures.

4) A non-linearity of approximately 10% could be expected at displacements of the order of 0.4 in. in the model (or 24 in. in the prototype). Linear theory gives satisfactory results even under strong ground motions such as the El Centro earthquake of 1940.
TABLE 2-1.  Natural sloshing frequencies of small scale model (1/60-scale model).

<table>
<thead>
<tr>
<th>Mode No.</th>
<th>Depth of Water (in.)</th>
<th>Natural Frequencies (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Test</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0.45</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>2.15</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3.02</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>3.95</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>0.55</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>2.37</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>3.15</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4.15</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.35</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2.00</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>3.20</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>3.92</td>
</tr>
</tbody>
</table>
TABLE 2-2. Sloshing response of water in torus tank under sinusoidal ground acceleration (1/60th scale model) \( a = 14" \), \( b = 8" \).

<table>
<thead>
<tr>
<th>Frequency (Hz)</th>
<th>Depth (in.)</th>
<th>Table Acceleration (g)</th>
<th>Displacement at Inner Wall (in.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Test</td>
<td>Theory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(Equivalent Volume)</td>
<td>Annular Tank</td>
</tr>
<tr>
<td>1.5</td>
<td>3.0</td>
<td>0.0109</td>
<td>0.069</td>
</tr>
<tr>
<td>1.8</td>
<td>3.0</td>
<td>0.00392</td>
<td>0.040</td>
</tr>
<tr>
<td>1.8</td>
<td>3.0</td>
<td>0.00785</td>
<td>0.079</td>
</tr>
<tr>
<td>1.8</td>
<td>3.0</td>
<td>0.0118</td>
<td>0.119</td>
</tr>
<tr>
<td>1.8</td>
<td>3.0</td>
<td>0.0157</td>
<td>0.157</td>
</tr>
<tr>
<td>1.8</td>
<td>3.0</td>
<td>0.0196</td>
<td>0.179</td>
</tr>
<tr>
<td>1.9</td>
<td>3.0</td>
<td>0.00438</td>
<td>0.058</td>
</tr>
<tr>
<td>1.9</td>
<td>3.0</td>
<td>0.00875</td>
<td>0.116</td>
</tr>
<tr>
<td>1.9</td>
<td>3.0</td>
<td>0.0131</td>
<td>0.179</td>
</tr>
<tr>
<td>1.9</td>
<td>3.0</td>
<td>0.0175</td>
<td>0.240</td>
</tr>
<tr>
<td>1.9</td>
<td>3.0</td>
<td>0.0218</td>
<td>0.271</td>
</tr>
<tr>
<td>2.4</td>
<td>3.0</td>
<td>0.0242</td>
<td>0.246</td>
</tr>
<tr>
<td>1.50</td>
<td>2.5</td>
<td>0.0118</td>
<td>0.083</td>
</tr>
<tr>
<td>1.50</td>
<td>2.5</td>
<td>0.0233</td>
<td>0.165</td>
</tr>
<tr>
<td>1.80</td>
<td>2.5</td>
<td>0.0163</td>
<td>0.216</td>
</tr>
<tr>
<td>1.80</td>
<td>2.5</td>
<td>0.0326</td>
<td>0.444</td>
</tr>
<tr>
<td>1.80</td>
<td>2.5</td>
<td>0.0245</td>
<td>0.330</td>
</tr>
<tr>
<td>1.50</td>
<td>3.5</td>
<td>0.0118</td>
<td>0.072</td>
</tr>
<tr>
<td>1.80</td>
<td>3.5</td>
<td>0.0163</td>
<td>0.150</td>
</tr>
<tr>
<td>1.80</td>
<td>3.5</td>
<td>0.0326</td>
<td>0.310</td>
</tr>
<tr>
<td>2.55</td>
<td>3.5</td>
<td>0.0325</td>
<td>0.252</td>
</tr>
</tbody>
</table>
TABLE 2-3. Extreme values in torus tank model tests under simulated El Centro 1940 earthquake (time scale ($= \sqrt{60} = 7.75$, depth of water = 3 inches).

<table>
<thead>
<tr>
<th>Test No</th>
<th>Peak Table Acceleration (g)</th>
<th>Peak Table Displacement (inches)</th>
<th>Water Displacement (inches)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Horizontal</td>
<td>Vertical</td>
<td>Horizontal</td>
</tr>
<tr>
<td>220378.2</td>
<td>0.115</td>
<td>0.0</td>
<td>0.048</td>
</tr>
<tr>
<td>220378.3</td>
<td>0.164</td>
<td>0.0</td>
<td>0.073</td>
</tr>
<tr>
<td>220378.4</td>
<td>0.237</td>
<td>0.0</td>
<td>0.100</td>
</tr>
<tr>
<td>220378.5</td>
<td>0.284</td>
<td>0.0</td>
<td>0.125</td>
</tr>
<tr>
<td>220378.6</td>
<td>0.338</td>
<td>0.0</td>
<td>0.149</td>
</tr>
<tr>
<td>220378.7</td>
<td>0.049</td>
<td>0.0</td>
<td>0.023</td>
</tr>
<tr>
<td>220378.8</td>
<td>0.0</td>
<td>0.083</td>
<td>0.0</td>
</tr>
<tr>
<td>220378.9</td>
<td>0.0</td>
<td>0.260</td>
<td>0.0</td>
</tr>
<tr>
<td>220378.10</td>
<td>0.107</td>
<td>0.067</td>
<td>0.046</td>
</tr>
<tr>
<td>220378.11</td>
<td>0.170</td>
<td>0.103</td>
<td>0.075</td>
</tr>
<tr>
<td>220378.12</td>
<td>0.232</td>
<td>0.163</td>
<td>0.100</td>
</tr>
<tr>
<td>220378.13</td>
<td>0.358</td>
<td>0.217</td>
<td>0.151</td>
</tr>
<tr>
<td>230378.1</td>
<td>0.477</td>
<td>0.0</td>
<td>0.201</td>
</tr>
</tbody>
</table>
FIG. 2-1a 1/60 SCALE MODEL OF MARK I PRESSURE SUPPRESSION POOL
FIG. 2-1b CONSTRUCTION OF 1/60 SCALE MODEL OF MARK I TORUS PRESSURE SUPPRESSION POOL
FIG. 2-lc 1/30 SCALE MODEL OF MARK I SUPPRESSION POOL TORUS WITH HEADERS AND DOWNCOMERS

XBL 789-2280

2-15

11.5" ID, 0.5" WALL PLASTIC TUBING, 16 SECTIONS

MODEL HEADER AND DOWNCOMERS

LOAD-MEASURING HEADER SUPPORTS

MOUNTING PLATE

MOUNTING STRAPS
FIG. 2-2 1/60 SCALE MODEL OF TORUS TANK ON SMALL SHAKING TABLE

FIG. 2-3 1/60 SCALE MODEL ON 20ft X 20ft SHAKING TABLE
FIG. 2-4  SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 220378.2. PEAK SHAKING TABLE ACCELERATION = 0.115g HORIZONTAL, 0.0g VERTICAL.
Fig. 2-5  Sloshing response of water in a torus tank (inner radius = 8 in., outer radius = 14 in., depth of water = 3 in.) under simulated El Centro 1940 earthquake, time scale = 7.7, Test No. 220378.2. Peak shaking table acceleration = 0.115g horizontal, 0.0g vertical.
FIG. 2-6 SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 220378.4. PEAK SHAKING TABLE ACCELERATION = 0.237g HORIZONTAL, 0.0g VERTICAL.
FIG. 2-7  SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 22037-6. PEAK SHAKING TABLE ACCELERATION = 0.338g HORIZONTAL, 0.0g VERTICAL.
FIG. 2-8  SLOSHING RESPONSE OF WATER IN A TORUS TANK (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = 7.7, TEST NO. 220378.9. PEAK SHAKING TABLE ACCELERATION = 0.0g HORIZONTAL, 0.260g VERTICAL.
FIG. 2-9 VARIATION OF PEAK WATER DISPLACEMENTS WITH INTENSITY OF SIMULATED EL CENTRO 1940 EARTHQUAKE GROUND MOTION, TIME SCALE $\sqrt{60}$
3. FINITE ELEMENT ANALYSIS OF EARTHQUAKE INDUCED SLOSHING IN AXISYMMETRIC TANKS

3.1 Introduction

The finite element analysis [18] has become a powerful tool in solving complex engineering problems. Since the finite element method is completely general, it was decided that instead of looking for a closed form solution to predict the sloshing displacements and hydrodynamic pressures due to earthquake ground motions in a torus tank, the finite element method would be a better alternative in that it would be more general and thus can be applied to tank shapes other than cylindrical and toroidal.

Previous work on the finite element analysis of sloshing in tanks was done by Edwards [19] in which the shell theory was used for the prediction of seismic stresses and displacements in a cylindrical tank filled with liquid, but the sloshing was not considered in this analysis.

Finite element analysis for liquid sloshing problems by Luck [20] gives only the mode shapes and frequencies in an elastic container. His analysis is based on the variational principle suggested by Tong [21].

In this investigation our main concern is to study the sloshing effects in pressure-suppression pools of boiling water reactors, namely the Mark I torus and the Mark III annular suppression pools. Such structures can be considered as effectively rigid for the sloshing problem and thus the coupled effect of water-structure interaction is neglected in this analysis. Also the nonlinear sloshing problem has been linearized [13] for this analysis.
The finite element equations were first derived for a completely general three dimensional problem and then were specialized to an axisymmetric tank subjected to arbitrary horizontal ground motions. The finite element equations were derived using the Galerkin principle [22]. More background information on Galerkin method may be found in References [18] and [23-26].

3.2 Mathematical Formulation:

3.2.1 Equation of motion

Consider a tank of arbitrary shape with rigid walls filled with a liquid whose free surface area is B2 as shown in Fig. 3-1. B1 represents the surface area of liquid in contact with the solid boundary of the container. V is the volume of the liquid and \( \delta \) is the surface water displacement with respect to the undisturbed liquid surface. Using the same assumptions as in Ref. [1] Sec. 2.2, the velocity potential \( \phi \) exists at every point in V and must satisfy the Laplace equation which in rectangular coordinates can be written as:

\[
\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = 0
\]  

(3-1)

where \( \phi = \phi (x, y, z, t) \).

Equation (3-1) will be solved by the finite element method subject to the appropriate time dependent boundary conditions as specified below.
3.2.2 Boundary conditions

Let \( v_n(t) \) be the velocity of the tank wall along its outward normal to the boundary at any point, then:

\[
\frac{\partial \phi}{\partial n} = v_n(t) \quad \text{on } B_1
\]  

(3-2)

where \( n \) is the outward normal to the solid boundary and \( v_n \) is a function of time \( t \).

It can also be shown that a liquid particle on the free surface \( B_2 \) must satisfy the following two conditions [13]

\[
\frac{\partial \phi}{\partial x} \frac{\partial \delta}{\partial x} - \frac{\partial \phi}{\partial z} \frac{\partial \delta}{\partial y} + \frac{\partial \phi}{\partial y} \frac{\partial \delta}{\partial y} + \frac{\partial \phi}{\partial t} \frac{\partial \delta}{\partial t} = 0
\]

(3-3)

and

\[
g \delta + \frac{\partial \phi}{\partial t} + \frac{1}{2} \left( \left( \frac{\partial \phi}{\partial x} \right)^2 + \left( \frac{\partial \phi}{\partial y} \right)^2 + \left( \frac{\partial \phi}{\partial z} \right)^2 \right) = 0
\]

(3-4)

where \( g \) is the acceleration of gravity.

Equations (3-3) and (3-4) which represent the non-linear free surface boundary conditions can be simplified and combined into one boundary condition by neglecting higher order terms and eliminating \( \delta \). This single linearized boundary condition [13] can be written as follows.

\[
\frac{1}{g} \frac{\partial^2 \phi}{\partial t^2} + \frac{\partial \phi}{\partial z} = 0 \quad \text{on } B_2.
\]

(3-5)
3.3 Finite Element Formulation

3.3.1 Derivation of finite element equations

In the finite element analysis, the continuum is divided into discrete elements or subregions which are interconnected at a finite number of points called nodes. The necessary formulation follows the Galerkin principle where we let the unknown field variable $\phi$, throughout the solution domain, be approximated as

$$
\phi = \sum_{j=1}^{N} N_j(x, y, z) \phi_j(t)
$$

(3-6)

in which $N_j$ are the shape functions defined piecewise, element by element, and $\phi_j(t)$ are the time dependent nodal values of the field variable i.e., the velocity potential in this case. In the summation process an appropriate function for the particular point in space must be used. The $N$ nodal values $\phi_j$ are obtained by solving a set of $N$ simultaneous equations each derived by equating the boundary and interior residuals calculated by multiplying with a weighting function and integrating over the domain.

In the Galerkin approach, the shape functions are taken as the weighting functions and for a typical node $i$ substituting Eq. (3-6) into Eqs. (3-1), (3-2) and (3-5) and equating the weighted and integrated interior and boundary residual, we have:

$$
\int_{V} N_i \left( \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} \right) \sum_{j=1}^{N} N_j \phi_j d\nu = \int_{B1} N_i \frac{\partial}{\partial n} \sum_{j=1}^{N} N_j \phi_j d\sigma + \int_{B2} \left\{ \frac{N_i}{g} \frac{\partial^2 \phi}{\partial t^2} \sum_{j=1}^{N} N_j \phi_j + N_i \frac{\partial}{\partial z} \sum_{j=1}^{N} N_j \phi_j \right\} d\sigma
$$

(3-7)
in which \( \int dv \) and \( \int ds \) represent the integrals over the volume and appropriate surfaces respectively. Consider the first term on the left hand side of Eq. (3-7) and write it in the following form

\[
\int_v N_1 \sum_{j=1}^{N} \frac{\partial^2 N_j}{\partial x^2} \phi_j \, dv = \int_v \frac{\partial}{\partial x} \left( N_1 \sum_{j=1}^{N} \frac{\partial N_j}{\partial x} \right) \phi_j \, dv - \int_v \frac{\partial}{\partial x} \sum_{j=1}^{N} \frac{\partial N_j}{\partial y} \phi_j \, dv
\]

(3-8)

Applying the Divergence theorem on the first integral on the right hand side of Eq. (3-8), we can rewrite it in the following form.

\[
\int_v N_1 \sum_{j=1}^{N} \frac{\partial^2 N_j}{\partial x^2} \phi_j \, dv = \int_{B_1} N_1 \sum_{j=1}^{N} \frac{\partial N_j}{\partial x} \, \ell_x \phi_j \, ds - \int_v \sum_{j=1}^{N} \frac{\partial N_j}{\partial x} \phi_j \, dv
\]

(3-9)

in which \( B = B_1 + B_2 \) and \( \ell_x \) is the direction cosine in the x-direction of the outward normal \( n \). Similar expression can be written for the second and third terms in Eq. (3-7). Substituting Eq. (3-9) and similar expressions into the right hand side of Eq. (3-7), we get

\[
\int_{B_1} N_1 \left( \sum_{j=1}^{N} \frac{\partial N_j}{\partial x} \ell_x \phi_j + \sum_{j=1}^{N} \frac{\partial N_j}{\partial y} \ell_y \phi_j + \sum_{j=1}^{N} \frac{\partial N_j}{\partial z} \ell_z \phi_j \right) \, ds
\]

\[
- \int_v \left( \sum_{j=1}^{N} \frac{\partial N_j}{\partial x} \phi_j + \sum_{j=1}^{N} \frac{\partial N_j}{\partial y} \phi_j + \sum_{j=1}^{N} \frac{\partial N_j}{\partial z} \phi_j \right) \, dv
\]

(3-10)

\[
= \int_{B_1} N_1 \sum_{j=1}^{N} \frac{\partial N_j}{\partial n} \phi_j \, ds + \int_{B_1} N_1 v \, ds + \frac{1}{g} \int_{B_1} N_1 \sum_{j=1}^{N} \phi \, ds + \int_{B_2} N_1 \sum_{j=1}^{N} \frac{\partial N_j}{\partial z} \phi_j \, ds
\]

in which \( \phi = \frac{d^2 \phi}{dt^2} \).
The boundary integral on the left hand side of Eq. (3-10) can be substituted by

\[ \int_B N_i \sum_{j=1}^{N} \frac{\partial N_i^j}{\partial \mathbf{n}} \phi_j \, ds \]  

