

USER'S Manual
for

QUAD4M

A COMPUTER PROGRAM TO EVALUATE THE SEISMIC
RESPONSE OF SOIL STRUCTURES USING FINITE ELEMENT
PROCEDURES AND INCORPORATING A COMPLIANT BASE

by

MARTIN HUDSON
I. M. IDRISI
MOHSEN BEIKAE

Sponsored by

The National Science Foundation
Washington, D. C.

Center for Geotechnical Modeling
Department of Civil & Environmental Engineering
University of California
Davis, California

May 1994
rev. 2003

USER'S Manual
for

QUAD4M

A COMPUTER PROGRAM TO EVALUATE THE SEISMIC
RESPONSE OF SOIL STRUCTURES USING FINITE ELEMENT
PROCEDURES AND INCORPORATING A COMPLIANT BASE

by

MARTIN HUDSON
I. M. IDRISI
MOHSEN BEIKAE

Sponsored by

The National Science Foundation
Washington, D. C.

Center for Geotechnical Modeling
Department of Civil & Environmental Engineering
University of California
Davis, California

May 1994
rev. 2003

Table of Contents

Abstract	1
Introduction	1
Evaluation of seismic response using the finite element procedure	2
Transmitting Boundaries	5
Seismic Coefficient Computation	6
Restart Capability	6
Damping	7
Implementation of Finite Element Model	10
Examples and Comparisons with Other Solutions	11
100-ft Layer of Sand	11
Example of a Seismic Coefficient Calculation	13
Summary	13
Acknowledgements	14
References	15
Appendix A - Program Listing of QUAD4M	A - 1
Appendix B - Sample Input	B - 1
Appendix C - Sample Output	C - 1

QUAD4M

A COMPUTER PROGRAM TO EVALUATE
THE SEISMIC RESPONSE OF SOIL STRUCTURES
USING FINITE ELEMENT PROCEDURES
AND INCORPORATING A COMPLIANT BASE

Martin Hudson¹, I.M. Idriss², and Mohsen Beikae³

Abstract

QUAD4M, a dynamic, time domain, equivalent linear two-dimensional computer program, was written as a modification of QUAD4 to implement a transmitting base, an improved time-stepping algorithm, seismic coefficient calculations, a restart capability, a change in the algorithm by which damping is set, and various computational enhancements to fully bring the program into the environment of the microcomputer. Various sample problems were run to verify the processing of the program, and the results are presented. The results compare well with SHAKE91, a one-dimensional closed-form solution program.

Introduction

The finite element method of analysis is a widely used computational procedure for the solution of problems in continuum mechanics, as well as many other fields. The procedure has been found very powerful for modelling the seismic response of soil deposits and earth structures. Programs to solve such response have been written using time domain solutions as well as frequency domain solutions in the past 30 years.

-
- ¹ Project Engineer, Law/Crandall, Los Angeles, California; formerly graduate student, University of California, Davis
 - ² Professor, Department of Civil Engineering and Environmental Engineering, University of California, Davis
 - ³ Senior Engineer, Woodward-Clyde Consultants, Santa Ana, California

QUAD4 (Idriss, Lysmer, Hwang, and Seed, 1973) was written as a two-dimensional, time domain solution to dynamic soil response. It incorporated for the first time independent damping in each element in the continuum.

QUAD4M incorporates into **QUAD4** a transmitting base so that the half-space beneath a mesh can be modeled and the need to assume a rigid foundation can be eliminated. The shear and compression wave velocities and the unit weight for the material underlying the mesh can be entered, and the response of the mesh on top of that half-space can now be modeled with greater accuracy.

In addition, seismic coefficients have been added in this version of the program. This feature is particularly useful in deformation analyses. The program also has a restart capability. The acceleration, velocity, and displacement are stored for the restart so that the program continues as if no interruption had occurred. This feature is useful for changing material properties during the shaking event.

Finally, **QUAD4M** incorporates a new method for the formulation of damping matrices which results in a significant reduction of the damping of higher frequencies commonly associated with the use of a Rayleigh damping formulation.

Evaluation of seismic response using the finite element procedure

The finite element procedure has been used extensively over the past 30 years for estimating the response of soil structures or deposits to static and dynamic loading conditions. It consists of numerically modeling a continuum with a finite number of elements interconnected at their common nodes.

The finite element procedure uses a system of equations represented in matrix form as:

$$[M]\ddot{u} + [C]\dot{u} + [K]u = R \quad (1)$$

where:

- [M] mass matrix (in this case using the assumption of a lumped mass formulation);
- [C] damping matrix;
- [K] stiffness matrix;
- R load vector, which is given by:

$$R = [M]\ddot{u}_o$$

- u relative displacement vector; and dots represent differentiation with respect to time;
- \ddot{u}_o outcrop acceleration.

As was originally constructed in QUAD4, the damping matrix is formulated using the assemblage of element damping matrices constructed using the Rayleigh formulation (1945):

$$[C]_q = \alpha_q[M]_q + \beta_q[K]_q \quad (2)$$

for each element q . The use of strain compatible damping at the element level was first introduced in Idriss, Lysmer, Hwang, and Seed (1973). The values of α_q and β_q are chosen as described in the section on damping.

The entire solution is iterated upon the number of times specified by the user, in order to obtain strain compatible damping and modulus values.

To solve equation (1), it is necessary to introduce equations relating \ddot{u} , \dot{u} , and u . The Newmark family of methods (e.g. Hughes, 1987) uses the following equations to fulfill the above requirement:

$$\begin{aligned}\underline{u}_N &= \underline{u}_{N-1} + \Delta t[(1-\gamma)\ddot{u}_{N-1} + \gamma\ddot{u}_N] \\ \underline{u}_N &= \underline{u}_{N-1} + \Delta t\underline{v}_{N-1} + \frac{\Delta t^2}{2}[(1-2\beta)\ddot{u}_{N-1} + 2\beta\ddot{u}_N]\end{aligned}\quad (3)$$

where N is the current time step (quantities unknown), and $N-1$ is the previous time step (quantities known). The use of equations (3) with $\gamma = 0.5$ and $\beta = 0.25$ is called the trapezoidal rule and provides a time-stepping algorithm with unconditional stability, quadratic convergence, and no numerical damping of any frequencies (Hughes, 1987).

Using the Trapezoidal rule, the following equations are obtained for solving the displacement, velocity, and acceleration at each time step:

$$\underline{u}_{N,1} = [\bar{K}]^{-1}[\bar{R}]_{N,1} \quad (4a)$$

$$\ddot{u}_{N,1} = \frac{4}{\Delta t^2}(\underline{u}_{N,1} - \underline{u}_N) - \frac{4}{\Delta t}\dot{u}_N - \ddot{u}_N \quad (4b)$$

$$\underline{u}_{N,1} = \underline{u}_N + \frac{\Delta t}{2}(\ddot{u}_N + \ddot{u}_{N,1}) \quad (4c)$$

$$[\bar{K}] = \frac{4}{\Delta t^2}[M] + \frac{2}{\Delta t}[C] + [K] \quad (4d)$$

$$[\bar{R}]_{N,1} = [R]_{N,1} + [M]\underline{A}_{N,1} + [C]\underline{B}_{N,1} + [K]\alpha\underline{u}_N \quad (4e)$$

$$\underline{A}_{N,1} = \frac{4}{\Delta t^2} \left(\underline{u}_N + \Delta t\underline{v}_N + \frac{\Delta t^2}{4}\ddot{u}_N \right) \quad (4f)$$

$$\underline{B}_{N,1} = \frac{2}{\Delta t}\underline{u}_N + \underline{v}_N \quad (4g)$$

Transmitting Boundaries

In order for a two-dimensional finite mesh to represent the response of an infinite field condition, the artificial reflection of seismic waves from side boundaries, as well as from the underlying half-space, should be minimized. Lysmer and Kuhlemeyer (1969) introduced a simple procedure to accomplish this. They suggested the use of dampers as illustrated in figure 1 for the case of a vibrating footing. In the case of a soil mass subjected to earthquake vibrations, the implementation of a compliant base in QUAD4M is the same as the Lysmer and Kuhlemeyer scheme.

The implementation of these dampers involves adding damping at each of the nodes that make up the base and sides of the finite model. For the present study, only the base dampers have been implemented. The base dampers are more essential to incorporate than the side dampers because the finite element system under consideration will always be placed over a half-space. The effects of side boundaries can be readily minimized by increasing the extent of the finite element mesh.

To mathematically implement these dampers, the parts of the applicable element matrices have the transmitting boundary damping term added to the diagonal terms. This produces an adjustable force in the x and y direction proportional to the velocity of the specified nodes. The coefficients added on to the diagonal terms are obtained as:

$$\begin{aligned} \text{Term for direction perpendicular to boundary: } & \rho V_p L \\ \text{Term for direction parallel to boundary: } & \rho V_s L \end{aligned}$$

The velocity of the P or S waves is used for the material in the half space below the finite element model, as is the density, ρ . The "tributary width" of the node, L is that length corresponding to half of the distance to the next node on both sides.

When a transmitting base is used, the input motion is a function of the material properties of the half-space below the mesh, and the properties and geometry of the mesh. This is the correct choice for a boundary condition when the input motion represents an outcrop acceleration, recorded at an outcrop of the half-space material. If an infinitely stiff ($V_s \rightarrow \infty$) rock is specified under the underlying stratum, then the input motion will not be affected by the mesh above.

Seismic Coefficient Computation

A seismic coefficient is the ratio of the force induced by an earthquake in a block of the mesh, over the weight of that block.

The forces acting on the block are computed by multiplying the shear and normal stresses acting on an element by the width of that element. Since the surface of the block is specified as going through the nodes (between elements) in QUAD4M, the average stress is found between the elements on either side of the interface. The summation of forces acting on a block is computed as a function of time. The seismic coefficient is then computed for each time step.

Restart Capability

A feature has been added in QUAD4M whereby at the conclusion of the calculations, the acceleration, velocity, and displacement of every node is saved. This can then be used to restart the program at the time step following the last time step used in the previous run of the program. Before the program is restarted, the soil properties can be changed. The program can be stopped and restarted as many times as is desired during the course of an earthquake.

Damping

The damping matrix is formulated using the assemblage of element damping matrices constructed in this manner:

$$[C]_q = \alpha_q [M]_q + \beta_q [K]_q \quad (5)$$

for each element q . The use of rayleigh damping in this manner results in a frequency dependent damping applied to the problem, with

$$\lambda_q = \frac{1}{2} \left(\frac{\alpha_q}{\omega} + \beta_q \times \omega \right) \quad (6)$$

The damping in soil is not frequency dependent. Therefore, the choice of α_q and β_q must be made that provides for damping values that have minimum variations over the range of frequencies of interest. In QUAD4, the constants were chosen in such a way that the damping was minimized at the fundamental frequency of the entire finite element model, ω_1 . The justification for this is that the first mode of vibration has the highest participation factor of all the modes. Using this criterion, the values of α_q and β_q are chosen as follows for each element:

$$\begin{aligned} \alpha_q &= \lambda_q \times \omega_1 \\ \beta_q &= \lambda_q / \omega_1 \end{aligned} \quad (7)$$

As is the case with all procedures that utilize a Rayleigh Damping formulation, the higher frequencies are overdamped. Therefore, in QUAD4M, a new scheme for setting damping is employed. Instead of using a single frequency (the fundamental frequency of the model), and a slope (0 at the fundamental frequency of the model) to establish the constants in

equation 6, two frequencies are used to establish these constants. The choice of these two frequencies has been studied using several different earthquakes and several different one-dimensional deposits. One-dimensional deposits were used because comparison with SHAKE-91 (Idriss and Sun, 1992) (which uses a constant value of damping for all frequencies) can be made. One frequency is chosen at the fundamental frequency of the model, as in QUAD4. The second frequency is established as

$$\omega_2 = n \omega_1 \quad (8)$$

where n is an odd integer. This choice was guided by the response of a shear beam in which the frequencies of higher modes are odd multiples of the frequency of the fundamental mode of the beam. The parameter n is chosen such that:

$$n = \text{closest odd integer greater than } \omega_1 / \omega_1 \quad (9)$$

where ω_1 is the predominant frequency of the input earthquake motion.

To set damping at two frequencies, the values of α_q and β_q are then given by the following expression for each element (Hudson, 1994):

$$\begin{aligned} \alpha_q &= 2\lambda_q \frac{\omega_1 \omega_2}{\omega_1 + \omega_2} \\ \beta_q &= 2\lambda_q \frac{1}{\omega_1 + \omega_2} \end{aligned} \quad (10)$$

The use of this two-frequency scheme results in under-damping between ω_1 and ω_2 , and over-damping outside that range. This scheme allows the model to respond to the predominant frequencies of the input motion without experiencing significant over-damping.

The element damping ratios, λ_q , are chosen based upon the average strain developed in the element. The value of ω_1 , the fundamental frequency of the system, is internally calculated by solving the following system of equations:

$$K\phi^1 = \omega_1^2 M\phi^1 \quad (11)$$

where the first mode shape is represented by ϕ^1 .

The difference between the QUAD4 and the QUAD4M damping schemes is illustrated using a 200 foot soil deposit with an average shear wave velocity, $V_s = 1200$ ft/sec, and a total unit weight, $\gamma = 120$ pcf. The modulus reduction and damping curves for sand were used for this example. The N-S component of the Santa Cruz record of the Loma Prieta earthquake was used as input rock outcrop motion.

The fundamental frequency of this 200-ft soil layer under small strain conditions is given by:

$$f_1 = \frac{V_s}{4H} = \frac{1200}{4 \times 200} = 1.5 \text{ hz}$$

The response spectrum for the input rock outcrop motion is shown in figure 3. The predominant period, T_i , of this motion is about 0.15 sec; the predominant frequency $f_i = 1/T_i = 6.7$ hz. Using Equation 8, the ratio $n = 6.7/1.5 = 4.1$, hence a value of $n = 5$ is used in the first iteration. For the last iteration, the strain compatible moduli lead to a fundamental period of 0.78 sec or $f_1 = 1.28$ hz. The ratio $f_i/f_1 = 5.23$; hence $n = 7$ is used in the last iteration and the variation of damping with frequency is shown in figure 2. Figure 4 illustrates the response spectrum obtained at the top of the layer using programs SHAKE91, QUAD4M, and QUAD4.

Implementation of Finite Element Model

The finite element computational procedure described previously was implemented by modifying program QUAD4 as presented in Idriss, Lysmer, Hwang, and Seed (1973). The revisions consisted of changing the time-stepping algorithm, incorporating transmitting base elements, incorporating the calculation of seismic coefficients for selected potential sliding blocks, adding restart capability, modifying the damping computation, and modifying some computational procedures. All other features of QUAD4 are still intact in the new program.

The time stepping method was changed from the Wilson-θ method to the trapezoidal rule as described previously.

The transmitting base elements were incorporated as described above, and the user must now specify the P and S wave velocities and the unit weight of the half-space below the base elements.

Seismic coefficients can now be computed. The user specifies elements within the block, and the nodes bounding the block.

The restart feature is utilized by specifying a switch in the input file, and providing a file name to which the restart input file is recorded. The restart input file echoes all the information in the original input file in addition to the displacement, velocity, and acceleration of each node at the end of the last time step. This input file can then be modified to continue the calculations with the previous properties or with a new set of properties.

The new damping scheme is controlled by specifying the predominant period of the input motion, obtained from the response spectrum. The output file records the two frequencies at which the damping is set.

Computational modifications were made to make the program conform to a structured Fortran language, implementing data structures to describe the elements and nodes. In

addition, the arrays and matrices have been made allocatable in size, so that the program adjusts its memory usage according to the input problem. When the code is compiled for Microsoft Windows 3.x using the Microsoft Fortran 5.1 compiler, or using the Microsoft Powerstation compilers, the program can accept as large a problem as there is available memory on the microcomputer.

Finally, a few of the subroutines were rewritten to be easier to follow and to increase their computational efficiency.

The new code, called QUAD4M, is presented in Appendix A.

Examples and Comparisons with Other Solutions

100-ft Layer of Sand

A 100-ft layer of sand having a total unit weight, $\gamma = 125$ pcf, $K_0 = 0.5$, and $K_{2\max} = 65$ was used, in which K_0 is the coefficient of earth pressure at rest, $K_{2\max}$ is a parameter relating maximum shear modulus, G_{\max} , and effective confining pressure, σ'_m , by $G_{\max} = 1000(K_2)_{\max} \sqrt{\sigma'_m}$. The effective confining pressure is given by $\sigma'_m = \frac{1+2K_0}{3} \sigma'_v$, in which σ'_v is the effective vertical pressure. Note that the values of σ'_v , σ'_m and G_{\max} are in pounds per square foot in these equations. This sand layer was assumed to be underlain by a half space having a shear wave velocity of 3000 ft/sec, a compression wave velocity of 7350 ft/sec and a total unit weight of 135 pcf.

The variation of shear modulus and damping with shear strain used for evaluating the response of this layer are shown in Figure 6 and Table 1. These values were taken from the SHAKE91 manual. The finite mesh used for the 100' sand layer is shown in figure 5.

The case of a 100' dense sand layer with the Santa Cruz record of the Loma Prieta earthquake scaled to 0.3g and 0.6g are shown in figures 7 through 10.

Figure 7 and 8 show the response spectra and time histories, respectively, obtained at the surface of the 100' layer when the Santa Cruz record of the Loma Prieta earthquake scaled to a peak acceleration of 0.3g is applied at the base. The QUAD4M and SHAKE91 results are compared in these figures. Figures 9 and 10 show the response spectra and time histories, respectively, obtained in the same manner, but using the input acceleration time history scaled to 0.6g.

Figures 11 and 12 show the variation in peak horizontal acceleration and maximum shear stress with depth for the case of the 0.3g input and the 0.6g input, respectively. These figures also show the results obtained using both QUAD4M and SHAKE91.

Table 1: Dynamic Soil Properties for Sand		
Shear Strain (%)	G/Gmax (%)	Damping (%)
.0001	100	.24
.0003	100	.42
.001	99	.8
.003	96	1.4
.01	85	2.8
.03	64	5.1
.1	37	9.8
.3	18	15.5
1	8	21

These comparisons show that the QUAD4M and SHAKE91 provide almost identical results for maximum shear stresses and for peak horizontal accelerations. Differences between the results of the two programs arise because of the use of the Rayleigh formulation for damping in QUAD4M.

Example of a Seismic Coefficient Calculation

A second analysis was performed to illustrate the seismic coefficient capabilities of QUAD4M. This problem consists of a 50 foot embankment underlain by 50 feet of soil. The embankment is shown in figure 13. The Young's modulus used is 1×10^6 , the unit weight is 120pcf, the Poisson's ratio is 0.45, and the shear wave velocity is hence 304fps.

In this example, twelve surfaces are chosen with various depths and extents into the embankment. The surfaces are shown in figure 14. The maximum seismic coefficient is shown for those twelve surfaces in figure 15. It can be seen that the surfaces extending farther into the embankment approximate more closely the semi-infinite solution. This indicates that the seismic coefficients are correctly converging to the free field solution.

Summary

QUAD4 has been updated to include the trapezoidal rule, transmitting boundaries and sliding block seismic coefficients have been added, restart capability has been introduced, the damping formulation has been changed, and various computational changes have been performed to update the program to use on a personal computer.

It is hoped that this new version of QUAD4 will provide improved means for calculating the response of soil deposits and soil structures during earthquakes using a time-domain method of analysis.

Acknowledgments

The work reported in this report was supported by Grant Number NSF BCS-90-11322 "I" from the National Science Foundation; the contract monitor was Dr. Cliff Astil. This support is gratefully acknowledged. Our thanks are also extended to Professor Karl Romstad of the University of California at Davis, for valuable discussions regarding the use of the Rayleigh damping formulation.

References

Hudson, Martin B (1994). *Behavior of Slopes and Earth Dams During Earthquakes*. Doctoral Thesis, University of California, Davis.

Hughes, Thomas J.R. (1987). *The Finite Element Method*. London: Prentice Hall.

Idriss, I.M., J. Lysmer, R. Hwang, and H. Bolton Seed (1973). *Quad-4: A Computer Program for Evaluating the Seismic Response of Soil Structures by Variable Damping Finite Element Procedures*. EERC Report 73-16. Berkeley: University of California, Berkeley.

Idriss, I. M. and Joseph I Sun (1992). *"A Computer Program for Conducting Equivalent Linear Seismic Response Analyses of Horizontally Layered Soil Deposits" - Program Modified based on the Original SHAKE program published in December 1972 by Schnabel, Lysmer & Seed*. Center for Geotechnical Modeling, Department of Civil & Environmental Engineering, University of California, Davis, California.

Lysmer, John, and Roger L. Kuhlemeyer (1969). "Finite Dynamic Model for Infinite Media" *Journal of the Engineering Mechanics Division, Proceedings of the American Society of Civil Engineers*. EM4:859-877.

Newmark, N.M. (1965). "Effects of Earthquakes on Dams and Embankments" *Geotechnique*. XV:139-160.

Rayleigh, L. (1945). *Theory of Sound, Vol. I*. New York: Dover Publication.

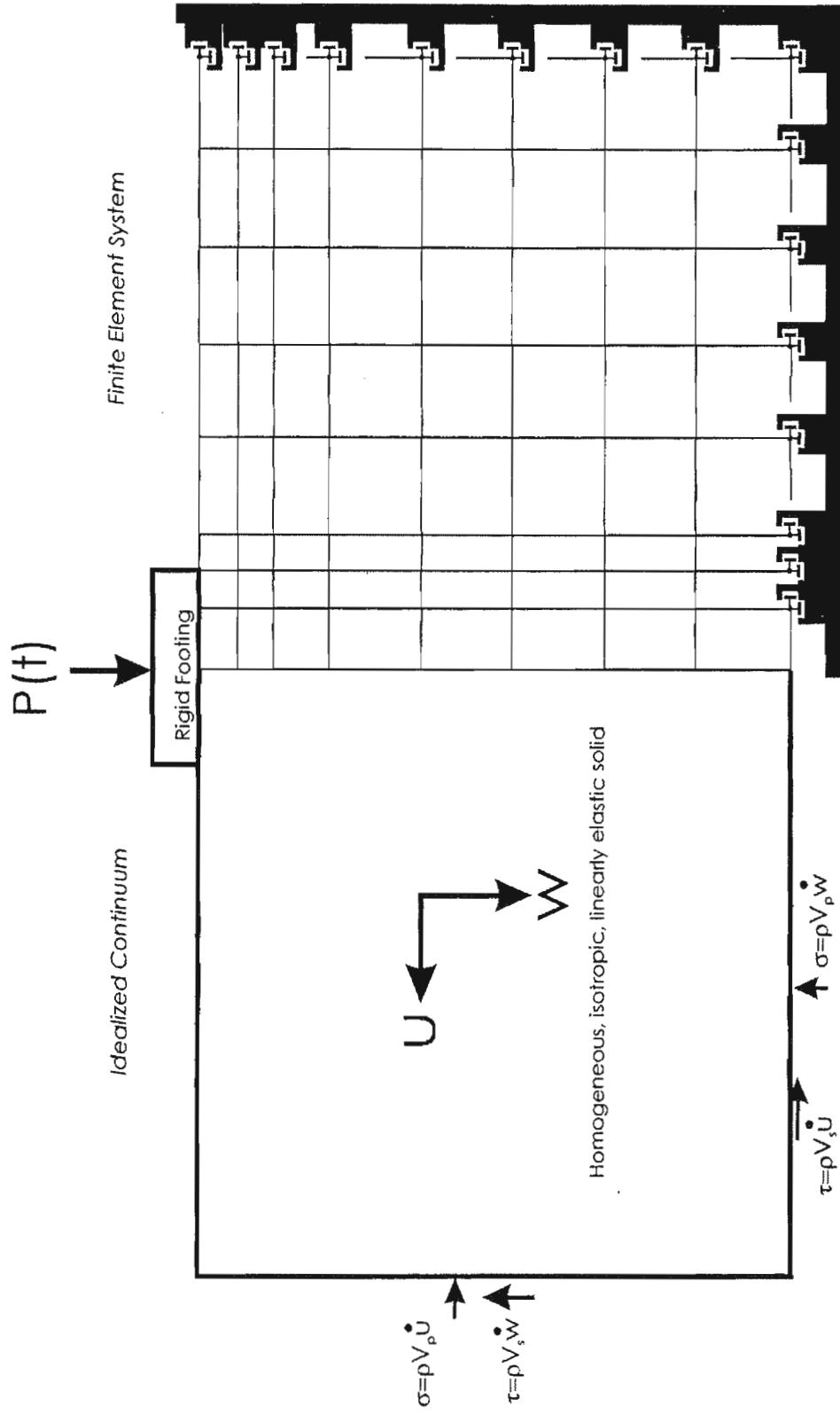


Figure 1
Finite Models for Footing on Half Space
(after Lysmer & Kuhlemeyer, 1969)