(3-11)

or by

\[ \int_{B_1} N_i \sum_{j=1}^{N} \frac{\partial N_i^j}{\partial \mathbf{n}} \phi_j \, ds + \int_{B_2} N_i \sum_{j=1}^{N} \frac{\partial N_i^j}{\partial \mathbf{n}} \phi_j \, ds. \]  

(3-12)

Replacing the boundary integral of left hand side of Eq. (3-10) with Eq. (3-12) and using the approximation (small slopes)

\[ \frac{\partial N_i}{\partial \mathbf{z}} \approx \frac{\partial N_i}{\partial \mathbf{n}} \quad \text{on } B_2, \quad \text{Eq. (3-10)} \]

can be simplified to the following form.

\[ \int_{B_1} \left[ \frac{\partial N_i}{\partial x} \sum_{j=1}^{N} \frac{\partial N_i^j}{\partial x} \phi_j + \frac{\partial N_i}{\partial y} \sum_{j=1}^{N} \frac{\partial N_i^j}{\partial y} \phi_j + \frac{\partial N_i}{\partial z} \sum_{j=1}^{N} \frac{\partial N_i^j}{\partial z} \phi_j \right] \, dv \]

(3-13)

\[ + \frac{1}{g} \int_{B_2} N_i \sum_{j=1}^{N} N_j \phi_j \, ds = \int_{B_1} N_i \mathbf{v} \, ds \]

or

\[ M \phi + K \phi = F \]

(3-14)

in which the elements of \( M, K \) and \( F \) are given by

\[ M_{ij} = \frac{1}{g} \int_{B_2} N_i N_j \, ds \]  

(3-15)

\[ K_{ij} = \int_{B_2} \left( \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} + \frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z} \right) \, dv \]  

(3-16)

\[ F_i = \int_{B_1} N_i \mathbf{v} \, ds \]  

(3-17)
where summation for $M_{ij}$ covers only the elements on the free surface boundary and the integral is carried out on the free surface of each element $EB_2$. Summation for $K_{ij}$ covers the contribution of each element and $EV$ is the element region. $EB_1$ refers only to the elements which lie on the solid boundary, $Bl$, and the loading term thus is associated with the elements that lie on the tank wall boundary.

The free surface matrix $M$ and the fluid matrix $K$ are comparable to the mass and stiffness matrices respectively used in structural dynamics. It is interesting to note that the free surface matrix $M$ gets the contribution only from the free surface elements. $M$ and $K$ are symmetric matrices and Eq. (3-14) is a set of second order linear differential equations which can be solved either by direct integration or by mode superposition.

3.3.2 Isoparametric formulation for axisymmetric tank under arbitrary horizontal ground motions

The following analysis will be restricted to rigid tanks which are symmetrical about the z-axis and are subjected to arbitrary horizontal ground motions alone. Since Eq. (3-13) involves only the first derivatives of shape functions, a 4-node quadrilateral element with linear interpolation functions will satisfy the convergence requirements. However, the more recently developed 4-to-8 variable node isoparametric element [27] has greater flexibility in accommodating the curved boundaries and is convenient for numerical integration. Therefore a variable 4-to-8 node, 2-dimensional isoparametric element will be used in the present formulation. Such an element can be used to model an axisymmetric problem or a two dimensional problem such as sloshing in a rectangular tank.
Figure 3-2(a) shows such a 4-to-8 variable node element lying in the x-z plane where z is the axis of symmetry. Any of the mid-side nodes 5 through 8 may or may not be present and can be eliminated if desired. Such a curvilinear 4-to-8 node element can be obtained by using an isoparametric mapping from a bi-unit square which has a local r-s coordinate system as shown in Fig. 3-2(b). The local coordinates r and s vary between -1 and +1. The nodes 1 through 4 are the corner nodes and the nodes 5 through 8 are the mid-side nodes corresponding to Fig. 3-2(a). Node 1 has the coordinates (1,1). The mapping between the local coordinate system (r,s) and the global (x,z) coordinate system must be unique in order to carry out the transformations properly. The coordinate transformation between the bi-unit square and the curvilinear element is given by:

\[
x_m(r,s) = \sum_{i=1}^{8} h_i(r,s) x_{im}
\]

\[
z_m(r,s) = \sum_{i=1}^{8} h_i(r,s) z_{im}
\]

in which \((x_{im}, z_{im})\) are the global coordinates of node \(i\) in element \(m\) and \(h_i\) are the interpolation functions in local coordinates corresponding to node \(m\). The interpolation functions \(h_i\) for any element \(m\) are defined as follows:
\[ h_1 = \frac{1}{4} (1+r)(1+s) - \frac{1}{2} h_5 - \frac{1}{2} h_8 \]
\[ h_2 = \frac{1}{4} (1-r)(1+s) - \frac{1}{2} h_5 - \frac{1}{2} h_6 \]
\[ h_3 = \frac{1}{4} (1-r)(1-s) - \frac{1}{2} h_6 - \frac{1}{2} h_7 \]
\[ h_4 = \frac{1}{4} (1+r)(1-s) - \frac{1}{2} h_7 - \frac{1}{2} h_8 \]
\[ h_5 = \frac{1}{2} (1-r^2)(1+s) \]
\[ h_6 = \frac{1}{2} (1-r)(1-s^2) \]
\[ h_7 = \frac{1}{2} (1-r^2)(1-s) \]
\[ h_8 = \frac{1}{2} (1+r)(1-s^2). \]  

(3-20)

Since a horizontal ground motion will excite only the antisymmetric modes (in the linearized case) of sloshing in an axisymmetric tank, we can approximate the distribution of the velocity potential within an element \( m \) in terms of the velocity potential at node 1 through 8 and the interpolation functions given by Eq. (3-20)

\[ \phi_m (r,s,\theta,t) = \sum_{n=1}^{\infty} \sum_{i=1}^{8} h_i(r,s) \cos n\theta \phi_{im}(t) \]  

(3-21)

in which \( \phi_{im} \) is the value of the velocity potential at node \( i \) of element \( m \).

It is obvious that if this shape function given by Eq. (3-21) is used in Eq. (3-17) to calculate the loading vector, the integral between the limits 0 and \( 2\pi \) will be non-zero only when \( n = 1 \) in the case of horizontal ground motion only, because \( v_n \) in such a case varies as a function
of \cos \theta and the \int_0^{2\pi} \cos \theta \cdot \cos n \theta \, d\theta = 0 \text{ for any } n \neq 1. Therefore Eq. (3-21) can be written as

\[ \phi_m(r,s,\theta,t) = \sum_{i=1}^{8} h_i(r,s) \cos \theta \cdot \phi_{im}(t) \]  
(3-22)

and

\[ \frac{\partial}{\partial r} \phi_m(r,s,\theta,t) = \sum_{i=1}^{8} \frac{\partial}{\partial r} h_i \cos \theta \cdot \phi_{im}(t) \]  
(3-23)

\[ \frac{\partial}{\partial s} \phi_m(r,s,\theta,t) = \sum_{i=1}^{8} \frac{\partial}{\partial s} h_i \cos \theta \cdot \phi_{im}(t) \]  
(3-24)

or in matrix form, the above two equations can be written as

\[ \begin{bmatrix} \frac{\partial \phi}{\partial r} \\ \frac{\partial \phi}{\partial s} \end{bmatrix}_m = \cos \theta \begin{bmatrix} p_m(r,s) \\ \phi_m(t) \end{bmatrix} \]  
(3-25)

in which \( p_m(r,s) \) contains the derivatives of interpolation functions derived from Eq. (3-20). Using the chain rule of differentiation, we can relate the global derivatives to the local derivatives.

\[ \begin{bmatrix} \frac{\partial \phi}{\partial r} \\ \frac{\partial \phi}{\partial s} \end{bmatrix}_m = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial z}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial z}{\partial s} \end{bmatrix} \begin{bmatrix} \frac{\partial \phi}{\partial x} \\ \frac{\partial \phi}{\partial z} \end{bmatrix}_m \]  
(3-26)

Jacobian matrix
by inverting

$$\begin{bmatrix}
\frac{\partial \phi}{\partial x} \\
\frac{\partial \phi}{\partial z}
\end{bmatrix}_m = J^{-1}_m
\begin{bmatrix}
\frac{\partial \phi}{\partial r} \\
\frac{\partial \phi}{\partial s}
\end{bmatrix}_m$$

(3-27)

in which $J^{-1}_m$ is the inverse of the Jacobian matrix in Eq. (3-26). Substituting for $\partial \phi / \partial r$ and for $\partial \phi / \partial s$ from Eq. (3-25) into Eq. (3-27), we obtain the relationship between the global derivatives and nodal values of $\phi$.

$$\begin{bmatrix}
\frac{\partial \phi}{\partial x} \\
\frac{\partial \phi}{\partial z}
\end{bmatrix}_m = J^{-1}_m \begin{bmatrix}
\frac{\partial \phi}{\partial r} \\
\frac{\partial \phi}{\partial s}
\end{bmatrix}_m$$

$$= \begin{bmatrix}
\frac{p_m}{2 \times 2} & \frac{m}{2 \times 8} & \frac{8 \times 1}
\end{bmatrix}
\begin{bmatrix}
(r, s) \\
\phi_m(t) \cdot \cos \theta
\end{bmatrix}$$

(3-28)

or

$$\begin{bmatrix}
\frac{\partial \phi}{\partial x} \\
\frac{\partial \phi}{\partial z}
\end{bmatrix}_m = B_m \begin{bmatrix}
\phi_m(t) \cos \theta
\end{bmatrix}$$

(3-29)

in which

$$B_m = J^{-1}_m \begin{bmatrix}
p_m
\end{bmatrix}.$$)

(3-30)

Thus

$$\begin{bmatrix}
\frac{\partial N}{\partial x} \\
\frac{\partial N}{\partial y}
\end{bmatrix}_m = B_m \cos \theta$$

(3-31)
3.3.3 Free surface (mass) matrix for axisymmetric element

The complete free surface matrix for the system is formed by direct summation of individual element matrices i.e.,

\[ M = \frac{1}{g} \sum_{m=1}^{n} M_m \]  \hspace{1cm} (3-32)

where \( n \) is the total number of free surface elements, and the element matrix \( M_m \) is a 2 \( \times \) 2 matrix given by

\[ M_m = \int_{EB2} N^T N \, ds. \]  \hspace{1cm} (3-33)

In case of axisymmetric case the surface integral can be transformed to a line integral. Consider a free surface element with nodes \( i \) and \( j \) (Fig. 3-3) and let

\[ r = \text{local coordinates for free surface boundary element} \]
\[ R(r) = \text{radius to any point } r \text{ between node } i \text{ and } j \]
\[ L = \text{length of the elements} \]
\[ L = \sqrt{(x_i - x_j)^2 + (z_i - z_j)^2}. \]  \hspace{1cm} (3-34)

Assuming that \( x \)-axis coincides with the horizontal plane, we can write the transformation.

\[ R(r) = \begin{bmatrix} 1 - r/L & r/L \end{bmatrix} \begin{bmatrix} x_i \\ x_j \end{bmatrix}. \]  \hspace{1cm} (3-35)

\[ ds = R(r) \, dr \, d\theta \]  \hspace{1cm} (3-36)

where \( \theta \) is defined in Fig. 3-2

\[ ds = \begin{bmatrix} 1 - r/L & r/L \end{bmatrix} \begin{bmatrix} x_i \\ x_j \end{bmatrix} \, dr \, d\theta. \]  \hspace{1cm} (3-38)
Thus

\[ N = \begin{bmatrix} 1 - \frac{r}{L} & \frac{r}{L} \end{bmatrix} \cos \theta. \]  

(3-39)

Therefore

\[ M_{m-n} = \int_0^{2\pi} \int_0^L \left( \frac{1-r/L}{r/L} \right) \left( \frac{1-r/L}{r/L} \right) R(r) \cos^2 \theta \cdot dr \, d\theta. \]  

(3-40)

\[ M_{m-n} = \pi \int_0^L \left( \frac{1-r/L}{r/L} \right) R(r) \, dr \, d\theta. \]  

(3-41)

Evaluation of the above matrix gives

\[ M_{ii} = \frac{\pi L}{4} \left[ x_i + \frac{X}{3} \right] \]  

(3-42)

\[ M_{jj} = \frac{\pi L}{4} \left[ \frac{x_i}{3} + x_j \right] \]  

(3-43)

\[ M_{ij} = \frac{\pi L}{12} \left[ x_i + x_j \right] \]  

(3-44)

3.3.4 Evaluation of fluid (stiffness) matrix for axisymmetric element

The complete system fluid matrix is formed by direct summation of element matrices

\[ K = \sum_{m=1}^n K_{m-n} \]  

(3-45)

where \( n \) is the number of liquid elements and the element matrix \( K_{m-n} \) is obtained by using Eq. (3-31) and Eq. (3-16).

\[ K_{m-n} = \int_{EV} \cos^2 \theta \, B_m^T \, B_m \, dv \]  

(3-46)

In case of an axisymmetric element, \( dv = R \, d\theta \, dr \), the volume integral then becomes
\[
K_m = \int_0^{2\pi} \int_{A_m} \cos^2 \theta \ R B_m^T B_m \ dA d\theta 
\]  

(3-47)

where \( R \) is the radius of any point and \( A \) is the area.

In the natural coordinate system \( dA = |J_m| \ dsdr \) where \( |J_m| \) is the determinant of the Jacobian matrix, therefore

\[
K_m = \pi \int_{-1}^{1} \int_{-1}^{1} R B_m^T B_m |J_m| \ dsdr. 
\]  

(3-48)

The above integral can be evaluated numerically using Gaussian quadrature as follows [18]

\[
K_m = \sum_{i=1}^{N} \sum_{j=1}^{N} H_i H_j f_i (r_i, s_i) 
\]  

(3-49)

where \( N \) refers to the order of integration; \( H_i \) and \( H_j \) are the weighting factors and

\[
f_i (r_i, s_i) = B_m^T (r_i, s_i) B_m (r_i, s_i) |J_m| (r_i, s_i) |R (r_i, s_i) |
\]  

(3-49)

3.3.5 Load vector for axisymmetric element

The loading vector \( F \) for the complete system is the sum of the contribution of individual elements and using Eq. (3-17) we can write

\[
F = \sum_{m=1}^{n} \int_{EB1} N_m^T \nu_m \ ds = \sum_{m=1}^{n} \frac{F_m}{m} 
\]  

(3-50)

where

\[
\frac{F_m}{m} = \int_{EB1} N_m^T \nu_m \ ds 
\]  

(3-51)

where summation is over all the \( n \) elements which are at the liquid-solid interface.
Consider a typical liquid-solid boundary element with nodes \( i \) and \( j \) and let \( r \) be a local coordinate as shown in Fig. 3-4. Let \( R(r) \) be the radius at any point \( r \) and \( L \) be the length of the element where
\[
L = \sqrt{(x_i - x_j)^2 + (z_i - z_j)^2}.
\]

\[
R(r) = \left[ (1-r/L) \begin{array}{c} r/L \\ x_i \\ x_j \end{array} \right]. \tag{3-52}
\]
\[
N = \left[ (1-r/L) \begin{array}{c} r/L \\ \cos \theta \end{array} \right]. \tag{3-53}
\]
\[
ds = R(r) \, dr \, d\theta.
\]

If \( v_x \) is horizontal ground velocity in the \( x \)-direction then
\[
v_n = v_x \cos \theta \cos \psi \quad \text{where} \quad \psi = \tan^{-1} \left( \frac{x_i - x_j}{z_i - z_j} \right);
\]
\[
F_m = v_x \int_0^L \int_0^{2\pi} \left( \begin{array}{c} (1-r/L) \\ r/L \end{array} \right) \left( \begin{array}{c} x_i \\ x_j \end{array} \right) \cos^2 \theta \cos \psi \, d\theta \, dr. \tag{3-54}
\]
integrating we get the following element load vector
\[
F_m = \frac{\pi LV_x \cos \psi}{6} \left( \begin{array}{c} 2x_i + x_j \\ x_i + 2x_j \end{array} \right) = \left( \begin{array}{c} F_i \\ F_j \end{array} \right). \tag{3-55}
\]

The contribution to the loading matrix comes from those elements which lie at the liquid-solid interface.