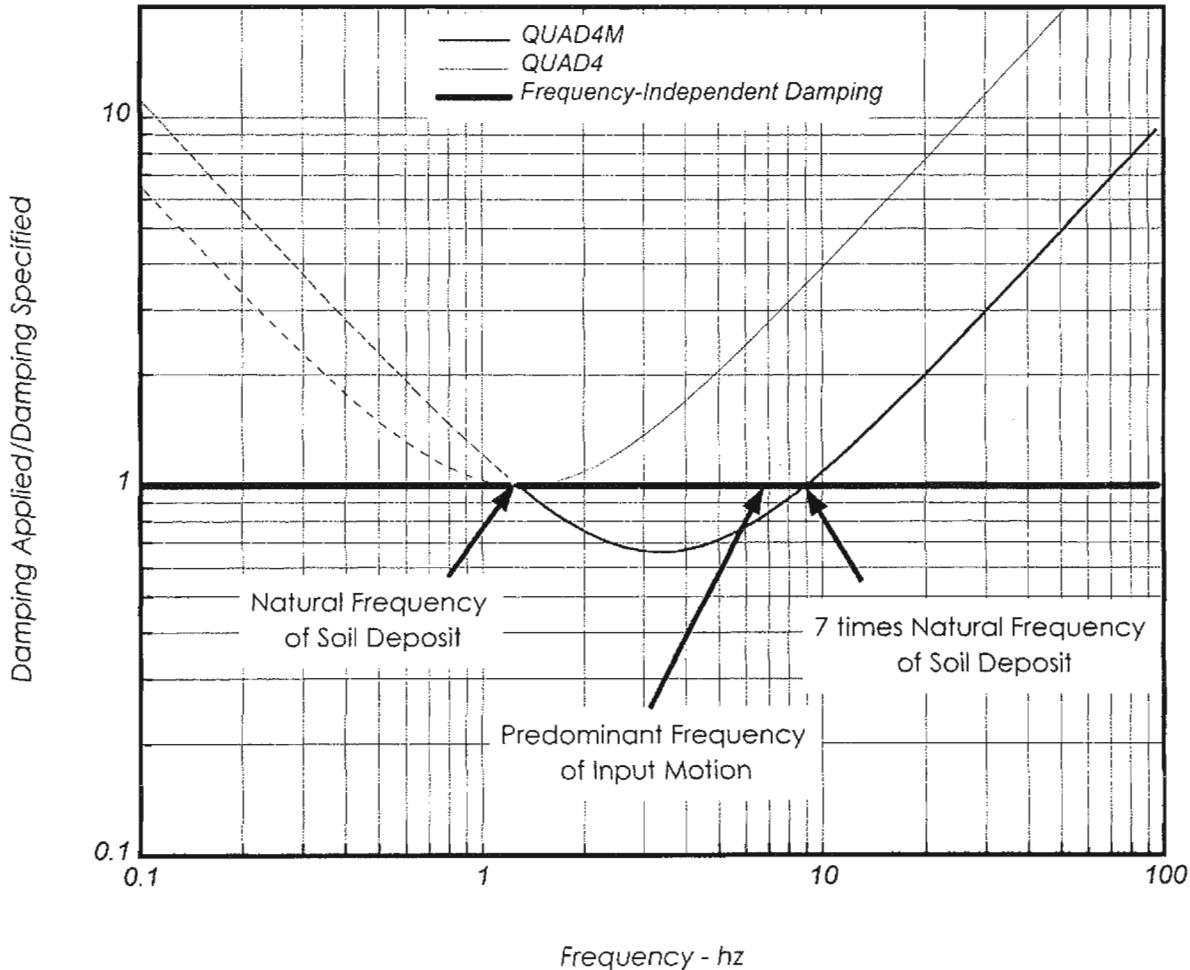


Figure 2
Variation of Damping with Frequency
200 foot sand deposit
Santa Cruz Record of Loma Prieta Earthquake

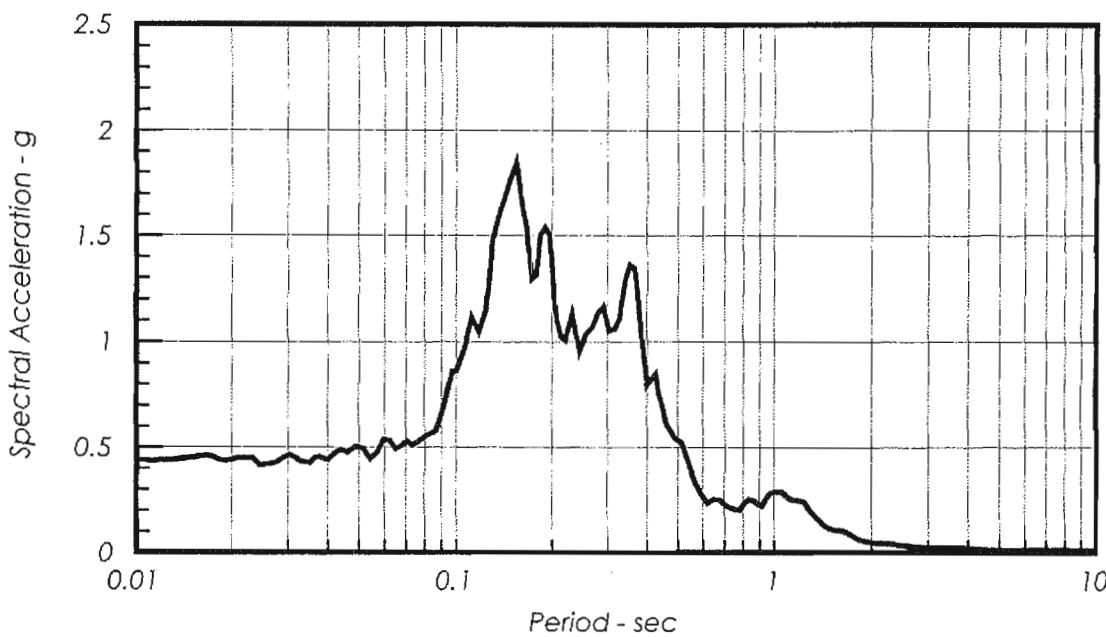


Figure 3
Response Spectrum for Earthquake Ground Motion used as
Input Rock Outcrop Motion in Sample Profile
Santa Cruz Record, 0 degree component,
Loma Prieta Earthquake

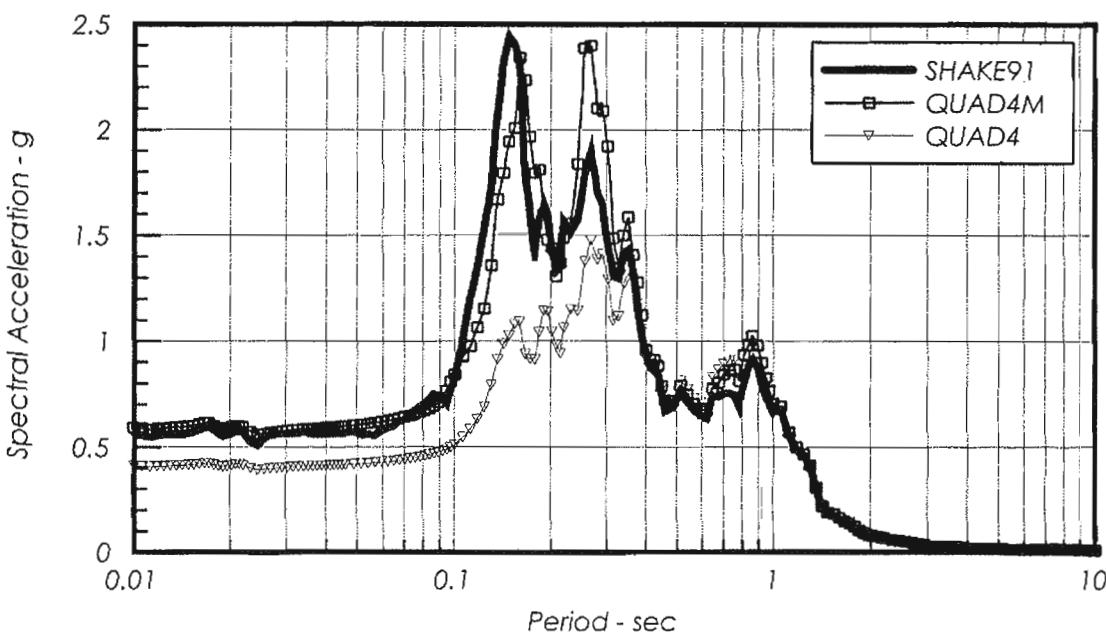


Figure 4
Computed Response
Spectral Ordinates at Ground Surface of Sample Profile
Using Various Damping Schemes

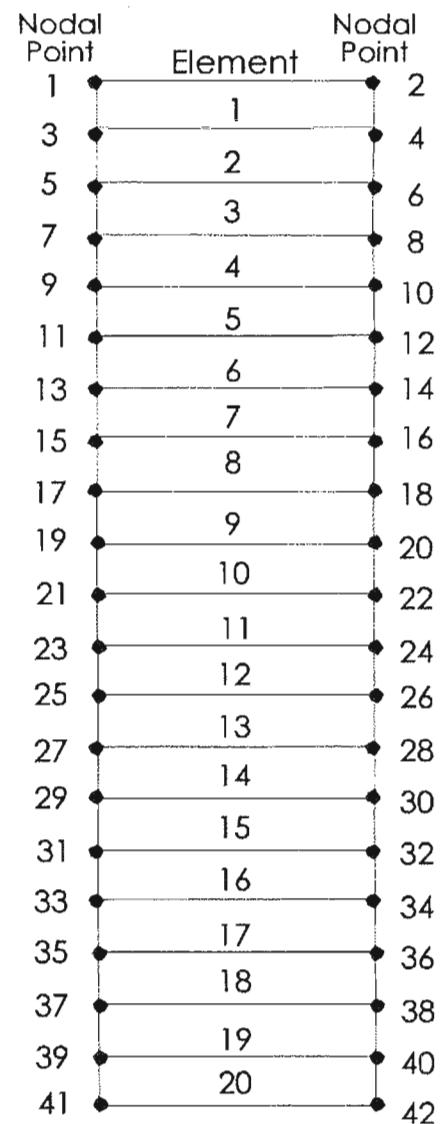
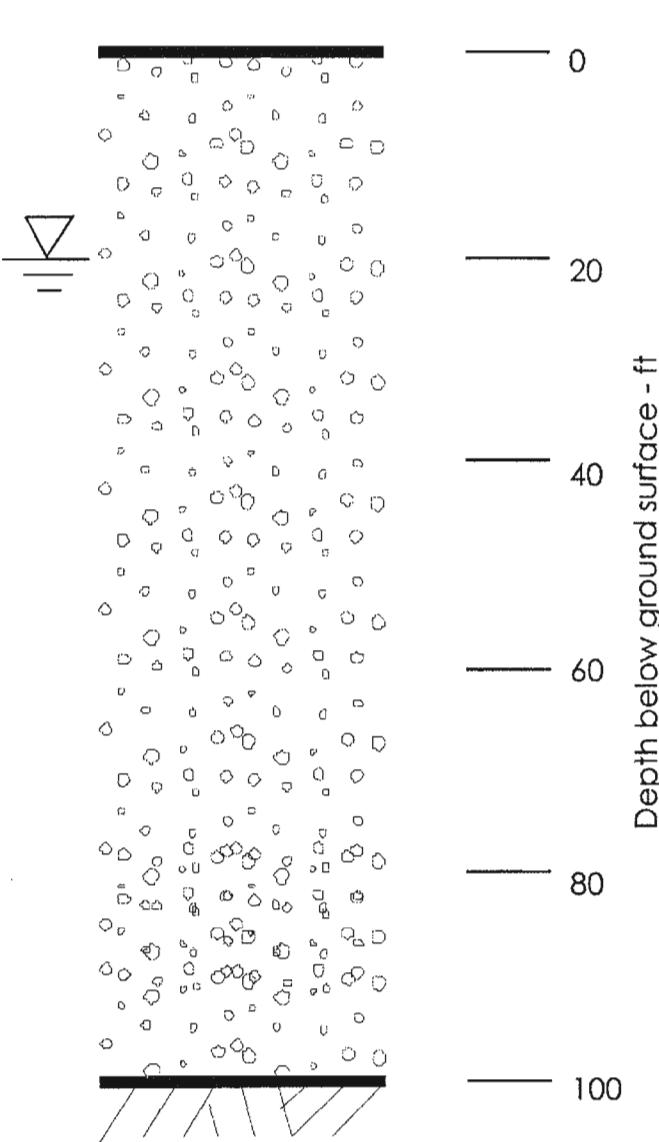


Figure 5
Finite Element Mesh Used for 100-ft Layer of Sand

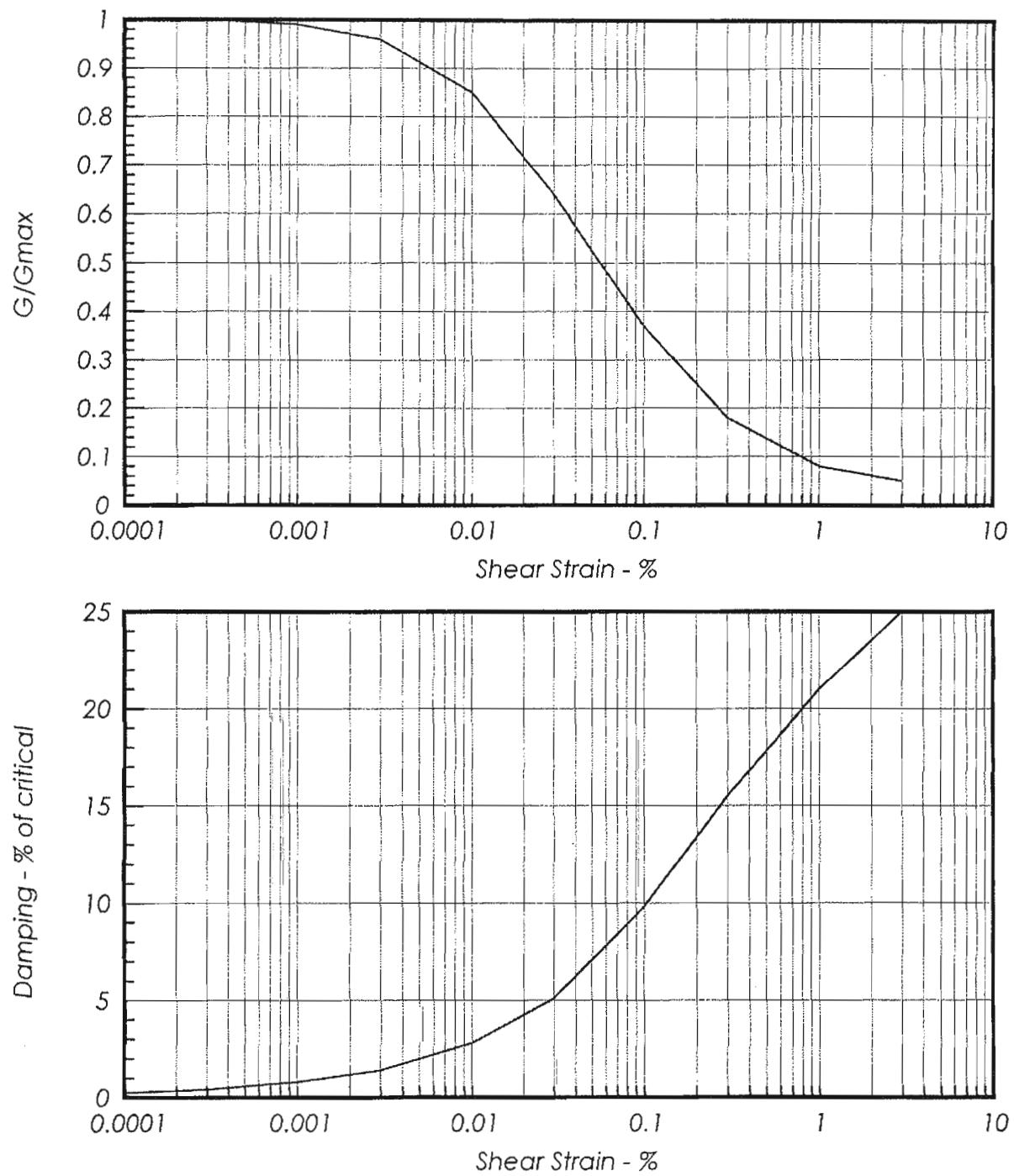


Figure 6
Dynamic Soil Properties for Sand

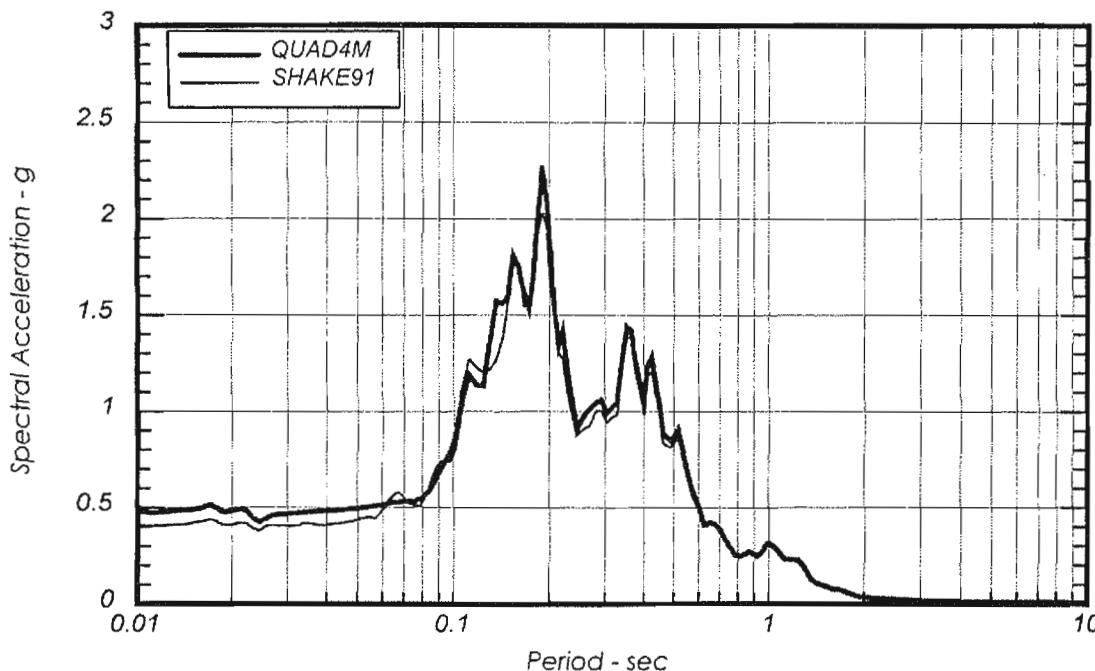


Figure 7
*Comparison of Response Spectra at Ground Surface
 Using Programs QUAD4M and SHAKE91
 for the 100 Foot Dense Sand Layer --
 Input Motion: Santa Cruz Record Scaled to 0.3g*

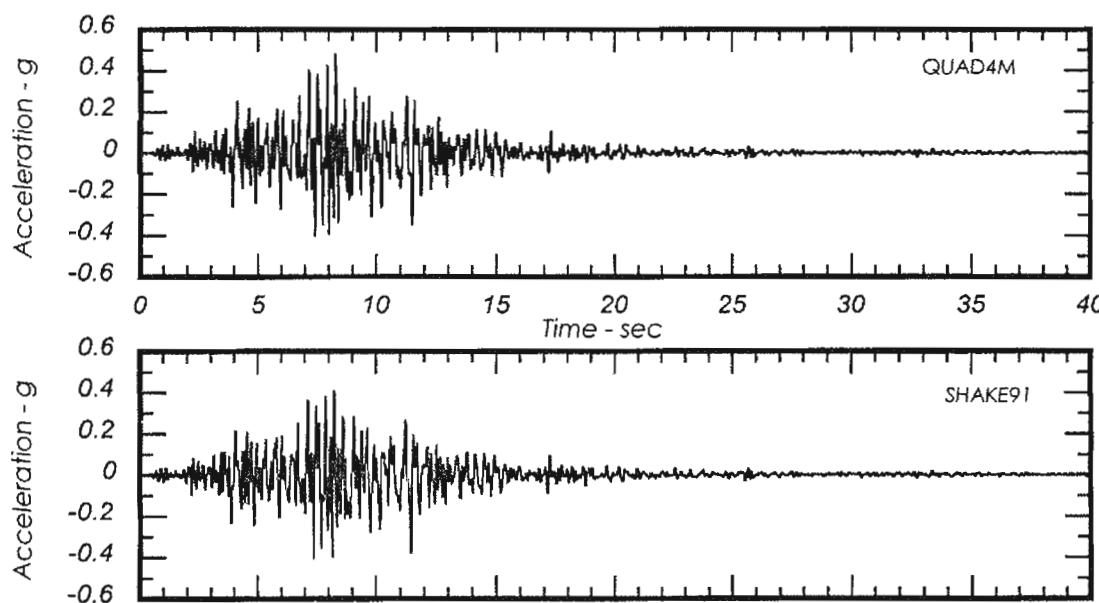


Figure 8
*Comparison of Time Histories at Ground Surface
 Computed Using QUAD4M and SHAKE91
 for the 100 Foot Dense Sand Layer --
 Input Motion: Santa Cruz Record Scaled to 0.3g*

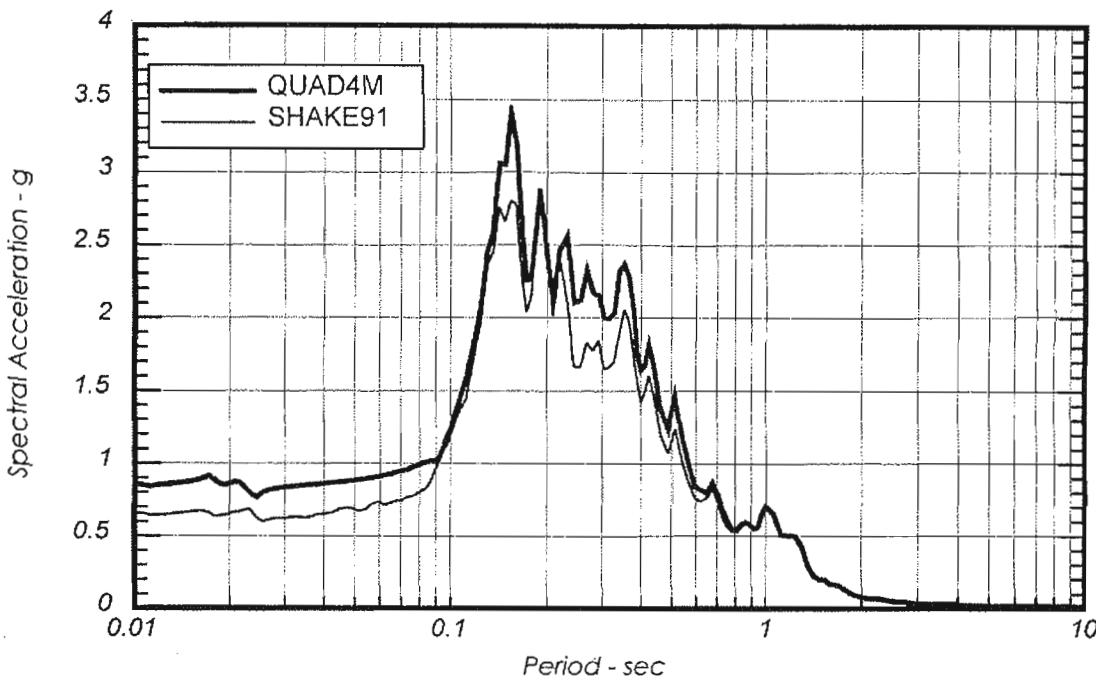


Figure 9

*Comparison of Response Spectra at Ground Surface
Using Programs QUAD4M and SHAKE91
for the 100 Foot Dense Sand Layer --
Input Motion: Santa Cruz Record Scaled to 0.6g*