### 3.4 Numerical Solution of Finite Element Equations

The discretization of the continuum into finite elements and the assemblage of free surface, liquid and loading element matrices results in a set of linear, coupled, second order ordinary differential equations. Since these equations are linear they can be uncoupled by an orthogonal transformation and the solution can be obtained using mode superposition [28, 29].
or they can be solved by direct step-by-step integration. In this study Newmark's step by step integration method [30,29] which is based on the following expressions was used.

\[ \dot{\phi}_{t+\Delta t} = \dot{\phi}_t + \Delta t \left( 1-\delta \right) \ddot{\phi}_t + \Delta t \delta \ddot{\phi}_{t+\Delta t} \]  \hspace{1cm} (3-56)

\[ \phi_{t+\Delta t} = \phi_t + \Delta t \dot{\phi}_t + \Delta t^2 \left( \frac{1}{2} - \alpha \right) \ddot{\phi}_t + \Delta t^2 \alpha \ddot{\phi}_{t+\Delta t} \]  \hspace{1cm} (3-57)

in which \( \Delta t \) is step size. \( \alpha \) and \( \delta \) are parameters which are selected to produce the desired stability and accuracy. In all the sample analyses carried out in this investigation, Newmark's constant-average-acceleration method \((\delta = 1/2 \text{ and } \alpha = 1/4)\) was used, which is an unconditionally stable method without numerical damping.

This is an implicit method and satisfies the equilibrium equations at time \( t + \Delta t \), i.e.,

\[ M \ddot{\phi}_{t+\Delta t} + K \dot{\phi}_{t+\Delta t} = F_{t+\Delta t} \]  \hspace{1cm} (3-58)

The above three equations can be combined into a step-by-step algorithm which involves the solution of a set of equations at each time step of the form.

\[ K^{*} \ddot{\phi}_{t+\Delta t} = F^{*} \]  \hspace{1cm} (3-59)

In this analysis \( K^{*} \) is independent of time and is formed and triangularized only once. To make the numerical algorithm more general, the option of combining the Wilson Theta method [31] with the Newmark method was incorporated in the computer program. The Wilson Theta method was first applied to Newmark's linear accelerator method in order to improve stability and to damp out high frequency oscillations which often develop in step by step integration. A summary of the Newmark-Wilson algorithm used in the computer program is given below.
3.4.1 The Newmark-Wilson algorithm for linear step-by-step integration

INITIAL CALCULATIONS

1. Initialize \( \phi_0 \) and \( \dot{\phi}_0 \) (taken to be zero) (\( \phi \) is a vector)
2. Form the free surface and fluid matrices \( (M, K) \)
3. Specify algorithm parameters \( \alpha, \delta \) and \( \theta \)

\[
\begin{align*}
\tau &= \theta \Delta t \\
a_0 &= \frac{1}{\alpha \tau} \\
a_1 &= \frac{\delta}{\alpha \tau} \\
a_2 &= \frac{1}{\alpha \tau} \\
a_3 &= \frac{1}{2\alpha} - 1 \\
a_4 &= \frac{\delta}{\alpha} - 1 \\
a_5 &= \frac{\tau}{2} (\delta/\alpha - 2) \\
a_6 &= \Delta t (1-\delta) \\
a_7 &= \Delta t \delta \\
a_8 &= \Delta t^2 \left( \frac{1}{2} - \alpha \right) \\
a_9 &= \alpha \Delta t^2
\end{align*}
\]

5. Form \( K^* = K + a_0 M \)
6. Triangularize \( K^* \): \( K^* = LDL^T \)

FOR EACH TIME STEP

1. Calculate the effective load vector \( F^* \) at time \( t+\tau \)

\[
F^* = F_{t+\tau} + M (a_0 \phi_t + a_2 \dot{\phi}_t + a_3 \ddot{\phi}_t).
\]

2. Solve for velocity potential \( \phi \) at \( t + \tau \):

\[
LDL^T \phi_{t+\tau} = F^*
\]

3. Calculate the velocity potential \( \phi \) and its derivatives at time \( t+\Delta t \):

\[
\begin{align*}
\phi_{t+\Delta t} &= \phi_t + \frac{\Delta t}{\theta} (\phi_{t+\tau} - \phi_t) \\
\dot{\phi}_{t+\Delta t} &= \dot{\phi}_t + \frac{\Delta \ddot{\phi}_t}{\theta} \\
\ddot{\phi}_{t+\Delta t} &= \ddot{\phi}_t + \frac{\Delta t}{\theta} \left( \frac{\delta}{\alpha} \phi_{t+\tau} - \ddot{\phi}_t \right) + \frac{\Delta \dddot{\phi}_t}{\theta} \\
\phi_{t+\Delta t} &= \phi_t + \Delta t \phi_t + a_8 \phi_t + a_9 \phi_{t+\Delta t}
\end{align*}
\]
4. Determine the sloshing displacements and hydrodynamic impulsive pressures at time \( t + \Delta t \)

\[
\delta_{t+\Delta t} = - \frac{1}{g} \phi_{t+\Delta t}
\]

\[
P_{t+\Delta t} = - \rho \phi_{t+\Delta t}
\]

3.5 Computer Program 'SLOSH2'

The program 'SLOSH2' developed to implement the finite element theory of the sloshing phenomenon in axisymmetric tanks is coded in standard FORTRAN IV language. The basic set up of SLOSH2 is the same as that of the computer program DOT [32] because of the fact that the finite element formulation of the sloshing problem is similar in certain respects to that of the heat conduction equations.

The earthquake input can either be as an accelerogram or a displacement-time history, digitized in the appropriate format. The program derives the velocity-time history by integration or differentiation depending upon the type of ground motion input. The earthquake input must be properly adjusted for base-line correction such that at the end of earthquake as acceleration goes to zero, the ground velocity and displacement also go to zero.

The 'effective' equilibrium equations (Eq. (3-59) are solved using the linear equation solver COLSOL [33]. This subroutine processes only those elements which are within the skyline of \( K^* \), thus minimizing the storage requirements as well as the number of operations. This subroutine is based on Gauss elimination and requires a symmetrical positive-definite system of equations.
A compact storage scheme is used in the computer program whereby a one-dimensional array is used to store only those elements of $K^*$ which are within its skyline. In the actual implementation of the computer program a lumped mass parameter system was used which not only simplifies the analysis considerably, but also minimizes the storage requirements. The three finite element groups namely, the free surface elements, the fluid elements and the liquid-solid interface elements, are processed in blocks and then stored on the disc for later use in order to increase the maximum capacity of the program.

It should be noted that the free surface elements and the liquid-solid interface elements which contribute to $M$ and $F$ respectively, are only two node elements for the axisymmetric case, whereas the liquid continuum itself is discretized by two-dimensional 4-to-8 node isoparametric elements which contribute to the fluid matrix $K$.

In this computer program, a variable dimension is used for dynamic allocation of primary storage into a single array in blank common. The lower primary storage locations are used for the storage of each block of element group data which is read in from the secondary storage (disc) as required during the solution phase. The user has to supply the maximum estimated number of storage locations required to store any individual element group in the lowest primary storage. Usually this number is determined by the number of liquid elements that form the $K$ matrix and not the free surface or liquid-solid interface elements. An estimation of the C.P.U. time required to run the program on CDC 6400 will be indicated in the next article. A user's manual and Fortran listing for SLOSH2 are given in Appendix A1 and A2 respectively.
3.6 Sample Analyses and Comparison with Test Data from Annular and Torus Tanks

Figures 3-5 and 3-6 show the finite element mesh layout for the 8 ft diameter annular tank and the 28 inches diameter torus tank models respectively which were studied in Ref. [1] and in Chapter 2 of this report. The annular tank chosen for this analysis is a simplified 1/15th scale model of pressure-suppression pool of Boiling Water Reactor GE Mark III (see Ref. [1] Figs. 5-3 and 5-7a) whereas the torus tank represents the 1/60th scale model (see Fig. 2-1) of the GE Mark I pressure-suppression pool. These analyses were carried out to check the accuracy of the finite element model against precise test data. A finer mesh size has been used near the free water surfaces because that is where the maximum sloshing displacements occur.

3.6.1 Annular tank

In Fig. 3-5 the x-axis is taken at the bottom of the tank and z-axis as the axis of symmetry of the annular tank. There are a total of 25, 10 and 17 elements in group Nos. 1, 2 and 3 respectively. Group No.1 contains 4-to-8 node elements whereas group Nos. 2 and 3 contain the free surface elements and liquid-solid interface elements respectively with two nodes each. There are a total of 52 nodes in this mesh. The finite element analysis was done using the digitized accelerogram recorded in Test No. 211276.1 of Ref. [1] Chapter 5 and a comparison of measured and predicted results is given in Figs. 3-7 and 3-8.

Figure 3-7 shows a comparison of the sloshing displacements at node 2 between the finite element solution and the test data under the
same ground motion. The results are shown for the first six seconds and it can be seen that there is close agreement between the test and finite element results.

Figure 3-8 shows a comparison of impulsive pressures between the measured and finite element results, and it can be seen again that the agreement between the two is excellent with test results consistently 5-10% higher compared with the finite element solution indicating a possible error of calibration of pressure gage and some error due to electrical noise. This comparison is given at node 52 (Fig. 3.5).

3.6.2 Torus tank

The finite element mesh layout for the torus tank model (Fig. 3-6) has a total of 24, 12 and 24 elements in group Nos. 1, 2 and 3 respectively. The z-axis is taken as the axis of symmetry of the torus and the x-axis at the bottom of the tank. The total number of nodes in this case is 73.

The earthquake input used for the finite element analysis was the recorded shaking table displacement instead of acceleration because the displacements gave zero velocity and acceleration on differentiation at the end of the earthquakes whereas the integration of the accelerogram usually gives finite amount of velocity and displacement without a base line correction.

In comparing the finite element solution with the test data for the torus tank it should be remembered that the two systems are not exactly the same: the finite element solution is for a perfectly round tank, whereas the test tank was made of a set of 16 straight segments as shown in Fig. 2-1. Figures 3-9, 3-10 and 3-11 show the comparison of sloshing displacements at node No.2 for increasing intensity of ground motion. In Figs. 3-9 and 3-10 the agreement between the two results during the first half
of the earthquake is excellent and for the rest of the earthquake quite satisfactory. In Fig. 3-11, although the sloshing response is relatively large, the linear finite element analysis still gives satisfactory results for practical purposes. The hydrodynamic pressures in this case were not measured on account of their relatively small magnitude in this small scale model.

The C.P.U. time required by the CDC 6400 to obtain the time-history response in the torus tank model analysis (Fig. 3-9) with the mesh size shown in Fig. 3-6 was 126 seconds.

3.7 Sample Analysis of Mark I Prototype Torus Tank Under El Centro 1940 Earthquake

Figure 3-12 shows the sloshing response in the prototype of Mark I torus under the full intensity of the 1940 El Centro earthquake (N-S component). The mesh layout for this analysis was similar to that shown in Fig. 3-6 except the overall dimensions which in this case are 60 times larger. The sloshing displacement shown in Fig. 3-12 is at node #2 with a maximum value of 24.3".
FIG. 3-1  TANK OF ARBITRARY SHAPE FILLED WITH LIQUID

XBL 789-2282
(a) TWO-DIMENSIONAL ELEMENT IN GLOBAL X-Z SYSTEM

(b) BI-UNIT SQUARE IN LOCAL r-S SYSTEM

FIG. 3-2 TWO-DIMENSIONAL MAPPING OF AN ISOPARAMETRIC ELEMENT

XBL 789-2275
FIG. 3-3 FREE SURFACE ELEMENT (AXISYMMETRIC CASE)

FIG. 3-4 BOUNDARY ELEMENT AT LIQUID-SOLID INTERFACE (AXISYMMETRIC CASE)
FIG. 3-5  FINITE ELEMENT MESH FOR ANNULAR TANK MODEL
FIG. 3-6  FINITE ELEMENT MESH LAYOUT FOR TORUS TANK MODEL
FIG. 3-7  COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #2 (TEST #211276.1) IN ANNULAR TANK MODEL (INNER RADIUS = 33.2 IN., OUTER RADIUS = 48.0 IN., DEPTH OF WATER = 16 IN.) UNDER SIMULATED EL CENTRO 1940 EARTHQUAKE, TIME SCALE = $\sqrt{T_5} = 3.9$, PEAK SHAKING TABLE ACCELERATION = 0.24g HORIZONTAL, 0.0g VERTICAL.
FIG. 3-8 COMPARISON OF IMPULSIVE PRESSURES BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #52 (TEST #211276.1) IN ANNULAR TANK MODEL.
FIG. 3-9 COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #2 (TEST #220378.2) IN TORUS TANK MODEL (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.)
FIG. 3-10 COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #2 (TEST #220378.3) IN TORUS TANK MODEL (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.)

PEAK ACCELERATION = 0.164 g
FIG. 3-11 COMPARISON OF SLOSHING DISPLACEMENTS BETWEEN THE TEST AND FINITE ELEMENT RESULTS AT NODE #2 (TEST #220378.4) IN TORUS TANK MODEL (INNER RADIUS = 8 IN., OUTER RADIUS = 14 IN., DEPTH OF WATER = 3 IN.)
FIG. 3-12  SLOSHING DISPLACEMENTS IN PROTOTYPE TORUS TANK AT NODE #2 UNDER ELCENTRO 1940 EARTHQUAKE (INNER RADIUS = 40 FT., OUTER RADIUS = 70 FT., DEPTH OF WATER = 15 FT.)
4. CONCLUSIONS

(1) A comparison of data from shaking table tests on annular and torus tanks confirms that the finite element analysis presented in this report can successfully predict hydrodynamic pressures and free surface displacement in rigid axisymmetric tanks under strong motion earthquakes. The finite element program is applicable not only to axisymmetric tanks but may also have possible application to offshore structures under seismic conditions.

(2) As sloshing response is not very sensitive to the precise geometry of the tank section, a modified annular tank solution gives satisfactory results in predicting the sloshing frequencies and displacements in torus tanks under horizontal ground motions.

(3) The validity of the theory developed herein is independent of the size of the model that was analyzed and tested. In the 1/60-scale torus model the sloshing response is produced mainly by the low frequency components of the reference earthquake ground motion, and these are correctly reproduced on the 20-foot shaking table.
ACKNOWLEDGMENTS

This investigation was sponsored by the United States Nuclear Regulatory Commission and was done with support from the United States Department of Energy.
REFERENCES


APPENDIX A1

SLOSH2 USER'S MANUAL

SLOSH2: A linear finite element program which determines the sloshing displacements and impulsive pressures in axisymmetric rigid tanks under arbitrary horizontal ground motions.

Developed by: Mohammad Aslam
Department of Civil Engineering
University of California, Berkeley

CONTENTS

A1.1 Program Description
A1.2 Program Capacity
A1.3 Input Data
    I. Problem Initiation and Title
    II. Master Control Card
    III. Nodal Coordinates
    IV. Solution Time and Step Size
    V. Earthquake Input
    VI. Element Data
        Type 1: Two Dimensional Finite Elements
        Type 2: Free Surface Elements
        Type 3: Solid-Liquid Interface Elements
    VII. New Problem Data
    VIII. Termination Card
A1.4 Output
Al.1 Program Description

The computer program SLOSH2 has been developed to predict sloshing displacements and impulsive pressures in a liquid filled axisymmetric container subjected to only horizontal ground motion. The tank is assumed to be rigid and fixed at the base. The program requires the following three types of elements.

1. Two dimension 4-to-8 node axisymmetric elements idealizing the liquid.
2. Two node free surface elements.
3. Two node elements representing the liquid-solid interface.

Al.2 Program Capacity

The program uses a variable dimensioning in order to make an optimum use of high speed storage. Element group data is stored block wise on the disc. The program capacity can be varied through two Fortran statements in the main program.

\[
\text{COMMON (n)} \\
\text{MTOT = n}
\]

The total memory \( n \) required can be estimated by the following formula.

\[
n = M + 2 \times \text{NPTM} + 10 \times \text{NUMNP}
\]

in which

- \( M = \text{NEL1} \times (4 \times \text{MXNODS} - 2) \)
- \( \text{NEL1} = \text{Number of elements in group 1} \)
- \( \text{MXNODS} = \text{Maximum nodes in any element of group 1} \)
- \( \text{NPTM} = \text{Number of points of earthquake input} \)
- \( \text{NUMNP} = \text{Total number of node points} \)
### Al.3 Program Input Data

The following format should be followed for the necessary input data.

#### I. Problem Initiation and Title (A5, 3X, 18A4) - one card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>MODE</td>
<td>Punch the word 'START'</td>
</tr>
<tr>
<td>9 - 80</td>
<td>HED</td>
<td>Title of the problem</td>
</tr>
</tbody>
</table>

#### II. Master Control Card (4I5) - one card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>NUMNP</td>
<td>Total number of nodes</td>
</tr>
<tr>
<td>6 - 10</td>
<td>NEG</td>
<td>Number of element groups</td>
</tr>
<tr>
<td>11 - 15</td>
<td>NUMEST</td>
<td>Estimated number of storage locations required ($M_1$) for element group 1. Zero or blank: defaults to 3000</td>
</tr>
<tr>
<td>16 - 20</td>
<td>MODEX</td>
<td>Execution mode. Specify (a) Zero: data check only</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(b) 1: execution</td>
</tr>
</tbody>
</table>

#### III. Nodal Coordinates (I5, 5X, 2F 10.0, I5)

As many cards as needed to generate total number of nodes NUMNP and their coordinates

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>N</td>
<td>Node number. See Note 1</td>
</tr>
<tr>
<td>11 - 20</td>
<td>X(N)</td>
<td>X coordinate</td>
</tr>
<tr>
<td>21 - 30</td>
<td>Y(N)</td>
<td>Z coordinate</td>
</tr>
<tr>
<td>35 - 35</td>
<td>KN</td>
<td>Node number difference between successive generated nodes (given on first card in a sequence).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specify. Zero: No generation. See Note 2.</td>
</tr>
</tbody>
</table>

**Note:**

(1) Node cards may not be in numerical order. Eventually, however, all nodes must be identified.
(2) The mesh generation parameter KN must appear on the first card of a series of nodal points to be generated. The intermediate nodes to be generated between nodes (say N1 and N2) will be located at equal intervals along the straight line joining the two nodes. KN is the nodal increment to be added to previous node number. The node difference N2-N1 must be exactly divisible by KN.

IV. Solution Time and Step Size (2I5, 3F10.0, 3I5)

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>NDT</td>
<td>Number of solution time steps. Specify 0: defaults to 1 step</td>
</tr>
<tr>
<td>6 - 15</td>
<td>DT</td>
<td>Step size</td>
</tr>
<tr>
<td>16 - 20</td>
<td>NPRINT</td>
<td>Time interval for printout of nodal displacements and pressures expressed as a multiple of the integration time step. Specify 0: defaults to 1</td>
</tr>
</tbody>
</table>

V. Earthquake Input

A. Control Information (2I5) - one card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>NBCF</td>
<td>Number of ground input components (use 1)</td>
</tr>
<tr>
<td>6 - 10</td>
<td>NPTM</td>
<td>Maximum number of points to describe the earthquake input. See Note 1.</td>
</tr>
</tbody>
</table>

Note:

(1) NPTM is the number of \([f(t), t]\) pairs used to define the earthquake ground motion which could be either an acceleration or displacement-time history record. At least two points are required to describe the input.
B. Earthquake Input Data

For one component of ground motion (horizontal in this case) a control card followed by as many cards as needed to define the earthquake.

1. Control Card (2I5, F10.0, I5) - First card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>NC</td>
<td>Function number. Specify equal to 1 in this case</td>
</tr>
<tr>
<td>6 - 10</td>
<td>NPTS(NC)</td>
<td>Number of time points used to describe the earthquake input (GE.2 and EQ. to NPTM)</td>
</tr>
<tr>
<td>11 - 20</td>
<td>FOM</td>
<td>Multiplication factor for conversion to right units. See Note 1</td>
</tr>
<tr>
<td>21 - 25</td>
<td>INPUT</td>
<td>Specify</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 - If acceleration is ground input</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 - If displacement is the given ground input.</td>
</tr>
</tbody>
</table>

2. \([f(t), t]\) Earthquake Data (8F10.0)

As many cards as needed to define NPTS (NC) pairs of points \([TFN(\text{NC},\text{I}), FN(\text{NC},\text{I})])\; four pairs per card. See Note 2.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 10</td>
<td>TFN(\text{NC},\text{I})</td>
<td>Time at point 1 : (t_1)</td>
</tr>
<tr>
<td>11 - 20</td>
<td>FN(\text{NC},\text{I})</td>
<td>Acceleration or displacement value at point 1 : (f(t_1))</td>
</tr>
<tr>
<td>21 - 30</td>
<td>TFN(\text{NC},\text{2})</td>
<td>(t_2)</td>
</tr>
<tr>
<td>31 - 40</td>
<td>FN(\text{NC},\text{2})</td>
<td>(f(t_2))</td>
</tr>
<tr>
<td>41 - 50</td>
<td>TFN(\text{NC},\text{3})</td>
<td>(t_3)</td>
</tr>
<tr>
<td>51 - 60</td>
<td>FN(\text{NC},\text{3})</td>
<td>(f(t_3))</td>
</tr>
<tr>
<td>61 - 70</td>
<td>TFN(\text{NC},\text{4})</td>
<td>(t_4)</td>
</tr>
<tr>
<td>71 - 80</td>
<td>FN(\text{NC},\text{4})</td>
<td>(f(t_4))</td>
</tr>
<tr>
<td></td>
<td>Next card(s)</td>
<td>as many as needed to define the earthquake input.</td>
</tr>
</tbody>
</table>
Note:

1. Factor of multiplication if necessary to make the units of ground input compatible with the units of the tank dimensions. This option is available only if input is in the form of accelerogram. In case of displacement history the units must be in inches, seconds and pounds.

2. Time values at successive points are assumed to increase in magnitude. Values of ground input other than TPN(NC,I) are calculated within the program using a linear interpolation.

VI. Element Data

Elements are divided into three groups (NEG). An element group is a series of elements of a particular type.

The elements in a particular group must be numbered sequentially starting with the number of the first element as specified on the element group control card.

Following are the three types of element groups used in this program.

Type 1 - Two Dimensional Finite Elements

These are 4-to-8 node axisymmetric isoparametric elements which lie in the global X-Z plane and are used to model the liquid continuum. Z-axis has been taken as the axis of revolution for the axisymmetric tank.

Type 2 - Free Surface Elements

These are 2 node axisymmetric elements which have been used to represent the free surface of the liquid. These
elements lie in the X-Z global plane where Z is the axis of revolution of the tank. These elements contribute to mass matrix.

Type 3 - Liquid-Solid Interface Elements

These are two node axisymmetric elements and lie in the global X-Z plane. These elements contribute to the loading vector.

Type 1 - Two Dimensional Finite Elements

A. Control Information (615) - one card.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>NGR</td>
<td>Element group indicator. Punch the number &quot;1&quot;.</td>
</tr>
<tr>
<td>6 - 10</td>
<td>NEL1</td>
<td>Number of elements in group 1.</td>
</tr>
<tr>
<td>11 - 15</td>
<td>MFST</td>
<td>Element number of the first element in this. See Note 1.</td>
</tr>
<tr>
<td>16 - 20</td>
<td>ITYP2D</td>
<td>Element type code. Specify Zero: axisymmetric</td>
</tr>
<tr>
<td>21 - 25</td>
<td>MXNODS</td>
<td>Maximum number of nodes used to describe any one element. Specify Zero: defaults to 4. (GE.4 and LE.8)</td>
</tr>
<tr>
<td>26 - 30</td>
<td>NINT</td>
<td>Numerical integration order to be used in Gaussian quadrature. Specify Zero: defaults to 2 (GE.2 and LE.4) See Note 2.</td>
</tr>
</tbody>
</table>

Note:

(1) Element numbers in any group may not start from 1 if MFST is specified.

(2) For rectangular elements, an integration order of 2 is sufficient. For non rectangular elements a higher order should be used.
B. **Element Data** (11I5)

As many data cards as are needed in order to generate the element data for the elements (NELL) in this group.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1 - 5   | M        | Element number  
|         |          | See Note 1   |
| 6 - 10  | NOD(1)   | Global node number of element node 1. |
| 11 - 15 | NOD(2)   | Global node number of element node 2. |
| 16 - 20 | NOD(3)   | Global node number of element node 3. |
| 21 - 25 | NOD(4)   | Global node number of element node 4. |
| 26 - 30 | NOD(5)   | Global node number of element node 5. |
| 31 - 35 | NOD(6)   | Global node number of element node 6. |
| 36 - 40 | NOD(7)   | Global node number of element node 7. |
| 41 - 45 | NOD(8)   | Global node number of element node 8.  
|         | IEL      | Number of nodes in the element.  
|         |          | Zero: defaults to MXNODS  |
| 51 - 55 | KG       | Node number increment for element generation (given on 1st card in a sequence)  
|         |          | Zero: defaults to 1  
|         |          | See Note 3  |

**Note:**

1. Elements must be input in increasing sequence, with MFST being the 1st element. Cards for the first and last element must be included.

2. If an element has less than 8 nodes (i.e., IEL.LT.8), input a zero or blank corresponding to the missing node location. For example, for a 6 node element with nodes 6 and 8 missing, the element node number array would be NOD(I) = [X X X X 0 X 0] where X entries represent the global node numbers.
(3) The node generation parameter KG must appear on the first element card of a sequence and is used to determine the node numbers for a group of missing elements in that sequence. If M is the first element of the sequence and the elements [M+1, M+2, ..., M+J] are the missing J elements, then the node numbers of the successive J elements are automatically incremented by the value KG given for the element M. Only the nonzero node numbers appearing on the M-th element card are incremented in this automatic generation. In the printout of the element data, generated elements are marked with an asterisk.

Type 2 - Free Surface Boundary Elements

A. Control Information (4I5) - one card.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>NGR</td>
<td>Element group number. Punch the number '2'.</td>
</tr>
<tr>
<td>6 - 10</td>
<td>NEL2</td>
<td>Number of elements in group 2</td>
</tr>
<tr>
<td>11 - 15</td>
<td>MFST</td>
<td>Number of the first element in group 2 (need not start with 1)</td>
</tr>
<tr>
<td>16 - 20</td>
<td>ITYP</td>
<td>Element type. Specify Zero: axisymmetric this is the only option available.</td>
</tr>
</tbody>
</table>

B. Element Data (4I5)

As many cards as needed to generate NEL2 elements.
### Columns Variable Description

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1 - 5   | M        | Element number  
See Note (1) |
| 6 - 10  | NOD(1)   | Global node number of element node I  |
| 11 - 15 | NOD(2)   | Global node number of element node J  |
See Note (2) |

**Note:**

(1) All elements must be input in ascending numerical order, starting with element number MFST. Cards for the first and last element must be included.

(2) The node generation parameter KG must be given on the first element card prior to a group of missing elements. In the print out of the element data, generated elements are prefixed by an asterisk.

### Type 3 - Liquid-Solid Interface Elements

#### A. Control Information (4I5) - one card.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1 - 5   | NGR      | Element group number  
Punch the number '3' |
| 6 - 10  | NEL3     | Number of elements in group 3 |
| 11 - 15 | MFST     | Number of first element in group 3 |
| 16 - 20 | ITYP     | Element type : Specify  
Zero: axisymmetric (the only option available) |
B. Element Data (3I5, 5X, F10.0)

As many cards as the number of elements NEL3 in group 3

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 5</td>
<td>M</td>
<td>Element number</td>
</tr>
<tr>
<td>6 - 10</td>
<td>NOD(1)</td>
<td>Global node number of element node I</td>
</tr>
<tr>
<td>11 - 15</td>
<td>NOD(2)</td>
<td>Global node number of element node J</td>
</tr>
<tr>
<td>21 - 30</td>
<td>COSS</td>
<td>X-direction cosine of the outward normal to the element.</td>
</tr>
</tbody>
</table>

VII. New Problem Data

A new problem may now be solved by adding data starting with Section I. Any number of problems can be solved in one run.

VIII. Termination Card (A4) - one card

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 4</td>
<td>MODE</td>
<td>Punch the word 'STOP'.</td>
</tr>
</tbody>
</table>

AI.4 Output

Output includes the nodal displacements and impulsive pressures. Displacements are meaningful only for the free surface nodes.
C PROGRAM SLOSH2(INPUT,OUTPUT,PUNCH,TAPES=INPUT,TAPE6=OUTPUT,TAPE1,TAPE2,TAPE3=PUNCH)

SLOSH2---A FINITE ELEMENT PROGRAM TO DETERMINE THE SLOSHING RESPONSE UNDER EARTHQUAKE GROUND MOTIONS IN AN AXI-SYMMETRIC RIGID TANK
DEVELOPED BY-- MOHAMMAD ASLAM, DEPARTMENT OF CIVIL ENGINEERING, UNIVERSITY OF CALIFORNIA, BERKELEY AUGUST 1978

COMMON /CNTRL1/ NUMNP,NEG,MODEX,NPAR(18),NG,KBC
COMMON /CNTRL2/ KST,NCT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP
COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15
COMMON /ELSTOR/ NUMEST,MIKREST,MAXEST
COMMON /JUNK/ HED(18),MTOT,NLINE
COMMON /ABC/ NHBC,NBCF,NPTM
COMMON /WORK/ WORK(200)
COMMON /CONST/ A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,THETA,DELTA,ALPHA,PI
COMMON /A/ (10000)

MTOT = 10000
200 MAXEST = 0

INPUT PHASE

PROGRAM MASTER CONTROL DATA
CALL DOTT
CALL INPUT ELEMENT INFORMATION
CALL ELCLAL

SOLUTION PHASE

BLANK COMMON STORAGE ALLOCATION

ARRAY ---DESCRIPTION--- DIMENSION
N1 TFN TIME VALUES AT POINTS NPTM*BCF
N2 FN FUNCTION VALUES AT POINTS NPTM*BCF
N3 NPTS NUMBER OF FUNCTION INPUT POINTS NBCF
N4 TD FIRST DERIVATIVE OF VEL POT. NUMNP
N5 TDD 2ND DERIVATIVE OF VEL. POT. NUMNP
N6 TTRU NUMNP
N7 P PRESSURE(DYNAMIC IMPULSIVE) NUMNP
N8 T VELOCITY POTENTIAL NUMNP
N9 MAXA ADDRESSES OF XK DIAGONAL ELEMENTS NUMNP+1
N10 XK EFFECTIVE STIFFNESS MATRIX NU
N11 Q LOADING VECTOR NUMNP
N12 C MASS MATRIX NUMNP
N13 E WATER DISPLACEMENTS AT SURFACE NUMNP
CALL ADRSK (A(N11),A(N12),NUMNP,NUK,NB)

SHIFT STORAGE TO ELIMINATE MODAL COORDINATE DATA

5 I = 1 + MAXEST
N12M = N12 - 1
DO 10 J=N3,N12M
   A(I) = A(J)
10  I = I + 1

N1 = 1 + MAXEST
N2 = N1 + NUMNP
N3 = N2 + NUMNP
N4 = N3 + NUM
N5=N4 + NUMNP
N6=N5 + NUMNP
N7=N6 + NUMNP
N8=N7 + NUMNP
N9 = N8 + NUMNP
N10 = N9 + NUMNP + 1
N11 = N10 + NUK
N12 = N11 + NUMNP
N13 = N12 + NUMNP
N14 = N13 + NUMNP
N15 = N14 + NUMNP
IF(N15.GT.MTOT) CALL ERROR (N15-MTOT)

IF(MODEX.EQ.0) GO TO 200

INITIALIZE STIFFNESS MATRIX (XX) AND LOADING VECTOR Q

N12M = N12 - 1
DO 15 I=N10,N12M
   A(I) = 0.0
15   

INITIALIZE VELOCITY POTENTIAL VECTOR AT AT TT(0)=T(0)

DO 20 I=1,NUMNP
   IT = N8 + I - 1
   ITT = N14 + I - 1
20   A(ITT) = A(IT)

INITIALIZE THE TIME STEP COUNTER

KSTEP = 0
TIME=0.