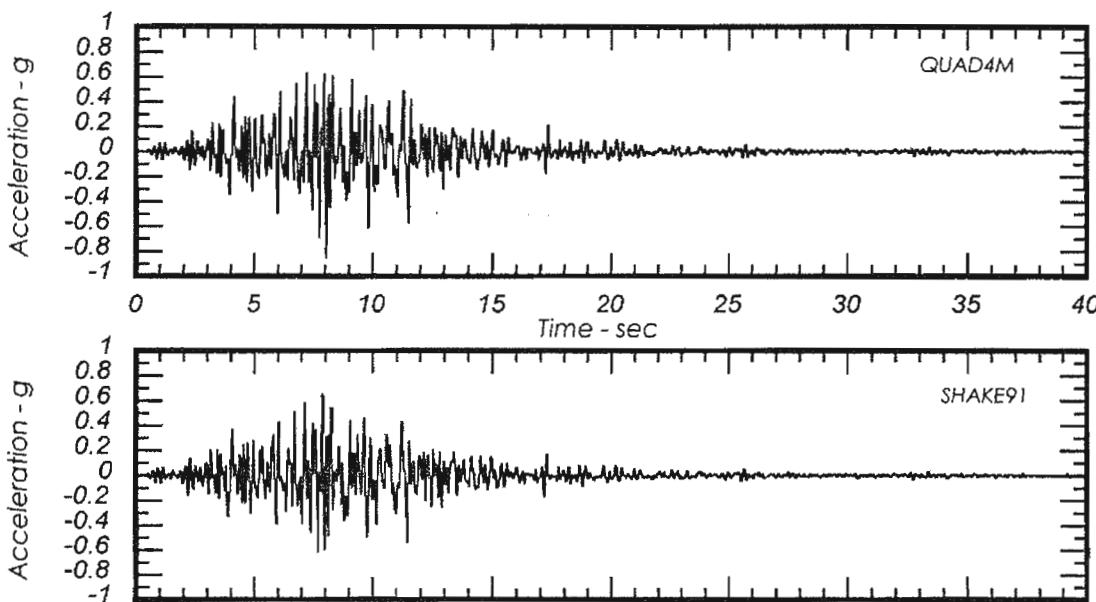


Figure 10

*Comparison of Time Histories at Ground Surface
Using Programs QUAD4M and SHAKE91
for the 100 Foot Dense Sand Layer --
Input Motion: Santa Cruz Record Scaled to 0.6g*

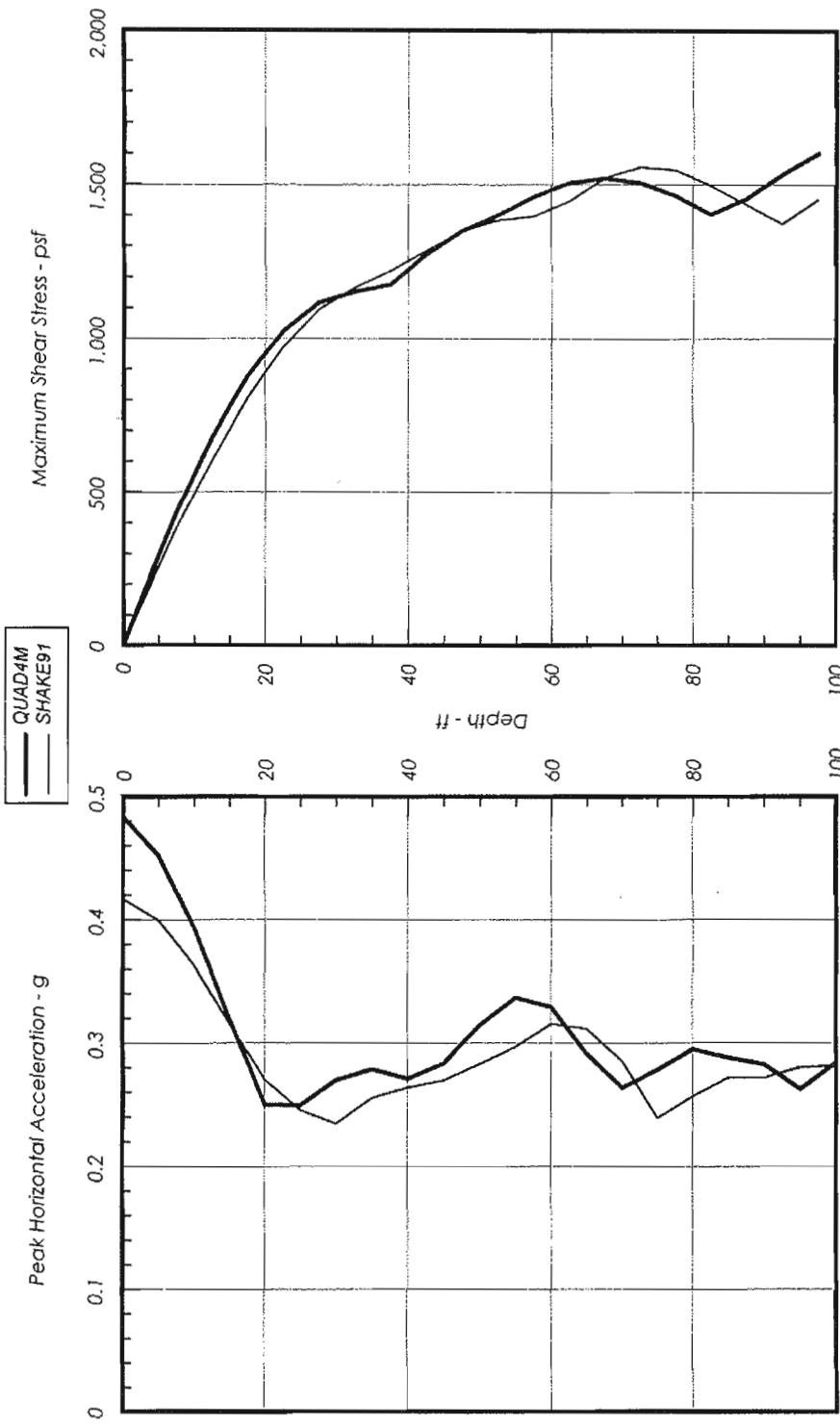


Figure 11
Comparison of Peak Horizontal Accelerations and Maximum Shear Stresses
Computed Using Programs QUAD4M and SHAKE91
for the 100' Dense Sand Layer --
Input Motion: Santa Cruz Record Scaled to 0.3g

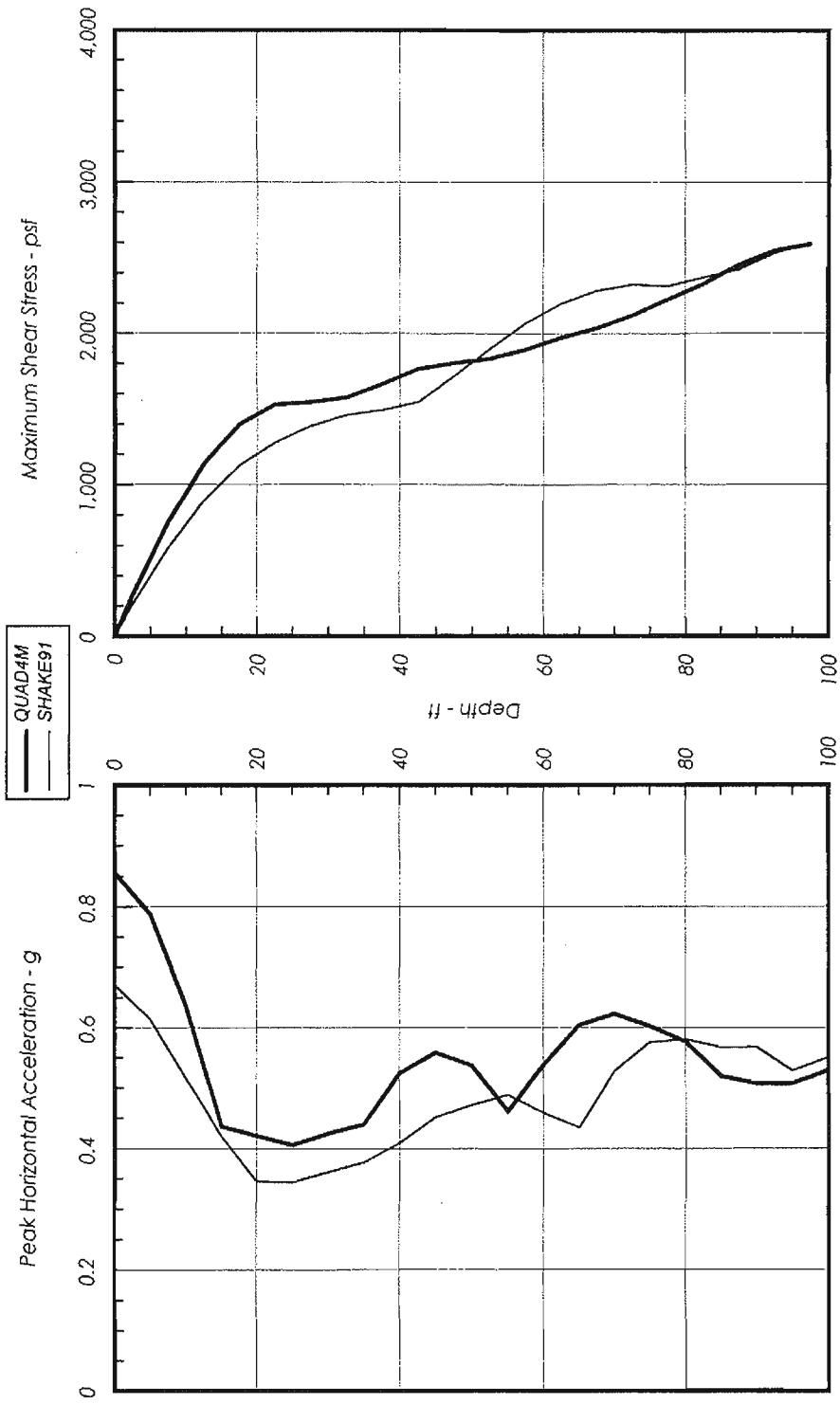


Figure 12
Comparison of Peak Horizontal Accelerations and Maximum Shear Stresses
Computed Using Programs QUAD4M and SHAKE91
for the 100' Dense Sand Layer --
Input Motion: Santa Cruz Record Scaled to 0.6g