INITIALIZE MASS MATRIX (LUMPED MASS SYSTEM)

N13M = N13 - 1
DO 25 I=N12,N13M
   A(I) = 0.0
25   

CALCULATE CONSTANTS OF INTEGRATION

PI=3.141592654
G=386.18
RO=0.00009351
THETA=1.0
DELTA=0.50
ALPHA=0.25
TAU=THETA*DT
A0=1.0/(ALPHA*TAU*TAU)
A1=DELTA/(ALPHA*TAU)
A2=1.0/(ALPHA*TAU)
A3=1.0/(2.*ALPHA)-1.
A4=DELTA/ALPHA-1.
A5=TAU*(DELTA/ALPHA-2.0)/2.
A6=DT*(1.-DELTA)
A7=DT*DELTA
A8=DT*DT*(.5.-ALPHA)
A9=ALPHA*DT*DT

ASSEMBLE THE EFFECTIVE SYSTEM STIFFNESS MATRIX (K*)

CALL ASSEMBLE

FORM THE EFFECTIVE K AND CALL IT XK

CALL KSTAR(A(N9),A(N10),A(N12))

INITIALIZE VELOCITY POTENTIAL AND ITS DERIVATIVES
N6M=N6-1 DO 36 I=N6+1 N6
A(I)=0.

TRIANGULARIZE THE EFFECTIVE CONDUCTIVITY MATRIX, (K*)

40 KTR = 0
CALL COLSOL (A(N10),A(N11),A(N9),NUMNP,MB,NUK,KTR)

---------------------
TIME MARCHING LOOP

INITIALIZE Q
N12M=N12-1 DO 44 I=N11,N12M
A(I)=0.

TX=THETA*DT

---------------------
KSTEP = KSTEP + 1
THH=TIME+TX
TIME = TIME + DT

FORM THE LOAD VECTOR

CALL FORMQDC(THH)

COMPUTE EFFECTIVE LOAD VECTOR

CALL OEFF(A(N11),A(N12),A(N8),A(N4),A(N5),NUMNP)

UPDATE (TT) VECTOR

DO 82 I=1,NUMP
ITT = N8 + I - 1
ITT = N14 + I - 1
82 A(ITT) = A(IT)

SOLVE THE EQUILIBRIUM EQUATIONS FOR VELOCITY POTENTIAL

84 KTR = 2
CALL COLSOL(A(N18),A(N11),A(N9),NUMNP,MB,NWK,KTR)

Q-VECTOR IS NOW T-VECTOR. SET T(I)=0(I) AND Q(I)=0.

DO 85 I=1,NUMNP
IT = N8 + I - 1
IQ = N11 + I - 1
A(IT) = A(IQ)
85 A(IQ) = 0.0

CALCULATE VEL. POTENTIAL AND ITS DERIVATIVE AT TIME+DT
CALL CALC(A(N8),A(N14),A(N4),A(N5),A(N7),A(N13),NUMNP)

PRINT AND/OR PUNCH THE NODAL DISPLACEMENTS AND PRESSURES.
IF REQUESTED, AT THIS TIME STEP
K = MOD(KSTEP,NPRINT)
IF(K,NE.0) GO TO 90
CALL OUT(A(N13),NUMNP,TIME,KSTEP)
CALL OUP(A(N7),NUMNP,TIME,KSTEP)
90 IF(KP.EQ.0) GO TO 92
L = MOD(KSTEP,KP)
IF(L,NE.0) GO TO 92
CALL PTEMP(A(N8),TIME,NUMNP)
92 CONTINUE
SD(I) = I3.
NN=KSTEP+1
CHECK FOR FINAL TIME STEP
IF(KSTEP.LT.NOT) GO TO 100
GO TO 200
END

SUBROUTINE DOTI
COMMON /CNTRL/ NUMNP,NEG,MODEX,NPAR(10),NG,KBC
COMMON /CNTRL2/ KST,NST,DT,TSTART,TAMP,NPRINT,NSTREF,TME,KP
COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15
COMMON /ELSTOR/ NUMEST,MIDEST,MAKEST
COMMON /JUNK/ HED(18),HTOT,HTLINE
COMMON /NBC/ NNBC,NBCF,NPTM
COMMON N(I)
DIMENSION MOD(2)
DATA MOD/SHSTART,SHSTOP /

========================== READ CONTROL INFORMATION ===========================
10 READ (5,1000) MOD,HED
IF(MODE.EQ.MOD(2)) STOP
IF(MODE.EQ.MOD(1)) GO TO 20
WRITE(6,3000)
GO TO 10
20 READ (5,1001) NUMNP,NEG,NUMEST,MODEX
IF(NUMEST.EQ.0) NUMEST = 4000
IF(NUMIP.GT.0) GO TO 30
WRITE(6,3061)
STOP
30 IF(NEG.GT.0) GO TO 40

DOTI 1
DOTI 2
DOTI 3
DOTI 4
DOTI 5
DOTI 6
DOTI 7
DOTI 8
DOTI 9
DOTI 10
DOTI 11
DOTI 12
DOTI 13
DOTI 14
DOTI 15
DOTI 16
DOTI 17
DOTI 18
DOTI 19
DOTI 20
DOTI 21
DOTI 22
DOTI 23
DOTI 24
DOTI 25
DOTI 26
DOTI 27
DOTI 28
DOTI 29
CALL TITLE (HED)
WRITE(6,2000) NUMNP,NME,NUMEST,NODEX
NLINE = 17

***********************************************************

BLANK COMMON STORAGE ALLOCATION

ARRAY ------ DESCRIPTION ------- DIMENSION
N1 NODAL X-COORDINATES NUMNP
N2 NODAL Y-COORDINATES NUMNP
N3 TFN TIME VALUES AT POINTS NPTM*NBCF
N4 FN FUNCTION VALUES AT POINTS NPTM*NBCF
N5 NPTS NUMBER OF FUNCTION INPUT POINTS NBCF
N6 TD FIRST DERIVATIVE OF VEL POT. NUMNP
N7 TDD 2ND DERIVATIVE OF VEL. POT. NUMNP
N8 TAU NUMNP
N9 P PRESSURE (DYNAMIC IMPULSIVE) NUMNP
N10 T VELOCITY POTENTIAL NUMNP
N11 MXA ADDRESSES OF (XX) DIAGONAL ELTS. NUMNP+1
N12 MH ACTIVE COLUMN HEIGHTS NUMNP

***********************************************************

READ NODAL POINT COORDINATE DATA

***********************************************************

N1 = 1 + NUMEST
N2 = N1 + NUMNP
N3 = N2 + NUMNP
IF(N3.GT.MTOT) CALL ERROR(N3-MTOT)

CALL COORD (A(N1),A(N2),NUMNP)

***********************************************************

READ SOLUTION TIME AND GROUND ACCELERATION

***********************************************************

READ(5,1002)NDT,DT,NPRINT,KP
KST=0 & TSTART=0.
TAMB=0. & NTSREF=0.

IF(NDT.EQ.0) NDT = 1
IF(NPRINT.EQ.0) NPRINT = 1
CALL TITLE (HED)
WRITE(6,2001) NDT,DT,NPRINT,KP
NLINE = 22

***********************************************************

READ THE EARTHQUAKE ACCELEROGRAMS

***********************************************************

READ (5,1007) NBCF,NPTM,NNBC
WRITE(6,2002) NBCF,NPTM
NLINE = NLINE + 9
50 N4 = N3 + NPTM*NBCF
N5 = N4 + NPTM*NBCF
N6 = N5 + NBCF
IF(N6.GT.MTOT) CALL ERROR (H6-MTOT)

CALL FUNC (A(N3),A(N4),A(N5),NPTM)

60 CALL TITLE (HED)
NLINE = 10
70 N7=N6+NUMNP
N8=N7+NUMNP
N9=N8+NUMNP
N10=N9+NUMNP
IF(N10.GT.MTOT) CALL ERROR (N10-MTOT)

C
C INITIATE INITIAL VEL. POTENTIAL
C
80 T1 = T10 + NUMNP
IF(N11.GT.MTOT) CALL ERROR (N11-MTOT)

CALL IN ITAL (A) • TAMS. Ht.:~1tIP)

FORMAT STATEMENTS

1000 FORMAT(55.3X,18A4)
1001 FORMAT(15)
1002 FORMAT(15,F10.0,215)
1003 FORMAT(315)
2000 FORMAT(25(IH*)/28H CONTROL INFORMATION/20(IH*)//
1 34H NUMBER OF NODAL POINTS ...... = 15/
2 34H NUMBER OF ELEMENT GROUPS ...... = 15/
3 34H MAX. ELEMENT GROUP STORAGE : = 15/
4 34H SOLUTION MODE ............... = 15/
4 22H EQ. 0. DATA CHECK/EX.16HEO. 1. EXECUTION//)
2001 FORMAT(50(IH*)/38H SOLUTION TIME AND PRINT. PUNCH ,
1 12H INFORMATION/50(IH*)//
5 48H NUMBER OF SOLUTION TIME STEPS .......... = 15/
6 48H SOLUTION TIME STEP INCREMENT ........ = F10.4/**
9 48H OUTPUT PRINT INTERVAL ................. = F10.4/**
8 48H OUTPUT PUNCH INTERVAL ............... = F10.4/**
2002 FORMAT(25(IH*)/25H TIME DEPENDENT FUNCTIONS/25(IH*)//
1 48H NUMBER OF TIME DEPENDENT FUNCTIONS ...... = 15/
2 48H MAXIMUM NUMBER OF (F(T),T) POINTS ...... = 15/
3000 FORMAT/**ERROR** PROBLEM DECN. MUST BEGIN WITH START CARD)
3001 FORMAT/**ERROR** NO. OF NODAL POINTS MUST BE .GT. ZERO)
3002 FORMAT/**ERROR** NO. OF ELEMENT GROUPS MUST BE .GT. ZERO)

RETURN

END

SUBROUTINE COORD (X,Y,NUMNP)

****:***:************:************:************:************:************

RETURN

END

DIMENSION X(1,Y(1)

COMMON /JUNK / HED(18),MTOT,NLINE

READ OF GENERATE NODAL POINT DATA

WRITE(6,2300)
WRITE(6,2301)
NLINE = NLINE + 12

COOR 7
COOR 8
COOR 9
COOR 10
COOR 11
COOR 12
COOR 13
COOR 14
COOR 15
HOLD = 0

10 READ (5,1000) N,X(N),Y(N),KH,JPR
IF(N.EQ.1) IPR=JPR
IF(NLINE.LT.55) GO TO 15
CALL TITLE (HED)
WRITE(6,2001)
NLINE = 10

15 WRITE(6,2002) N,X(N),Y(N),KH
NLINE = NLINE + 1
IF(NOLD.EQ.0) GO TO 30

CHECK IF GENERATION IS REQUIRED
IF(KHOLD.EQ.0) GO TO 30
NUM = (N-NOLD)/KNOLD
NUM = NUM - 1
DX = (X(N)-X(NOLD))/NUM
DY = (Y(N)-Y(NOLD))/NUM
K = NOLD
DO 20 J=1,NUM
KK = K
K = K + KNOLD
X(K) = X(KK) + DX
20 Y(K) = Y(KK) + DY

30 NOLD = N
KNOLD = KN
IF(N.NE.NUMP) GO TO 10

IF(IPR.EQ.1) GO TO 200

PRINT ALL NODAL POINT DATA
CALL TITLE (HED)
WRITE(6,2003)
NLINE = 9
NROW = NUMP/3 + 1
NR = 0

DO 100 I=1,NUMP,3
NR = NR + 1
IP = I + 2
IF(NR.EQ.NROW) IP = NUMP
IF(NLINE.LT.55) GO TO 50
CALL TITLE (HED)
WRITE(6,2003)
NLINE = 9

50 WRITE(6,2004) (N,X(N),Y(N),N*1,IP)
100 NLINE = NLINE + 1

FORMAT STATEMENTS
1000 FORMAT(15.5,2F10.8,15.11)
2000 FORMAT(28(1H4/28H NODAL POINT COORDINATE DATA/28(1H*))/
2001 FORMAT(19(1H*)/19H A. INPUT NODE DATA/19(1H*)/
1 4X,4HNODE,5X,7HX-COORD,5X,7HY-COORD,5X,4HDIFF/)
2002 FORMAT(3X,15.2F12.3,3X,15)
2003 FORMAT(23(1H*)/23H B. GENERATED NODE DATA/23(1H*)/
1 3(4X,4HNODE,5X,7HX-COORD,5X,7HY-COORD,5X)/
2004 FORMAT(3(3X,15.2F12.3,5X))
C 200 RETURN
END
SUBROUTINE FUNC (TFN,FN,NPTS,NPTM1)
C***************************************************************
C DEFINE ALL BOUNDARY CONDITION FUNCTIONS
C***************************************************************
DIMENSION TFN(NPTM1,1),FN(NPTM1,1),NPTS(1)
COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,TSTRF,TIME,KP
COMMON /JUNK / HED(18),MTOT,NLINE
COMMON /NBC / NNBC,NBCF,NPTM
COMMON /WORK / FORM(4),WORK(196)
WRITE(6,2001)
NLINE = NLINE + 3
DO 100 LL=1,NBCF
READ (5,1000) NC,NPTS(NC),FOM,INPUT
WRITE(6,2002) NC,NPTS(NC),FOM
NLINE = NLINE + 1
IF(NPTS(NC).GE.2.AND.NPTS(NC).LE.NPTM) GO TO 20
WRITE(6,3000)
STOP
C READ TIME FUNCTION VERSUS TIME TABLE
20 NT = NPTS(NC)
READ (5,1001) (TFN(K,NC),FN(K,NC),K=1,NT)
C CHECK THAT TIME POINTS ARE IN INCREASING ORDER
TOLD = -1.
DO 30 K=1,NT
IF(TFN(K,NC).GT.TOLD) GO TO 30
WRITE(6,3001)
STOP
30 TOLD = TFN(K,NC)
C 50 DO 70 K=1,NT
IF(NLINF.LT.55) GO TO 60
CALL TITLE (HED)
WRITE(6,2000)
WRITE(6,2001)
NLINE = 10
60 WRITE(6,2003) K,TFN(K,NC),FN(K,NC)
70 NLINE = NLINE + 1
C INTEGRATE THE ACCELEROGRAM TO OBTAIN THE VELOCITY
FI=FN(1,NC)
FN(1,NC)=0.
IF(INPUT.EQ.2) GO TO 55
DO 80 K=2,NT
L=K-1
FI1=FI(K,NC)
FN(K,NC)=FN(L,NC)+(FI+FI1)*(TFN(K,NC)-TFN(L,NC))*FOM/2.
FI=FI1
WRITE(6,01)K,TFN(K,NC),FN(K,NC)
81 FORMAT(60X,15.2F15.5)
CONTINUE
GO TO 100

DIFFERENTIATE DISPLACEMENT TO GET VELOCITY

DO 110 I=2,NT
J=I-1
IF(I.EQ.NT) GO TO 111
K=I+1
A=TFN(I,NC)-TFN(J,NC)
B=TFN(K,NC)-TFN(I,NC)
C=FNK(I,NC)
D=(C-F1)/A
E=(FNK(K,NC)-C)/B
WRITE(6,81)I,TFN(I,NC),FNK(I,NC)
CONTINUE
110 CONTINUE

FNK(I,NC)=(FNK(I,NC)-C)/(TFN(K,NC)-TFN(J,NC))
WRITE(6,81)K,TFN(K,NC),FNK(K,NC)

CONTINUE

FORMT STATMENTS

1000 FORMAT(2I5,F10.15)
1001 FORMAT(8F10.0)
2000 FORMAT(25(I1*)/25H TIME DEPENDENT FUNCTIONS/25(I1*)/)
2001 FORMAT(4X,8HFUNCTION,4X,9HNUMBER OF,6X,18HTIME POINT,4X,4HTIME, 1 4X,8HFUNCTION/5X,SHNUMBER,4X,11HTIME POINTS,7X,SHNUMBER,
2 5X,SHVALUE,5X,SHVALUE/)
2002 FORMAT(4X,15,7X,15,40X,**MULTIPLICATION FACTOR=*,F10.4)
2003 FORMAT(3X,16,F12.3,F12.6)
3000 FORMAT(/49H **ERROR** (NPTS) MUST BE .GE. 2 AND .LE. (NPTM)
3001 FORMAT(/52H **ERROR** BC FUNCTION TIME POINTS ARE OUT OF ORDER)

RETURN
END

SUBROUTINE INITIAL (T.TAMB.NUMHP)

DIMENSION T(NUMHP)

ICON=0
TAMB=0.
DO 100 I=1,NUMHP
100 T(I)=TAMB
END

SUBROUTINE ECAL

COMMON /CTRL1/ NUMHP,NEG,MODEX,NPAR(10),NC,KBC
COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15, N16
COMMON /ELSTOR/ NUMEST,MIDEST,HAXEST
COMMON /JUNK/ HEF(10),MTOT,FLH
COMMON /WORK/ NST(10),WORK(130)
COMMON R(1)
DIMENSION LABEL(2,2)
DATA LABEL/*HAXISYM,CHMETRIC,SHP L A ,6HN A R */
A2-11

************ THIS ROUTINE CALLS THE APPROPRIATE ELEMENT ROUTINES FOR READING, GENERATING AND STORING THE ELEMENT DATA ************

TAPE ALLOCA TION

TAPE 1 - STORES ELEMENT GROUP DATA

************ TWO DIMENSIONAL FINITE ELEMENTS ************

NPAR(1) = 1
NPAR(2) = NUMBER OF TWO DIMENSIONAL ELEMENTS (NEL1)
NPAR(3) = NUMBER OF FIRST ELEMENT IN THIS GROUP (MFST)
NPAR(4) = ELEMENT TYPE CODE (ITYP2D)
EQ.0. AXISYMMETRIC
NPAR(5) = MAXIMUM NUMBER OF NODES (MNODS)
NPAR(6) = NUMERICAL INTEGRATION ORDER (NINT)

************ FREE SURFACE ELEMENTS ************

NPAR(1) = 2
NPAR(2) = NUMBER OF FREE SURFACE ELEMENTS (NEL2)
NPAR(3) = NUMBER OF FIRST ELEMENT IN THIS GROUP (MFST)
NPAR(4) = ELEMENT TYPE CODE (ITYP)
EQ.0. AXISYMMETRIC FREE SURFACE ELEMENT

************ SOLID BOUNDARY ELEMENTS ************

NPAR(1) = 3
NPAR(2) = NUMBER OF SOLID BOUNDARY ELEMENTS (NEL3)
NPAR(3) = NUMBER OF FIRST ELEMENT IN THIS GROUP (MFST)
NPAR(4) = ELEMENT TYPE CODE (ITYP)
EQ.0. AXISYMMETRIC SOLID-LIQUID BOUNDARY ELEMENTS

************ ZERO ACTIVE COLUMN HEIGHT ARRAY (MHT) ************

N12 = N11 + NUMNP + 1
N13 = N12 + NUMNP
IF(N13.GT.MTOT) CALL ERROR (N13-MTOT)

DO 5 I=N12,N13
5 A(I) = 0.0

REWIND 1

LOOP OVER ALL ELEMENT GROUPS

DO 100 NG=1,NEG
CALL TITLE (HED)
WRITE(6,2000) NG
NLINE = 7

READ (5,1000) NPAR
NGR = NPAR(1)
GO TO (1,2,3) NGR

-----------
ELEMENT GROUP 1
-----------

1 IF(NPAR(2).GT.0) GO TO 10
WRITE(6,3000)
STOP

10 IF(NPAR(6).LE.4) GO TO 20
WRITE(6,3001)
STOP

20 IF(NPAR(3).EQ.0) NPAR(3) = 1
IF(NPAR(5).EQ.0) NPAR(5) = 4
IF(NPAR(6).EQ.0) NPAR(6) = 2
IF(NPAR(7).EQ.0) NPAR(7) = 1
IF(NPAR(8).EQ.0) NPAR(8) = 1
IT = NPAR(4) + 1

WRITE(6,2001) NGR, (LABEL(I,IT),I=1,2),NPAR(2),NPAR(3),NPAR(5)
CALL ADRS1
GO TO 50

C ELEMENT GROUP 2

C 2 IF(NPAR(2).GT.0) GO TO 30
WRITE(6,3000)
STOP

30 IF(NPAR(5).EQ.0) NPAR(5) = 1
IT = NPAR(4) + 1

WRITE(6,2002) NGR, (LABEL(I,IT),I=1,2),NPAR(2),NPAR(3)
CALL ADRS2
GO TO 50

C ELEMENT GROUP 3

C 3 IF(NPAR(2).GT.0) GO TO 40
WRITE(6,3000)
STOP

40 IF(NPAR(3).EQ.0) NPAR(3) = 1
IT = NPAR(4) + 1
WRITE(6,2002) NGR, (LABEL(I,IT),I=1,2),NPAR(2),NPAR(3),NPAR(5)
CALL ADRS3

C 50 IF(MIDEST.GT.MAXEST) MAXEST = MIDEST
IF(MIDEST.LE.NUMEST) GO TO 60
GO TO 100

C STORE ALL ELEMENT GROUP INFORMATION ONTO TAPE 1

C 60 WRITE(I) MIDEST,NPAR,MAST.(A(I),I=1,MIDEST)

C 100 CONTINUE

C IF(MAXEST.LE.NUMEST) GO TO 300
WRITE(6,3002) MAXEST
STOP

C FORMAT STATEMENTS

C 1000 FORMAT(10I5)
2000 FORMAT(23(I1*),/20H ELEMENT DATA, GROUP,13/23(I1*),/)
ECL 141
2001 FORMAT(26H ELEMENT GROUP INDICATOR =13,18H (TWO DIMENSIONAL,2A6)
A2-13

1 10H ELEMENTS //
2 30H NUMBER OF ELEMENTS .......... = 15//
3 30H NO. OF FIRST ELEMENT IN GROUP .. = 15/
4 30H MAX. NO. OF NODES PER ELEMENT .. = 15/
5 30H NUMERICAL INTEGRATION ORDER .... = 15/

2002 FORMAT(26H ELEMENT GROUP INDICATOR =13.,
1 2H (.2A6,36H FREE SURFACE ELEMENTS
2 30H NUMBER OF ELEMENTS .......... = 15/
3 30H NU. OF FIRST ELEMENT IN GROUP .. = 15/

3000 FORMAT(/51H **ERROR** NO. OF ELTS. IN GROUP MUST BE .GT. ZERO),
3001 FORMAT(/53H **ERROR** NO. OF INTEGRATION PTS. MUST BE .LE. FOUR),
3002 FORMAT(/41H **ERROR** (NUMEST) MUST BE INCREASED TO 75).

C 300 RETURN

END

SUBROUTINE ADS1

COMMON /CNTRL1/ NUMP,NEG,NODEX,NPAR(10),NGB,KBC
COMMON /DIM / N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15
COMMON /ELSTOR/ NUMEST,MIDEST,MIDEST
COMMON /WORK / M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(198)
COMMON A(4)

C ********** BLANK COMMON STORAGE ALLOCATION **********

C ARRAY ----------------- DESCRIPTION ----------------- DIMENSION
M1 LM ELEMENT CONNECTIVITY ARRAY Noded Nod1 Nod2
M2 XX ELEMENT COORDINATES 2*Noded Nod1 Nod2
M3 IELT NO. OF NODES DESCRIBING ELEMENT Nod1
M4 NODS MIDSIDE NODES LOCATION ARRAY Noded Nod1

C ********** COMMON STORAGE ALLOCATION **********

Nod1 = NPAR(2)
Noded = NPAR(5)
Nod2 = 2*Noded
Noded = 10*Noded-4

M1 = 1
M2 = M1 + M2*Nod1
M3 = M2 + Nod2*Nod1
M4 = M3 + Nod1
M5 = M4 + Nod1

NLAST=15-1

C WRITE(6,2000) NLAST
MIDEST = NLAST
IF(NDSDIM.EQ.0) NDSDIM = 1

C CALL ELGR1 (A(N1),A(N2),A(N12),A(M1),A(M2),A(M3),A(M4),I1,NODS,NMD
INDSDIM)

C 2000 FORMAT(38H LENGTH OF ELEMENT INFORMATION .. = 15///

C RETURN

SUBROUTINE ELGR1 (X,Y,MT,LM,XY,IELT,NODS,M1,NODS,NMD,NDS,NDSDIM)

C ********** INPUT INFORMATION FOR 4- TO 8-NODE ISOPARAMETRIC ELEMENTS **********

C
C DIMENSION X(1), Y(1), MHT(1), LM(MXNODS, 1), XY(NDM, 1), IELT(1),
1 NODS(NDSDIM, 1)
C COMMON /CTRNL/, NUMNP, NEG, MODEX, NPAR(10), NG, KBC
C COMMON /JUNK / HED(1), MTOT, MLINE
C COMMON /WORK / DUM(10), NOD(8), NODM(8), NODSM(8), WORK(166)
C DIMENSION AST(2)
DATA AST/2H, 2H/*
C NEL = NPAR(2)
MST = NPAR(3)
C
..............................
C CALL TITLE (HED)
C
C READ AND GENERATE ELEMENT INFORMATION
C
C
C CALL TITLE (HED)
WRITE(6, 2003) NG
WRITE(6, 2004) (1.1-1.8)
NLINE = 10
N = 1
IMEM = NME + NEL1 - 1
C 100 READ (5, 1002) M, NOD, IEL, KG
C IF(MTYP.EQ.0) MTYP = 1
IF(IEL.EQ.0) IEL = MXNODS
IF(KG.EQ.0) KG = 1
IF(MXNODS.GE.IEL) GO TO 110
WRITE(6, 3002) M
STOP
C 110 IF(M-IMEM) 200, 120, 200
C SAVE ELEMENT INFORMATION FOR GENERATION OF ADDITIONAL ELEMENTS
C
C 120 DO 130 I = 1, 8
130 NODM(I) = NOD(I)
IF(IEL.EQ.4) GO TO 150
II = 0
DO 140 I = 5, 8
NN = NOD(I)
IF(NN.EQ.0) GO TO 140
II = II + 1
NODSM(II) = I
140 CONTINUE
C 150 IELM = IEL
KKK = KG
ASTT = AST(1)
C STORE PERMANENT ELEMENT INFORMATION
C
C 200 12 = 0
DO 230 I = 1, IELM
IF(I.EQ.4) GO TO 210
JJ = NODSM(I-4)
C
II = NODM(JJ)
GO TO 220
210 II = NODM(I)
220 LM(I,N) = II
12 = 12 + 2
XY(12-I.N) = X(II)
230 XY(12..N) = Y(II)
C
IELT(N) = IELM
IF(IELM.EQ.4) GO TO 250
NH = IELM - 4
DO 240 I=1,NN
240 NOD5(I,N) = NOD5M(I)
C
UPDATE COLUMN HEIGHTS AND BANDWIDTH
C
250 CALL COLHT (MHT, IELM, LM(I,N))
C
IF(NLINE.LT.55) GO TO 260
CALL TITLE (HMED)
WRITE(6,2003) NG
WRITE(6,2004) (1,1=1,8)
NLINE = 10
C
260 WRITE(6,2005) ASTT, IMEM, NODM, IELM
NLINE = NLINE + 1
IF(IEM.NEQ.NLAST) GO TO 300
C
N = N + 1
IMEM = IMEM + 1
C
CHECK IF ELEMENT DATA IS TO BE STORED FOR CURRENT ELEMENT
C
IF(IEM.EQ.M) GO TO 120
C
GENERATE NODE NUMBERS FOR NEXT ELEMENT
C
DO 270 I=1,8
IF(NODM(I).EQ.0) GO TO 270
NODM(I) = NODM(I)+KKK
270 CONTINUE
C
CHECK IF NEXT ELEMENT CARD IS TO BE READ
C
ASTT = ASTT(2)
IF(IEM.GT.M) GO TO 100
C
GENERATE INFORMATION FOR NEXT ELEMENT
C
GO TO 200
C
280 WRITE(6,3003) M
STOP
C
FORMAT STATEMENTS
C
1002 FORMAT(1115)
2003 FORMAT(30(1H*)/27H ELEMENT INFORMATION. GROUP.(13/30(1H*))/) ELGI 98
2004 FORMAT(4X,A68L,3X,10(1H*).12HNODE NUMBERS,10(1H*).3X.5H
6HNO. OF/SX.3HNO..3X.8(3X,II).4X.3HNO..5X.9HNOES/)
2005 FORMAT(4X,4X.814.10X.15)
3002 FORMAT(//,10H **ELEMENT,15.34H EXCEEDS MAXIMUM NUMBER OF NODES**) ELGI 174
3003 FORMAT(///,26H **ERROR**. ELEMENT CARD #15.16H OUT OF SEQUENCE) ELGI 175
SUBROUTINE ADRS2

COMMON /CNTRL/ NUMNP, NEG, MODEX, NPAR(10), NG, KBC
COMMON /DIM/ N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, N11, N12, N13, N14, N15
COMMON /ELSTOR/ NUMEST, MIDEST, MAXEST
COMMON /ELGR/ N1, M2, M3, M4, M5, M6, M7, M8, M9, M10, WORK(103)
COMMON A(1)

*********************************************************************

BLANK COMMON STORAGE ALLOCATION

******************************************************************************

DIMENSION M1, LM ELEMENT CONNECTIVITY ARRAY 2*NEL2
DIMENSION M2, XX ELEMENT X-COORDINATES 2*NEL2
DIMENSION M3, CL ELEMENT LENGTHS NEL2

***************************************************************

NEL2 = NPAR(2)

M1 = 1
M2 = M1 + 2*NEL2
M3 = M2 + 2*NEL2
M4 = M3 + NEL2
NLAST = M4 - 1

WRITE(6,2000) NLAST
MIDEST = NLAST

CALL ELGR2(A(N1), A(N2), ACM1, A(M2), A(M3))

2000 FORMAT(38H LENGTH OF ELEMENT INFORMATION ... = 15//)

RETURN
END

SUBROUTINE ELGR2(X,Y,LM,XX,CL)

***************************************************************

INPUT INFORMATION FOR FREE SURFACE ELEMENTS

***************************************************************

DIMENSION X(1), Y(1), LM(2,1), XX(2,1), CL(1)

COMMON /CNTRL/ NUMNP, NEG, MODEX, NPAR(10), NG, KBC
COMMON /JUNK/ HED(18), NTOT, NLINE
COMMON /NMC/ NNBC, NBCF, NPTH
COMMON /WORK/ DUM(10), MOD(2), NODM(2), WORK(186)
DIMENSION AST(2)
DATA AST/2H, 2H /

NEL2 = NPAR(2)
MFST = NPAR(3)
KBC = 0

READ AND GENERATE ELEMENT INFORMATION

******************************************************************************
C
N = 1
IMEM = MFST
NLAST = MFST + NEL2 - 1
CALL TITLE (HED)
WRITE (6,2003) NG
WRITE (6,2004)
NLINE = 10