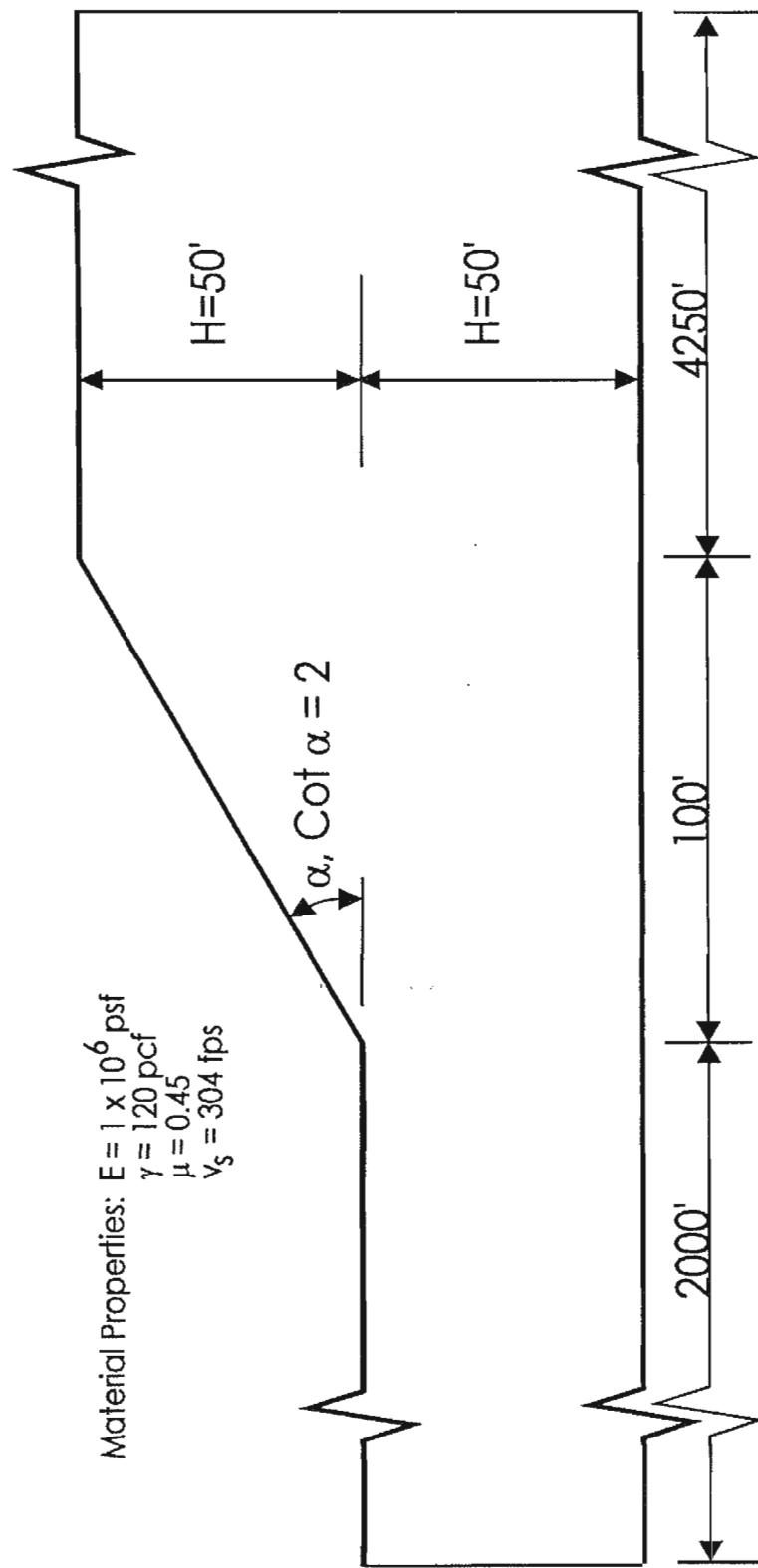


Figure 13
 Geometry and Material Properties for Bank of Sample Problem

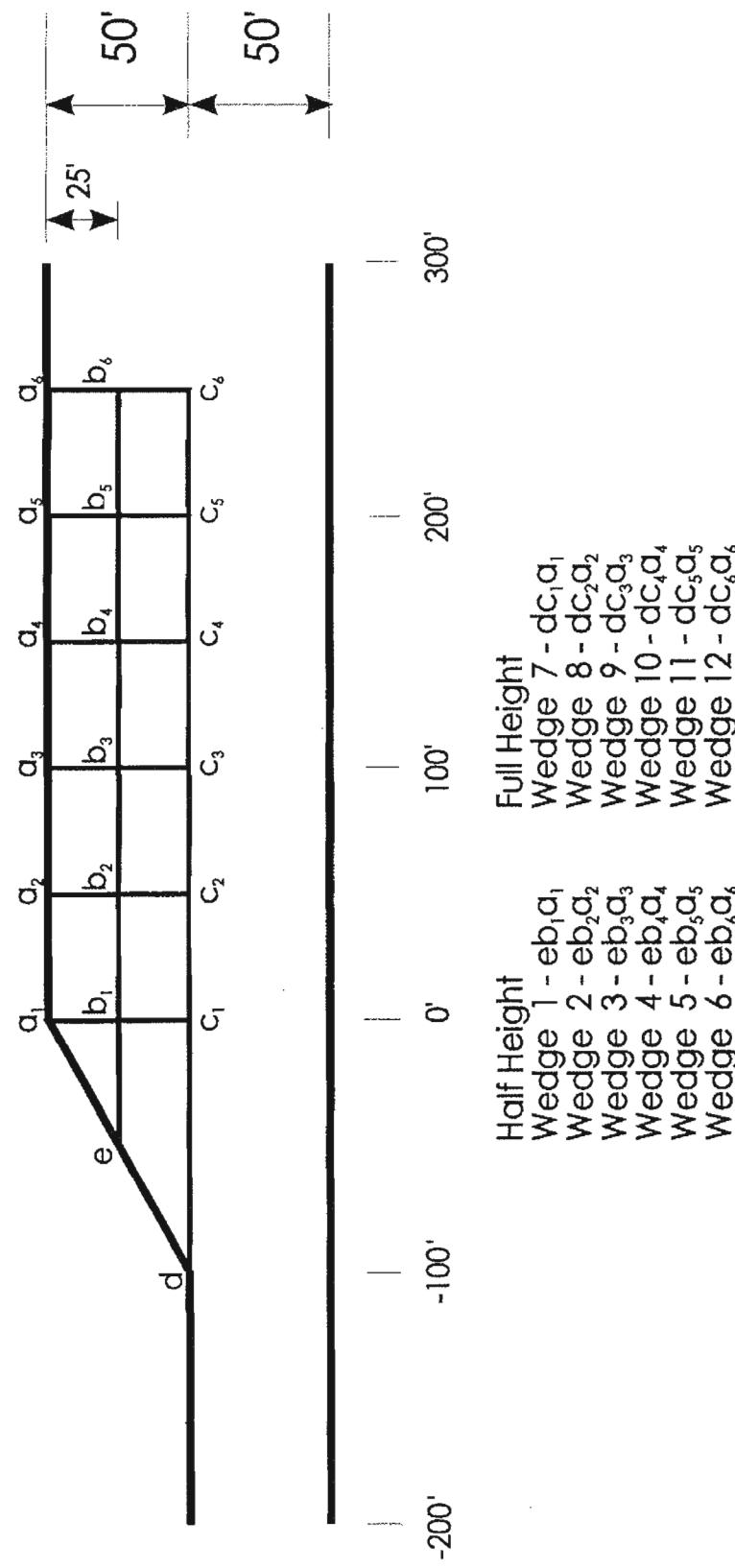


Figure 14

Description of Wedges Used in Seismic Coefficient Evaluation

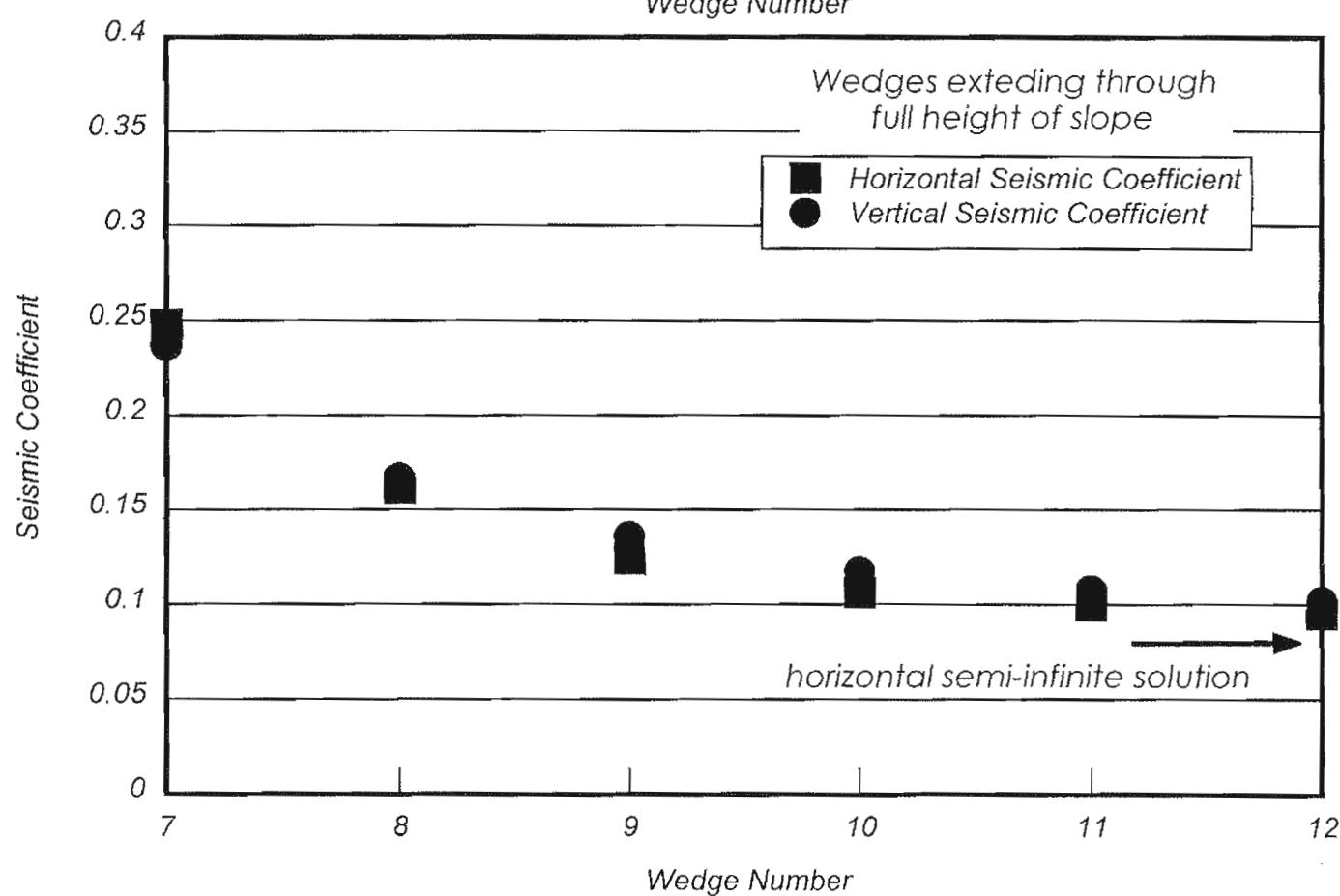
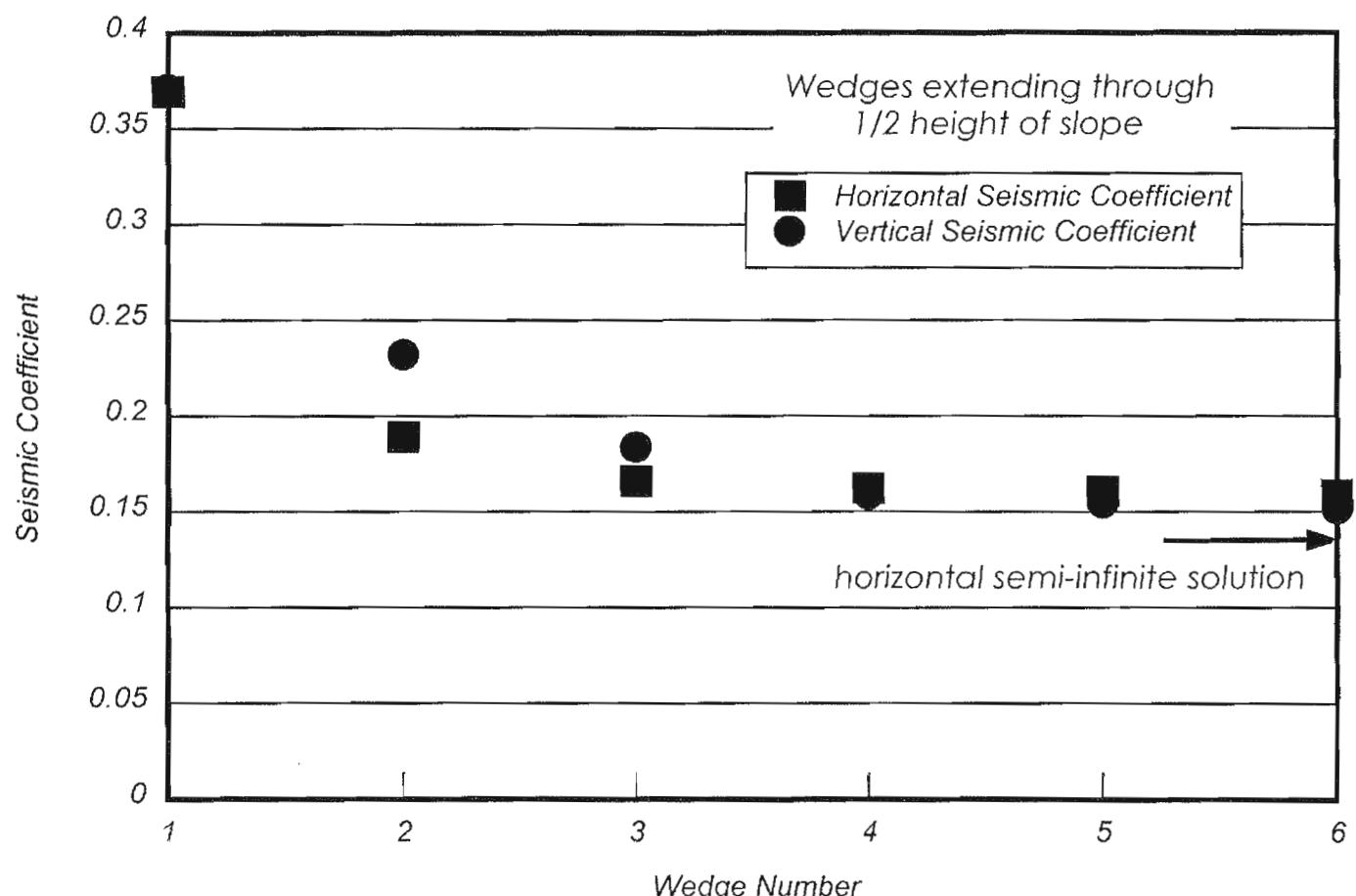


Figure 15
Peak Seismic Coefficients
Example Slope Problem

Appendix A - Program Listing of QUAD4M

Rev. 2003

Program QUAD4M is listed herein. The program has utilized some features of MICROSOFT FORTRAN 5.1 not implemented in the ANSI FORTRAN 77 specifications. These include data structures and dynamic allocation. When using MICROSOFT FORTRAN 5.1, the program can be compiled to run under MICROSOFT WINDOWS 3.1 or DOS, independent of the source listing, but the advantage to using WINDOWS 3.1 is virtually unrestricted access to memory, and in combination with the dynamically allocatable variables, allows large finite element representations to be analyzed on a microcomputer. When using MICROSOFT FORTRAN POWERSTATION 1.0, the program runs many times faster, and has full access to machine memory, but can only be compiled for DOS or Windows NT.