C 100 READ (5,1003) M,NOD,KG
C IF(KG.EQ.0) KG = 1
II = NOD(1)
JJ = NOD(2)
C IF(M-IMEM) 280,120,280
C SAVE ELEMENT INFORMATION FOR GENERATION OF ADDITIONAL ELEMENTS
C 120 NODM(1) = II
NODM(2) = JJ
KKK = KG
ASTT = AST(1)
XL = SORT((X(JJ)-X(II))**2 + (Y(JJ)-Y(II))**2)
C STORE PERMANENT ELEMENT INFORMATION
C 200 DO 230 I=1,2
IJ = NODM(I)
LM(I,N) = IJ
230 XXX(I,N) = X(IJ)
C CL(N) = XL
C IF(NLINE.LT.55) GO TO 250
CALL TITLE (HED)
WRITE (6,2003) NG
WRITE (6,2004)
NLINE = 10
C 250 WRITE (6,2005) ASTT,IMEM,NODM
NLINE = NLINE + 1
IF(IMEM.EQ.NLAST) GO TO 300
C N = N + 1
IMEM = IMEM + 1
C CHECK IF ELEMENT DATA IS TO BE STORED FOR CURRENT ELEMENT
C IF(IMEM.EQ.N) GO TO 120
C GENERATE NODE NUMBERS FOR NEXT ELEMENT
C DO 270 I=1,2
270 NODM(I) = NODM(I) + KKK
C CHECK IF NEXT ELEMENT CARD IS TO BE READ
C ASTT = AST(2)
IF(IMEM.GT.N) GO TO 100
C GENERATE INFORMATION FOR NEXT ELEMENT
GO TO 200

200 WRITE(6,3002) M
STOP

FORMAT STATEMENTS

1003 FORMAT(4I5)
2003 FORMAT(30(IH*))/27H ELEMENT INFORMATION. GROUP.13/30(IH*)//
2004 FORMAT(4X.4HEL,T.4X.6HI-NODE,4X.6HJ-NODE,4X.5HX.15///)
1 3HNO. 25X.3HNO.//
2005 FORMAT(A2.15.5X.I5.5X.15)
30132 FORMAT(26H **ERROR** ELEMENT CARD OUT OF SEQUENCE)

COMMON /CTRL1/ NUMIP,NEG,MODEX,NPAR(10),NG,KBC
COMMON /ELSTOR/ NUMEST,MIHEST,MAHEST
COMMON /DIM / N1,N2,N3,M4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,M5AD5
COMMON /WORK / M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)
COMMON A(1)

********************************************************************
BLANK COMMON STORAGE ALLOCATION
********************************************************************

M1 ELEMENT LOCATION ARRAY NEL3
M2 X-COORDINATES 2*NEL3
M3 ELEMENT LENGTHS NEL3
M4 SINE OF ANGLE SI NEL3
M5 COSINE OF ANGLE SI NEL3

NEXP = NPAR(2)
M1 = 1
M2=M1+2*NEL3
M3=M2+2*NEL3
M4=M3+NEL3
M5=M4+NEL3
M6=M5+NEL3
M7=M6+1
WRITE(6,2000) NLAST
WRITE(6,2000) NLAST
CALL ELEGR3(A(N1),A(N2),A(M1),A(M2),A(M3),A(M4),A(M5))

2000 FORMAT(38H LENGTH OF ELEMENT INFORMATION .. = I5//)
RETURN

SUBROUTINE ELEGR3(X,Y,LM,XX,CL,SINS,COS)

DIMENSION X(1),Y(1),LM(2),XX(2),CL(1),SINS(1),COS(1)

COMMON /CTRL1/ NUMIP,NEG,MODEX,NPAR(10),NG,KBC
COMMON /JUNK / MEDI(18),HTOT,NLIE
COMMON /NDBC / NNDBC,NBCF,NPTM
COMMON / WORK / DUM(18), NOD(2), NODM(2), WORK(186)
DIMENSION AST(2), DATA AST/2H, 2H */

 ME3-NPAR(2)
 MFS = NPAR(3)

 ----------------------------------------
 KBC = 0
 ----------------------------------------
 READ AND GENERATE ELEMENT INFORMATION
 ----------------------------------------

 N = 1
 IMEM = MFS
 NLAST = MFS + ME3 -1
 CALL TITLE (HED)
 WRITE(6,2003) NG
 WRITE(6,2004)
 NLINE = 10

 100 READ(5,1003) H, NOD, KG, CS, SS

 IF(KG, LE, 0.0) KG = 1
 II = NOD(1)
 JJ = NOD(2)

 IF(M-IMEM) 280, 120, 200

 SAVE ELEMENT INFORMATION FOR GENERATION OF ADDITIONAL ELEMENTS

 120 NODM(1) = II
 NODM(2) = JJ
 YD = ABS(Y(II) - Y(JJ))
 IF(YD, GT, 0.000000001) GO TO 5
 SI = 9999999999.9
 GO TO 6

 5 SI = (X(II) - X(JJ)) / (Y(II) - Y(JJ))

 6 KKK = KG
 ASTT = AST(1)
 XL = SQRT((X(JJ) - X(II))**2 + (Y(JJ) - Y(II))**2)

 STORE PERMANENT ELEMENT INFORMATION

 200 DO 230 I = 1, 2
 JJ = NODM(I)
 LMM(I, N) = JJ

 230 XX(I, N) = X(IJ)

 CL(N) = XL
 SI = ATAN(SI)
 SINS(N) = SIN(SI)
 COSN(N) = COS(SI)
 CS(N) = CS
 SINS(N) = SS

 IF(NLINE, LT, 55) GO TO 250
 CALL TITLE (HED)
 WRITE(6,2003) NG
 WRITE(6,2004)
 NLINE = 10

 250 WRITE(6,2005), ASTT, IMEM, NODM
C
LINE = NLINE + 1
IF(INEM.EQ.NLAST) GO TO 300
N = N + 1
INEM = INEM + 1
C
CHECK IF ELEMENT DATA IS TO BE STORED FOR CURRENT ELEMENT
IF(INEM.EQ.M) GO TO 120
C
GENERATE NODE NUMBERS FOR NEXT ELEMENT
DO 270 I=1,2
270 NODM(I) = NDM(I) + KKK
C
CHECK IF NEXT ELEMENT CARD IS TO BE READ
ASTT = AST(2)
IF(INEM.GT.M) GO TO 100
C
GENERATE INFORMATION FOR NEXT ELEMENT
GO TO 200
C
200 WRITE(6,3002) M
STOP
C
FORMAT STATEMENTS
C
1003 FORMAT(415.2F10.0)
2003 FORMAT(30(IH*),/27H ELEMENT INFORMATION. GROUP,13/30(IH*)//)
2004 FORMAT(4X,5HELT.,4X,SHI-NOE.,4X,6HJ-NOE.,4X,SDIREC/5X,
1 3HNO.,25X,6HCOSINE//)
2005 FORMAT(A2,15.5X,15.5X,15)
3002 FORMAT(/26H **ERROR** ELEMENT CARD =15,16H OUT OF SEQUENCE)
C
300 RETURN
END
SUBROUTINE COLHT (MHT,ND,LM)
DIMENSION LM(1),MHT(1)
C
FIND SMALLEST GLOBAL NODE NUMBER (LS) FOR ELEMENT
LS=100000
DO 100 I=1,ND
100 IF (LM(I)) BGT 80.100.80
80 IF (LM(I) - LS) BGT 90.100.100
90 LS=LM(I)
100 CONTINUE
C
COMPUTE COLUMN HEIGHT ABOVE DIAGONAL (ME) AND CHECK IF MAXIMUM
 DO 200 I=1,ND
200 II=LM(I)
 IF (II.EQ.0) GO TO 200
 ME=II - LS
 IF (ME.GT.MHT(II)) MHT(II)=ME
 CONTINUE
C
RETURN
END
SUBROUTINE ERROR (N)
WRITE(6,2000) N
2000 FORMAT(//31H **ERROR** STORAGE EXCEEDED BY 16)
STOP
END

SUBROUTINE TITLE (HED)
DIMENSION HED(18)

THIS ROUTINE PRINTS THE TITLE CARD AT TOP OF OUTPUT PAGE

WRITE(6,2000) HED
2000 FORMAT(1HI,10A4,39X,8HDOT 1976/)
RETURN
END

SUBROUTINE ADRSK (MAXA,MHT,NUMNP,NUK,MA)

TO CALCULATE ADDRESSES OF DIAGONAL ELEMENTS IN A
BANDED MATRIX WHOSE COLUMN HEIGHTS ARE KNOWN.

MA = MAXIMUM BAND WIDTH
MHT = ACTIVE COLUMN HEIGHTS ABOVE DIAGONAL
MAXA = ADDRESSES OF DIAGONAL ELEMENTS
NUK = MAXIMUM STORAGE REQUIRED

DIMENSION MAXA(1),MHT(1)

MAXA(1) = 1
MAXA(2) = 2
MA = 0
IF (NUMNP.EQ.1) GO TO 100
DO 10 I = 2,NUMNP
IF (MHT(I).GT.MA) MA = MHT(I)
10 MAXA(I) = MAXA(I-1) + MHT(I) + 1
100 MA = MA + 1
NUK = MAXA(NUMNP+1) - 1

RETURN
END

SUBROUTINE ASSEMK

ASMK: 1

ASMK: 2

ASMK: 3

ASMK: 4

ASMK: 5

ASMK: 6

ASMK: 7

ASMK: 8

ASMK: 9

ASMK: 10

ASMK: 11

ASMK: 12

ASMK: 13

ASMK: 14

ASMK: 15

ASMK: 16

ASMK: 17

ASMK: 18

ASMK: 19

ASMK: 20

ASMK: 21

ASMK: 22
NGR = NPAR(1)
GO TO (1,2,3) NGR

---------
ELEMENT GROUP 1
---------

1 MYNODES = NPAR(5)
NDM = 2*MYNODES
ND5DIM = MYNODES-4
IF(ND5DIM.EQ.0) ND5DIM = 1

CALL COND1 (A(11),A(12),A(13),A(14),A(15),MYNODES,NDM,ND5DIM)
GO TO 100

---------
ELEMENT GROUP 2
---------

CALL COND2 (A(11),A(12),A(13))
GO TO 100

---------
ELEMENT GROUP 3
---------

CONTINUE

100 CONTINUE

RETURN

END

SUBROUTINE COND1 (LM,XY,IEL,T,HDM,T,MNODS,NDM,ND5DIM)

***********************************************************************
FORM THE EFFECTIVE SYSTEM STIFFNESS MATRIX (K*)
***********************************************************************

DIMENSION LM(MYNODES,1),XY(MYNODES,1),IELT(1),MCDS(ND5DIM,1)

COMMON /CHTRL1/ NUNIP,NEG,MODEX,NPAR(10),NG,NEC
COMMON /CHTRL2/ KST,NDT,DT,TSTWRT,TAR,P,UFPP,HTSPEF,THV
COMMON /DIM/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15,N16,CN
COMMON /TODIM/ HEL,MNODS,MTYPE,MNDS
COMMON /WORK/ BUM(10),SR(64),SC(8),HF(8),THOD(8),WORK(102)
COMMON H(1)

NEL1 = NPAR(2)

DO 100 H=1,NEL1
NEL = H
MNODS = IELT(H)
nNDS = MNODS - 4
NDOF = MNODS+MNODS

ZERO ELEMENT STIFFNESS MATRIX SR(MNODS,MNODS)

DO 10 I=1,NDOF
10 SR(I) = 0.

ZERO ELEMENT MASS VECTOE SC(MNODS)

DO 40 I=1,NDORS
40 SC(I) = 0.

RETURN

END
FORM ELEMENT STIFFNESS MATRIX USING GAUSS QUADRATURE

CALL FORMKI (SK,SC,XY(1,N),NODS(1,N),NODS)

ASSEMBLE EFFECTIVE ELEMENT STIFFNESS MATRIX (SK*) INTO EFFECTIVE STRUCTURAL STIFFNESS MATRIX (K*)

CALL ADDBAN (A(N10),A(N9),SK,LM(1,N),NODS)

CONTINUE

RETURN

SUBROUTINE FORMKI (SK,SC,XY,NODS,NODS)

DIMENSION SK(NODS,NODS), SC(NODS), XY(1), NODS(1)

COMMON /CNTRL1/ NUMP, NODEX, NPAR(10), H, KBC

COMMON /CNTRL2/ KST, NDT, DT, TSTART, TAMB, NTSREF, TIE, E

COMMON /WORK/ DUM(98), H(8), P(2,8), B(2,8), U(2,2)

DATA >G/0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0/1, 0.5773502691896, 0.5773502691896, 0, 0, 0, 0, 0, 0, 0/2, 0.7745966924150, 0.7745966552415, 0, 0, 0, 0, 0, 0, 0/3, 0.8611363115941, 0.8611363115941, 0, 0, 0, 0, 0, 0, 0/1, 0.0000000000000, 0.0000000000000, 1.0000000000000, 0, 0, 0, 0, 0, 0/2, 0.5555555555555, 0.5555555555555, 0.5555555555555, 0, 0, 0, 0, 0, 0/3, 0.3478548451375, 0.3478548451375, 0.3478548451375, 0, 0, 0, 0, 0, 0, 0/4, 0.0, 0, 0, 0, 0, 0, 0, 0, 0/5, 0, 0, 0, 0, 0, 0, 0, 0, 0/6, 0, 0, 0, 0, 0, 0, 0, 0, 0/7, 0, 0, 0, 0, 0, 0, 0, 0, 0/8, 0, 0, 0, 0, 0, 0, 0, 0, 0/

ITYP2D = NPAR(4)
NINT = NPAR(6)

LOOP OVER ALL INTEGRATION POINTS

DO 100 LX'=1,NINT
R = XG(LX,NINT)
DO 100 LY'=1,NINT
S = XG(LY,NINT)
WT = UGT(LX,NINT)*UGT(LY,NINT)

FIND INTERPOLATION FUNCTIONS (H) AND THEIR DERIVATIVES (P).
FIND JACOBIAN (XJ) AND ITS DETERMINANT (DETJ).