In order to compile under MICROSOFT FORTRAN 5.1, the program was split into 3 modules, QUAD4M1, QUAD4M2, and QUAD4M3. Also, the program must be compiled using the options /AH and /Gt in the compile command. In addition, in order to open enough files at once, a patch may need to be performed on the compiler. This patch and instructions for use are included on the source disk and are available from MICROSOFT.

The program was also compiled to run under MICROSOFT FORTRAN POWERSTATION 1.0, which needs no special considerations for compilation of this program.

The included program was compiled using POWERSTATION and instructions for use are included on the disk.

Acceleration time histories, stress time histories, and seismic coefficient time histories can be requested as described below. The program requires an input data file, earthquake time

history files for the horizontal and vertical components, if needed, and a soil property curve file. The format of these files is described in the beginning comment lines of the program.

PROGRAM QUAD4M

C QUAD4M
U.C. Davis, 1993
by Martin Byrd Hudson,
T.M. Idriss,
and Madsen Beikae

C Modified from QUAD4, 1973.
C by T.M. Idriss, J. Lysmer, R. Huang and H. Bolton Seed

C FINITE ELEMENT ANALYSIS OF PLANE SOIL STRUCTURES SUBJECT TO HORIZONTAL EARTHQUAKE EXCITATION AT THE BASE. THE EQUIVALENT LINEAR METHOD USED INVOLVES DIRECT STEP-BY-STEP INTEGRATION OF EQUATIONS OF MOTION. STRAIN COMPATIBLE RAYLEIGH DAMPING IN ALL ELEMENTS. STRAIN COMPATIBLE SHEAR MODULUS IN ALL ELEMENTS

C ORIGINAL PROGRAM WRITTEN BY T.M. IDRISSS - SUMMER 1968 - USING TRIANGULAR ELEMENTS

C MODIFICATIONS

1. ADDITION OF QUADRILATERAL ELEMENT BY J. LYSMER AND R. HUANG 1971-1972
2. INCORPORATION OF DYNAMIC STORAGE BY N. SERFF - 1972
3. FACILITY TO PUNCH STRESS HISTORIES ADDED BY N. SERFF - 1973
4. ADDITION OF DYNAMIC ALLOCATION.
5. REMOVAL OF SUBROUTINE BANSOL.
6. SUBSTITUTION OF BANEG WITH NEW SUBROUTINE EIGEN.
7. REMOVAL OF USE OF TEMPORARY DISK FILES.
8. BY MARTIN BYRD HUDSON - 1991
9. ADDITION OF COMPLIANT BASE BY MARTIN BYRD HUDSON AND MADSSEN BEIKAE - 1991
10. CHANGE TIME STEPPING METHOD FROM WILSON THETA TO TRAPEZOIDAL RULE.
- USE OF STRUCTURES FOR VARIABLES.
- ELIMINATION OF MOST LINE NUMBERS BY MARTIN BYRD HUDSON - 1991
7. ADDITION OF SEISMIC COEFFICIENTS BY MARTIN BYRD HUDSON AND MADSSEN BEIKAE - 1992
8. CHANGE IN INPUT STRUCTURE, ADDITION OF RESTART CAPABILITY BY MARTIN BYRD HUDSON - 1993
9. CHANGE DAMPING: DAMPING NOW SET AT TWO PERIODS BY MARTIN BYRD HUDSON - 1993

C 10. *Für Seismik Caculator - 2003*

C ** INPUT FILE STRUCTURE:

***** NOTE: COMMENT LINES ARE INTERSESSERED IN THE *****
***** INPUT FILE AS NOTED BELOW *****

C * LINE 1

C TITLE
C * LINE 2 (Comment line)
C * LINE 3
C UNITS SPECIFICATION FOR UNITS USED
C "E" FOR ENGLISH
C "S" FOR SI

C * LINE 4 (Comment line)
C * LINE 5 (5F10.0) DAMPING REDUCTION FACTOR DRF
C PRM FACTOR FOR CONVERTING MAX. STRAIN TO EQUIVALENT UNIFORM STRAIN (USUALLY 0.55 TO 0.75)

C THE REMAINDER OF LINE 5 IS NECESSARY ONLY IF A COMPLIANT BASE IS USED

C ROCPW P WAVE VELOCITY (FEET/SEC OR METER/SEC) OF ROCK
C ROCKVS S. WAVE VELOCITY (FEET/SEC OR METER/SEC) OF ROCK
C ROCKRH AVERAGE ROCK UNIT WEIGHT (pcf OR N/m³)

C * LINE 6 (Comment line)
C * LINE 7 (315) NUMBER OF FINITE ELEMENTS
C NELM
C NDPT TOTAL NUMBER OF NODAL POINTS
C NSLP NUMBER OF SURFACES FOR SEISMIC COEFFICIENT ANALYSIS

C * LINE 8 (Comment line)
C * LINE 9 COMPUTATION SWITCHES (915)
C KOMAX NUMBER OF TIME STEPS IN INPUT EARTHQUAKE RECORD
C KGEQ NUMBER OF LAST TIME STEP FOR LAST ITERATION
C N1EQ NUMBER OF FIRST TIME STEP FOR LAST ITERATION
C N2EQ NUMBER OF FIRST TIME STEP FOR FIRST ITERATIONS
C N3EQ NUMBER OF LAST TIME STEP FOR FIRST ITERATIONS
C NUBR NUMBER OF ITERATIONS ON SOIL PROPERTIES
C KV SWITCH FOR READING VERTICAL COMPONENT OF EQ. RECORD (2 IF VERT. COMPONENT TO BE READ, 1 IF NO VERT. COMPONENT TO BE READ)
C KSAV SWITCH FOR SAVING STATE OF SYSTEM AT LAST STEP OF LAST ITERATION FOR RESTART
C (0: DON'T SAVE, 1: SAVE)

C * LINE 10 (Comment line)
C * LINE 11 - EARTHQUAKE FILE DESCRIPTORS (5F10.0, 4I5, F10.0)
C DTEQ TIME STEP OF INPUT MOTION (SECONDS)
C EOMUL(1) SCALING FACTOR FOR HORIZ. INPUT MOTION
C EOMUL(2) SCALING FACTOR FOR VERT. INPUT MOTION
C UGMAX(1) SPECIFIED MAXIMUM HORIZONTAL ACCELERATION TO WHICH THE INPUT MOTION WILL BE SCALLED. (G)
C UGMAX(2) SPECIFIED MAXIMUM VERTICAL ACCELERATION TO WHICH THE INPUT MOTION WILL BE SCALLED. (G)

C NOTE: EITHER EOMUL OR UGMAX IS USED BY THE PROGRAM TO OBTAIN THE SCALING FACTOR FOR THE INPUT MOTION. THEREFORE, FOR EACH COMPONENT, EITHER EOMUL OR UGMAX MUST BE 0

C HORX NUMBER OF HEADER LINES IN THE HORIZONTAL INPUT TIME HISTORY
C HORY NUMBER OF DATA POINTS PER LINE IN THE HORIZONTAL INPUT TIME HISTORY
C NPLX NUMBER OF DATA POINTS PER LINE IN THE VERTICAL INPUT TIME HISTORY. 0 INDICATES 'NOT SPECIFIED'. NUMBER OF DATA POINTS PER LINE IN THE VERTICAL INPUT TIME HISTORY. 0 INDICATES 'NOT SPECIFIED'.

MODULE: QUAD4M1

Page A - 4

```

C PRINPUT          PERIOD CORRESPONDING TO MAXIMUM SPECTRAL ACCELERATION OF
C                               THE HORIZONTAL INPUT MOTION (SECONDS).
C * LINE 12 (Comment line)
C * LINE 13: (A) EARTHH          NAME OF FILE WITH HORIZONTAL INPUT TIME HISTORY
C * LINE 14: (A) EQINPMT(1)      EARTHQUAKE FILE INPUT FORMAT. ENTER '*' IN COLUMN 1 IF
C                               FREE FORMAT IS DESIRED, OR REGULAR FORMAT SPECIFIER IN
C                               PARENTHESES. USED FOR HORIZONTAL INPUT TIME HISTORY.
C                               NOTE: WHEN USING FORMAT SPECIFIER,
C                               POINTS PER LINE MUST BE INCLUDED. e.g. (8F9.6).
C
C * LINE 15: (A) (ONLY USE IF KV=2) EARTHQV         NAME OF FILE WITH VERTICAL INPUT TIME HISTORY
C
C * LINE 16: (A) (ONLY USE IF KV=2) EQINPMT(2)      EARTHQUAKE FILE INPUT FORMAT. ENTER '*' IN COLUMN 1 IF
C                               FREE FORMAT IS DESIRED, OR REGULAR FORMAT SPECIFIER IN
C                               PARENTHESES. USED FOR VERTICAL INPUT TIME HISTORY.
C                               SEE NOTE FOR EQINPMT(1) ABOVE.
C
C * LINE 17 (Comment line)
C * LINE 18: (315) (ALL 3 VARIABLES 0 OR 1)           SWICH TO READ STRESS OUTPUT FILE DESCRIPTORS: 1=READ
C                               SWICH TO READ ACCELERATION OUTPUT FILE DESCRIPTORS: 1=READ
C                               SWICH TO READ SEISMIC COEFFICIENT OUTPUT FILE DESCRIPTORS: 1=READ
C
C * STRESS OUTPUT FILE DESCRIPTORS (ONLY USE LINES 19-22 IF SOUT=1)
C
C LINE 19 (Comment line)
C
C LINE 20: (A) SHSTMT          OUTPUT FORMAT:
C                               EITHER
C                               'COMBINED'
C                               WHICH COMBINES ALL REQUESTED STRESS HISTORIES INTO
C                               ONE FILE SUCH THAT EACH STRESS HISTORY IS OUTPUT AS A
C                               SEPARATE COLUMN WITHIN THAT FILE
C                               OR
C                               'MULTIPLE'
C                               WHICH PRODUCES EACH STRESS HISTORY AS A SEPARATE
C                               FILE WITH FORMAT B8F 1.
C
C LINE 21: (A) SPFILEOUT        OUTPUT FILE NAME ROOT (8 CHARACTERS MAX IF COMBINED
C                               OUTPUT, OR 6 CHARACTERS MAX IF MULTIPLE OUTPUT)
C
C LINE 22: (A) SSUFFIX          OUTPUT FILE NAME SUFFIX (3 CHARACTERS MAX)
C
C * ACCELERATION OUTPUT FILE DESCRIPTORS (ONLY USE LINES 23-26 IF ADOT=1)
C
C LINE 23 (Comment line)
C
C LINE 24: (A) AMSTMT          OUTPUT FORMAT:
C                               EITHER
C                               'COMBINED'
C                               WHICH COMBINES ALL REQUESTED ACCELERATION HISTORIES
C                               INTO ONE FILE SUCH THAT EACH STRESS HISTORY IS
C                               OUTPUT AS A SEPARATE COLUMN WITHIN THAT FILE
C                               OR
C                               'MULTIPLE'

WHICH PRODUCES EACH ACCEL HISTORY AS A SEPARATE FILE
WITH FORMAT 8F9. 6

LINE 25: (A) AFILEOUT        OUTPUT FILE NAME ROOT (8 CHARACTERS MAX IF COMBINED
OUTPUT, OR 6 CHARACTERS MAX IF MULTIPLE OUTPUT)
LINE 26: (A) ASUFFIX          OUTPUT FILE NAME SUFFIX (3 CHARACTERS MAX)

* SEISMIC COEFFICIENT OUTPUT FILE DESCRIPTORS (ONLY USE LINES 27-30 IF KOUT=1)
LINE 27 (Comment line)
LINE 28: (A) KSIFMT          OUTPUT FORMAT:
C EITHER
C   'COMBINED'
C   WHICH COMBINES ALL REQUESTED ACCELERATION HISTORIES
C   INTO ONE FILE SUCH THAT EACH STRESS HISTORY IS OUTPUT
C   AS A SEPARATE COLUMN WITHIN THAT FILE
C   OR
C   'MULTIPLE'
C   WHICH PRODUCES EACH ACCEL HISTORY AS A SEPARATE FILE
WITH FORMAT 8F9. 6

LINE 29: (A) KFILEOUT        OUTPUT FILE NAME ROOT (8 CHARACTERS MAX IF COMBINED
OUTPUT, OR 6 CHARACTERS MAX IF MULTIPLE OUTPUT)
LINE 30: (A) KSUFFIX          OUTPUT FILE NAME SUFFIX (3 CHARACTERS MAX)

* RESTART FILE NAME DESCRIPTOR (ONLY USE LINES 31-32 IF KSAV=1)
LINE 31 (Comment line)
LINE 32: FILE NAME (EXCLUDING PATH) FOR SAVING FINAL STATE OF SOU FOR RESTART (A)
SAVFILE

* SEISMIC COEFFICIENT LINES (ONLY USE LINES 33-38 IF NSLP>0; REPEAT NSLP TIMES)
LINE 33 (Comment line)
LINE 34: (215)
NSEG(1)  NUMBER OF NODES INTERSECTED BY SURFACE 1
ESEG(1)  NUMBER OF ELEMENTS WITHIN SURFACE 1
LINE 35 (Comment line)
LINE 36: (1515) (must be repeated if NSEG(1)>15)
NSEG(1,J) NODE J INTERSECTED BY SURFACE 1 (NSEG(1) NODES)
LINE 37 (Comment line)
LINE 38: (1515) (must be repeated if ESEG(1)>15)
ELSEG(1,J) ELEMENT J WITHIN SURFACE 1 (ESEG(1) ELEMENTS)

* LINE 39 (Comment line)
* LINE 40 - ELEMENT LINES (REPEAT LINE 40 NELM TIMES) (615, SF10, 0, 15)
C ELEMENT NUMBER
C EL(N), NODE(1) 1ST NODAL POINT NUMBER
C EL(N), NODE(2) 2ND NODAL POINT NUMBER
C EL(N), NODE(3) 3RD NODAL POINT NUMBER
C EL(N), NODE(4) 4TH NODAL POINT NUMBER
C NOTE 1. SPECIFY NODAL POINTS IN COUNTER-CLOCKWISE
C DIRECTION
C
C 2. EL(N), NODE(3) = EL(N), NODE(4) PRODUCES
C
C