CALL SHAPE1 (R,S,XY,H,P,NODS,XJ,DETJ)

EVALUATE JACOBIAN INVERSE (XJI) AND GLOBAL DERIVATIVE OPERATOR (B)
AT EACH INTEGRATION POINT (R,S) WITHIN THE ELEMENT

CALL IERIV1 (XY,H,P,B,XJ,DETJ,RAD,ITYP2D)

FAC = WT*RAD*DETJ
VOL = VOL + FAC

FORM SK = B(TRANSPOSE)*EK*D FOR INTEGRATION POINT (R,S)

DO 50 I=1,NODS
BTI = B(1,I)

50 CONTINUE
BT2 = B(2,1)
DO 50 J=1,NODS
50  SK(I,J) = SK(I,J) + (BT1*B(1,J)+BT2*B(2,J))*FAC
C
100 CONTINUE
C
NODM = NODS - 1
DO 200 I=1,NODM
II = I + 1
DO 200 J=II,NODS
200  SK(J,I) = SK(I,J)
C
600 RETURN
END

SUBROUTINE SHAPE1 (R,S,XY,H,P,HODS,XJ,DETJ)

******************************************************************************

1. TO FIND INTERPOLATION FUNCTIONS (H) AND DERIVATIVES (P) CORRESPONDING TO THE NODAL POINTS OF A 4- TO 8-NODE TWO DIMENSIONAL ISOPARAMETRIC ELEMENT

2. TO FIND JACOBIAN (XJ) AND ITS DETERMINANT (DETJ)

NODE NUMBERING CONVENTION

2  5  1
0 . . . . . 0 . . . . . 0
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. S . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
6 0 . . . R 0 8
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
. . . . . . . . . . . . .
0 . . . . . 0 . . . . . 0

******************************************************************************

DIMENSION  XY(2,1),H(1),P(2,1),NODS(1),XJ(2,2)
COMMON /TODIM / NEL,NODS,MTYPE,NNDS
DIMENSION IPERM(4)
DATA IPERM/2,3,4,1/

INTERPOLATION FUNCTIONS (4-NODE ELEMENT)

RP = 1.0 + R
SP = 1.0 + S
KM = 1.0 - R
SM = 1.0 - S
R2 = 1.0 - R*R
S2 = 1.0 - S*S

H(1) = 0.25*RP*SP
H(2) = 0.25*R*RP
H(3) = 0.25*R*SP
H(4) = 0.25*RP*SM

LOCAL DERIVATIVES OF INTERPOLATION FUNCTIONS (4-NODE ELEMENT)
P(1.1) = 0.25*SP
P(1.2) = -P(1.1)
P(1.3) = -0.25*SM
P(1.4) = -P(1.3)
P(2.1) = 0.25*RP
P(2.2) = 0.25*RM
P(2.3) = -P(2.2)
P(2.4) = -P(2.1)

INTERPOLATION FUNCTIONS AND LOCAL DERIVATIVES FOR MIDS:DE NODES

IF(NODS.EQ.4) GO TO 50

I = 0
IF (I.GT.NND5) GO TO 40

5 H(5) = 0.50*R2*SP
P(1.5) = -R*SP
P(2.5) = 0.50*R2
GO TO 2

6 H(6) = 0.50*RM*S2
P(1.6) = -0.50*S2
P(2.6) = -RM*S2
GO TO 2

7 H(7) = 0.50*R2*SM
P(1.7) = -R*SM
P(2.7) = -0.50*R2
GO TO 2

8 H(8) = 0.50*RP*S2
P(1.8) = 0.50*S2
P(2.8) = -RP*S2
GO TO 2

MODIFY INTERPOLATION FUNCTIONS H(1) TO H(4) AND LOCAL DERIVATIVES

40 IH = 0
IF(IH.GT.NND5) GO TO 50

41 IH = IH + 1
IN = NODS(IH)
II = IN - 4
I2 = IPERH(I1)
H(II) = H(II) - 0.5*H(IN)
H(I2) = H(I2) - 0.5*H(IN)
H(IH+4) = H(IN)
DO 45 J=1,2
P(J,II) = P(J,II) - 0.5*P(J,IN)
P(J,I2) = P(J,I2) - 0.5*P(J,IN)
45 P(J,II+4) = P(J,IN)
GO TO 41

EVALUATE THE JACOBIAN MATRIX AT POINT (R,S)

50 DO 100 I=1,2
DO 100 J=1,2
SUM = 0.0
DO 90 K=1,NODS

A2-25
90 SUM = SUM + P(I,K) * XJ(J,K)
100 XJ(I,J) = SUM
C COMPUTE THE DETERMINANT OF THE JACOBIAN MATRIX AT POINT (R,S)
C
DETJ = XJ(1,1) * XJ(2,2) - XJ(1,2) * XJ(2,1)
DUM = ABS(DETJ)
IF(DUM.GT.1.0E-8) GO TO 500
WRITE(6,3000) NEL
STOP
C 3000 FORMAT(/&SH **ERROR** ZERO JACOBIAN DETERMINANT FOR ELEMENT NO..15)
C 500 RETURN
END SHP
SUBROUTINE DERIV1 (XY,H,P,B,XJ,DETJ,RAD,ITYP2D)
C EVALUATION OF THE GLOBAL DERIVATIVE OPERATOR (B) AT A POINT (R,S)
FOR A QUADRILATERAL ELEMENT HAVING PLANAR OR AXISYMMETRIC GEOMETRY
C
DIMENSION XY(2,1),H(1),P(2,1),B(2,1),XJ(2,2)
COMMON /TODIM/ NEL,NODS,MTYPE,NND5
COMMON /WORK / DUM(145),XJ(2,2),WORK(51)
C COMPUTE INVERSE OF THE JACOBIAN MATRIX
C
DETJI = 1.0/DETJ
XJI(1,1) = -XJ(1,1)*DETJI
XJI(1,2) = XJ(1,2)*DETJI
XJI(2,1) = -XJ(2,1)*DETJI
XJI(2,2) = XJ(2,2)*DETJI
C EVALUATE GLOBAL DERIVATIVE OPERATOR ( B-MATRIX )
DO 10 K=1,NODS
B(1,K) = XJI(1,1)*P(1,K) + XJI(1,2)*P(2,K)
B(2,K) = XJI(2,1)*P(1,K) + XJI(2,2)*P(2,K)
10 CONTINUE
C RAD = 1.0
IF(ITYP2D.NE.0) GO TO 500
C COMPUTE THE RADII AT POINT (R,S) FOR AXISYMMETRIC SOLIDS
C
RAD = 0.0
DO 50 K=1,NODS
RAD = RAD + H(K)*XY(1,K)
50 CONTINUE
C IF(RAD.GT.1.0E-8) GO TO 500
WRITE(6,3000) NEL
STOP
C 3000 FORMAT(/&SH **ERROR** ZERO RADII ENCOUNTERED IN ELEMENT NO..15)
C 500 RETURN
END DERI
SUBROUTINE ADDBAN (A,MAXA,S,LH,NDOF)
C ASSEMBLE ELEMENT STIFFNESS INTO COMPACTED GLOBAL STIFFNESS
C
DIMENSION A(I),MAXA(I),S(I),LM(I)

DO 200 J=1,NDOF
JJ = LM(J)
MJ = MAXC(JJ)
DO 200 I=1,NDOF
II = LM(I)
IJ = JJ - II
IF(IJ) .GT. 100.00,100
100 A(KK) = A(KK) + S(LS)
200 CONTINUE
RETURN
END

SUBROUTINE ADDC(R,SC,LM,IIL,NUMNP)
C
C ADDC
C ADD ELEMENT MASS TO GLOBAL LUMPED MASS VECTOR ADDC-
C
C DIMENSION A(I),SC(I),LM(I) ADDC
C
DO 100 I=1,IIL
II=LM(J)
100 A(I1) = A(I1) + SC(I)
RETURN
END ADDC

SUBROUTINE CON2(LM,XX,CL)
C
C CON2
C CALCULATE MASS MATRIX ADDC-
C
C DIMENSION LM(2,1),XX(2,1),CL(1) ADDC-
C COMMON /CTRL1/,NUMNP,SEG.MODEX,NPAR(10),NC,KBC ADDC-
C COMMON /CTRL2/,KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,TP ADDC-
C COMMON /DIM/,NT1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12,N13,N14,N15,CM2 ADDC-
C COMMON/CONST/CDON,ALPHA,THETA,DELTA ADDC-
C COMMON /XCG/WORK, DUM(HA,SC(2),WORK(188) ADDC-
C COMMON A(I) ADDC-
C MEL2 = NPAR(2)
C DO 100 N=1,NEL2
C
C AXI - SYMMETRIC FREE SURFACE BOUNDARY ELEMENTS
C
30 XI = XX(1,N)
XJ = XX(2,N)
WRITE(6.5) XI,XJ,CL(N),G,N
FORMAT(50X,E15.5,I10)
SC(1) = (2.0,XI+XJ)CL(N)/(6,36)
SC(2) = (2.0,XI+XJ)CL(N)/(6,36)
C
CALL ADDC(A(N12),SC,LM(I,N),2,NUMNP)
100 CONTINUE
RETURN
END

SUBROUTINE KSTAR(MAXA,XK,C)

C******************************************************************************
C  THIS SUBROUTINE CALCULATES EFFECTIVE K
C******************************************************************************
COMMON /CTRL1/ HUMHP,HEG,MODEX,NPAR(10),NG,KBC
COMMON/CONST /A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,THETA,DELT,ALPHA,PI,GKSTAR
+..RO
DIMENSION MAXA(1),XK(1),C(1)

NNI=MAXA(1)

DO 100 I=1,NNI
   XK(I)=XK(I)+C(I)*A0
100 CONTINUE
RETURN
END

SUBROUTINE eOlSOL(A,V,MAXA,NN,MA,m,KKK)

******************************************************************************
*  TO SOLVE SIMULTANEOUS EQUATIONS AX=V IN COPE, USING
*  COMPACTED STORAGE AND COLUMN REDUCTION SCHEME.
******************************************************************************
A = MATRIX STORED IN COMPACTED FORM
V = VECTOR TO BE REDUCED
MAXA = VECTOR CONTAINING ADDRESSES OF DIAGONAL ELEMENTS OF A
FLAG FOR TRIANGULARIZATION (A=LU) AND/OR SIMPLE FORWARD
REDUCTION (LY=V) AND BACKSUBSTITUTION (UX=Y)0
KKK=0 TRIANGULARIZATION ONLY
KKK=1 TRIANGULARIZATION PLUS SOLUTION
KKK=2 FORWARD REDUCTION AND BACKSUBSTITUTION ONLY
KKK=3 BACKSUBSTITUTION ONLY
******************************************************************************

DIMENSION A(NWA),V(1),MAXA(1)

MA1=MA-1
IF (KKK-2) 100,700.000

TRIANGULARIZATION

100 IF(NN.EQ.1) GO TO 800
   N=1
   IF (A(I)) 80,90.05,110
80 WRITE (6,3000) N
   STOP
90 WRITE (6,3001) N
   STOP

DO 200 N=2,NN
   KL=MAXA(N) + 1
   KU=MAXA(N+1) - 1
   IF (KU-KL) 200,210.210
210 DO=0.
   KN=MAXA(N)
   K=KN
   DO 220 KK=KL,KU
      K=K-1
      K1=MAXA(K)
      C=A(K1)/A(K)
555
220 CONTINUE
RETURN
END

B = B + C*K(A(KK))
A(KK) = C
A(KH) = A(KH) - B
C
IF (A(KN)) 222, 224, 226
WRITE (6.3000) N
STOP
WRITE (6.3001) N
STOP
MR = MIN0(MR, NN-N)
IF (MR) 200, 200, 228
MN = KU - KL + 1
DO 240 J = 1, MR
MJ = MAX(N+J, 1) - MJ - 1
IF (MNJ) 240, 240, 2313
ND = MIN0(MN, MNJ)
C = 0.
KU = KH + ND
IC = MJ - KN
DO 300 K = KL, KU
V(N) = V(N) / A(K)
IF (V(N) EQ. 0) RETURN
C
==================================================================
FORWARD REDUCTION
==================================================================
DO 400 N = 2, HN
KL = MAX(A(N), 1) + 1
KU = MAX(A(N+1), 1) - 1
IF (KU-1) 400, 410, 410
K = N
C = 0.
DO 420 K = KL, KU
V(N) = V(N) / A(K)
V(N) = V(N) - C
400 CONTINUE
GO TO 800
C
==================================================================
BALK SUBSTITUTION
==================================================================
DO 480 N = 1, HN
K = MAX(A(N), 1)
V(N) = V(N) / A(K)
IF (V(N) EQ. 0) RETURN
N = NN
DO 500 L = 2, NN
KL = MAX(A(N), 1) + 1
KU = MAX(A(N+1), 1) - 1
IF (KU-1) 500, 510, 510
K = N
DO 520 K = KL, KU
K = K - 1
520 V(K) = V(K) - A(KK) * V(N)
**FORMAT STATEMENTS**

```
300 FORMAT(//45H **STOP** STIFFNESS NOT POSITIVE DEFINITE.../)
10X,27H NEGATIVE PIVOT IN POSITION I4)
3001 FORMAT(//33H **STOP** ZERO PIVOT IN POSITION I4)
```

**CALCULATE THE LOADING VECTOR**

```
COMMON /CNTRL1/ NUMIP,NEG,MODEX,NPAR(10),HG,KBC
COMMON /CNTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP
COMMON /DIM /N1,N2,N3,N4,N5,N6,N7,N8,N9,N10
COMMON /Nbc /NNbc,Nbcf,Nptm
COMMON /Work /M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)
COMMON A(I)
DIMENSION HST(10)
EQUIVALENCE (HST(1),M1)
```

**REWIND 1**

```
REWIND 2
```

**LOOP OVER ALL ELEMENT GROUPS**

```
DO 100 IGR=LNEG
```

**READ**

```
READ (1) MIDEST,NPAR,NST,(A(I),I=1,MIDEST)
```

**IF(NGR,NE,3) GO TO 100**

```
CALL FLUX2 (A(M1),A(M2),A(M3),A(M4),A(M5),A(N1),A(N2),A(N3),A(N8),FMOC
1A(N11),A(N14),NPTM,TTH)
```

**100 CONTINUE**

```
RETURN
```

**SUBROUTINE FORMOC(TTH)**

```
SUBROUTINE FORMOC(TTH)
```

**RETURN END**

```
RETURN END
```

**SUBROUTINE FLUX2**

```
SUBROUTINE FLUX2 (LM,XX,CL,SINS,COS,S,TEN,FN,NPTM,T,T,ITH)
```

**RETURN END**

```
RETURN END
```

**DOUBLE PRECISION M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)**

**COMMON /CTRL1/ NUMIP,NEG,MODEX,NPAR(10),HG,KBC**

**COMMON /CTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP**

**COMMON /DIM /N1,N2,N3,N4,N5,N6,N7,N8,N9,N10**

**COMMON /Nbc /NNbc,Nbcf,Nptm**

**COMMON /Work /M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)**

**COMMON A(I)**

**DIMENSION HST(10)**

**EQUIVALENCE (HST(1),M1)**

**REWIND 1**

**REWIND 2**

**LOOP OVER ALL ELEMENT GROUPS**

```
DO 100 IGR=LNEG
```

**READ**

```
READ (1) MIDEST,NPAR,NST,(A(I),I=1,MIDEST)
```

**IF(NGR,NE,3) GO TO 100**

```
CALL FLUX2 (A(M1),A(M2),A(M3),A(M4),A(M5),A(N1),A(N2),A(N3),A(N8),FMOC
1A(N11),A(N14),NPTM,TTH)
```

**100 CONTINUE**

**RETURN**

```
RETURN
```

**SUBROUTINE FLUX2**

```
SUBROUTINE FLUX2 (LM,XX,CL,SINS,COS,S,TEN,FN,NPTM,T,T,ITH)
```

**RETURN END**

```
RETURN END
```

**DOUBLE PRECISION M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,WORK(190)**

**COMMON /CTRL1/ NUMIP,NEG,MODEX,NPAR(10),HG,KBC**

**COMMON /CTRL2/ KST,NDT,DT,TSTART,TAMB,NPRINT,NTSREF,TIME,KP**

**COMMON /DIM /N1,N2,N3,N4,N5,N6,N7,N8,N9,N10**

**COMMON /Nbc /NNbc,Nbcf,Nptm**

**DIMENSION L(I,1),L(1,1),L(1,1),CL(1),SINS(1),COS(1),T(1),FN,NPTM,1)**

**COMMON /Npar (10)**

**COMMON /Npar (4)**

**DO 10C N1=1,NEL3**

**N1 = LM(1,H)**

**JJ = LM(2,H)**
SUBROUTINE INTERP(TFH,FN,NPT,TIME,VAL)

DIMENSION TFH(1),FN(1)

DO 10 N=1,NPT
DTIME = TFH(N) - TIME
IF(DTIME.GT.0.) GO TO 15
10 CONTINUE

15 DIFF = TFH(N) - TFH(N-1)
VAL = FN(N) - (FN(N) - FN(N-1)) * DTIME / DIFF

RETURN

SUBROUTINE QEFF(Q,C,TD,TDD,NUMNP)

COMMON/CONST/A0,A1,A2,A3,A4,A5,A6,A7,A8,A9,THETA,DELTA,ALPHA,PI,G0EFF+
+RO

DIMENSION Q(1),C(1),T(1),TD(1),TDD(1)

DO 10 I=1,NUMNP
Q(I) = Q(I) + C(I) * (A0 * T(I) + A2 * TD(I) + A3 * TDD(I))
10 RETURN

SUBROUTINE PTEIP(T,TIME,NUMNP)

DIMENSION T(1)

This report was done with support from the Department of Energy. Any conclusions or opinions expressed in this report represent solely those of the author(s) and not necessarily those of The Regents of the University of California, the Lawrence Berkeley Laboratory or the Department of Energy.
TECHNICAL INFORMATION DEPARTMENT
LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
BERKELEY, CALIFORNIA 94720