```

MODULUS E: QIADAM

PROGRAM: QIADD4M

```

CHARACTER EQINP#*20,EARTHNY#*72,EARTHNY#*72
INTEGER SOUT,AOUT,KOUT,HORX,HORY,ESSEG,EL,SEG,ESEGMAX
CHARACTER #2 FILEN,FILEOUT,10PROP#*75
RECORD *EL ALLOCATABLE () :
RECORD *HOLE#(NO) ALLOCATABLE () :
DIMENSION ZMASZ ALLOCATABLE () :
SMASS ALLOCATABLE () :
& R ALLOCATABLE () :
& AT ALLOCATABLE () :
& BT ALLOCATABLE () :
& X1 ALLOCATABLE () :
& X2 ALLOCATABLE () :
& DSMAX ALLOCATABLE () :
& TIME1 ALLOCATABLE () :
& NBC ALLOCATABLE () :
& U2G ALLOCATABLE () :
& X ALLOCATABLE () :
& ST ALLOCATABLE () :
& DS ALLOCATABLE () :
& 10PROP ALLOCATABLE () :
& SDATA ALLOCATABLE () :
& DDATA ALLOCATABLE () :
& NSGP ALLOCATABLE () :
& NSEG ALLOCATABLE () :
& ESEG ALLOCATABLE () :
& ELSEG ALLOCATABLE () :

INPUT DATA
WRITE(*,9000)
WRITE(*,9001) FILEIN
READ(*,'(A)') FILEIN
OPEN (5,FILE=FILEIN,STATUS='OLD')
WRITE(*,9002)
READ(*,'(A)') DATAIN
OPEN (7,FILE=DATAIN,STATUS='OLD')
WRITR(*,'(A,A)') Enter Output directory (with endi
& nes) . . . (.) for current dir):
WRITE(*,9000)
READ(*,'(A)') DIROUT
WRITE(*,'(A)') . . . output File Name (without director
READ(*,'(A/A)') TITLE,UNITS
IF (UNITS.EQ.'E') THEN
  GRAV = 32.2
ELSE IF (UNITS.EQ.'S') THEN
  GRAV = 9.81
ELSE
  WRITE(*,'(A)') 'INVALID UNITS SPECIFICATION'
END IF

```

MODULE E: QUAD4M1

PROGRAM: QUAD4M

MODUL E: QUAD4M1

PROGRAM: QUAD4M

MODUL E: QUADAM

PROGRAM: QUAD4M

```

C Marks nodes according to boundary conditions and echoes boundary
C conditions
C Written for QUAD4M
C U.C. Davis, 1983
C Modified from QUADA, 1973.
C by Martin Byrd Hudson, I.M. Idriss, Madsen Beikae
C by I.M. Idriss, J. Lymer, R. Hwang and H. Bolton Seed

STRUCTURE /NODE/
REAL,XORD,YORD,TRIBLEN,BETAS,BETAP,X2I(2),X1I(2),Y1I(2)
INTEGER,BC,OUT
END STRUCTURE
COMMON//CONST1/TITLE,DRF,NODEP,NB1,NBP,HELM,UNITS,GRAV,NSLP,NF
CHARACTER TITLE*72,UNITS*1
COMMON//CONST2/KGMAX,KGEQ,NIEQ,NEQ2,N3EQ,NUMB,KV,KSAV
COMMON//CONST3/DTEQ,EPOL(2),PRM,UQMAX(2),EQNH,EQNV,UG(2),PRINTPUT
CHARACTER*72,EGNH,EGNV
RECORD /NODE/NODC*/
DIMENSION NODC(*)

K = NO, BC + 1
IF (NO,BC,0) THEN ! free Node
  WRITE(6,152) M, NO, XORD, NO, YORD, NO, TRIBLEN
ELSE IF (NO,BC,1) THEN ! Cannot move in x-dir
  WRITE(6,1012) M, NO, XORD, NO, YORD, NO, TRIBLEN
N=N+1
NDC(N)=2**M-1
ELSE IF (NO,BC,2) THEN ! Cannot move in y-dir
  WRITE(6,1022) M, NO, XORD, NO, YORD, NO, TRIBLEN
N=N+1
NDC(N)=2**M-1
ELSE IF (NO,BC,3) THEN ! Can't move in x or y
  WRITE(6,1002) M, NO, XORD, NO, YORD, NO, TRIBLEN
  IF (M, GE, NB1) RETURN
N=N+1
NDC(N)=2**M-1
ELSE IF (NO,BC,4) THEN ! trans. base node
  WRITE(6,1003) M, NO, XORD, NO, YORD, NO, TRIBLEN
  IF (KV, EQ, 1) THEN
    N=N+1
    NDC(N)=2**M-1
  END IF
ELSE ! (6,*)
  WRITE(6,*)
  RETURN
END IF

FORMAT (18.3F15.3, 'X FIXED')
FORMAT (18.3F15.3, 'Y FIXED')
FORMAT (18.3F15.3, 'CANNOT MOVE IN BOTH X AND Y DIRECTIONS')
FORMAT (18.3F15.3, 'CANNOT MOVE IN BOTH X AND Y DIRECTIONS')
FORMAT (18.3F15.3, 'COMPLIANT BASE NODE')
FORMAT (18.3F15.3, 'CANNOT MOVE IN X DIRECTION')
FORMAT (18.3F15.3, 'CANNOT MOVE IN Y DIRECTION')

SUBROUTINE ECHO(EL,NO,ROCKBETAS,ROCKBETAS,IPROP,ODATA,ODATA,
  DATA,NUMPOINTS,NBC,NGSEG,ESEG,
  ELSEG,MAXW,MAXL)
  E
  C **** SUBROUTINE ECHO(EL,NO,ROCKBETAS,ROCKBETAS,IPROP,ODATA,ODATA,
  DATA,NUMPOINTS,NBC,NGSEG,ESEG,
  ELSEG,MAXW,MAXL)
  C
  C Echoes input data to screen and sets up file names
  C Written for QUAD4M
  C
  C by Martin Byrd Hudson, I.M. Idriss, Madsen Beikae
  C
  C Modified from QUADA. 1973.
  C
  C by I.M. Idriss, J. Lymer, R. Hwang and H. Bolton Seed
  C
  C STRUCTURE /ELEMENT/
  REAL,GRAV,G,E,EN,AREA,XL,TIME2,STG(3),SIGMAX(3),EPSMAX,PO,DENS
  INTEGER, NODE(4), TYPE, LSTR
  END STRUCTURE
  STRUCTURE /NODE/
  REAL,XORD,YORD,TRIBLEN,BETAS,BETAP,X2I(2),X1I(2),Y1I(2)
  END STRUCTURE
  COMMON//CONST1/TITLE,DRF,NODEP,NB1,NBP,HELM,UNITS,GRAV,NSLP,NF
  CHARACTER TITLE*72,UNITS*1
  COMMON//CONST2/KGMAX,KGEQ,NIEQ,NEQ2,N3EQ,NUMB,KV,KSAV
  CHARACTER*72,EGNH,EGNV
  COMMON//CONST3/DTEQ,EPOL(2),PRM,UQMAX(2),EQNH,EQNV,UG(2),PRINTPUT
  CHARACTER SHISTMT*1,KHISTMT*1,SFILEMT*1,AFILEMT*1
  KFILEOUT*8,SSUFFIX*3,ASUFFIX*3,KSUFFIX*3,DROUT*72,
  SUFFIX*12,DATAIN*72
  COMMON//CONST5/HARY,EARTHY,EARTHOM,EGINPMFT(2),ROCKVP,ROCKVS,
  ROCKRH,RIHSISTR,RIHSTAC,RLPX,MLPY
  CHARACTER EGINPMFT*20,EARTHOM*72,EARTHON*72
  INTEGER,HARY,EGEN,ELSEG,ELSEG
  CHARACTER FILENAME*12,TDROP*75
  DIMENSION TDROP(2,*),ODATA(2,NUMPROPS,*),ODATA(NUMPROPS,*),
  DATA(NUMPROPS,*),NUMPOINT(S(2,*),NBC(*),NGSEG(*)),
  ELSEG(NSLP,*),ESEG(*),ELSEG(NSLP,*)
  RECORD /ELEMENT/EL(*)
  RECORD /NODE/ NO(*)
  WRITE(*,BZ)

```

MODULE F: QUAD4M1

PROGRAM: QUAD4M

MODUL E: OIIADAM 1

PROGRAM: QUADAM

PROGRAM: QUAD4M

Page A - 11

```

        END IF
        END DO
        WRITE(40,'(/A,\')')
      ELSE
        I=0
        DO N=1,NSLP/2
          IF (N<N) QUT.EQ.1.OR.N>(N).QUT.EQ.3) THEN
            I=I+1
            FILENAME=FILEOUT((LEN(TRIM(FILEOUT)))/'0')/
              CHAR(47+1)//'/'//ASUFFIX
            OPEN(1+39,FILE=DIROUT((LEN(TRIM(OROUT)))/FILENAME))
            WRITE(1+39,'(A/A,A,F10.3,A)') TITLE,
              Acceleration History ('S'),
              Time Step = ' , DTEQ, ' Sec'
            WRITE(1+39,'(A,14,A)') Node , N, 'X',
              WRITE(*,1132) NODE , N, 'Y DIR', FILENAME
            WRITE(6,1132) NODE , N, 'Y DIR', FILENAME
          END IF
          IF (N>N) QUT.EQ.2.OR.(N>N).QUT.EQ.3) THEN
            I=I+1
            FILENAME=FILEOUT((LEN(TRIM(FILEOUT)))/'0')/
              CHAR(47+1)//'/'//ASUFFIX
            OPEN(1+39,FILE=DIROUT((LEN(TRIM(OROUT)))/FILENAME))
            WRITE(1+39,'(A/A,A,F10.3,A)') TITLE,
              Acceleration History ('S'),
              Time Step = ' , DTEQ, ' Sec'
            WRITE(1+39,'(A,14,A)') Node , N, 'Y',
              WRITE(*,1132) NODE , N, 'Y DIR', FILENAME
            WRITE(6,1132) NODE , N, 'Y DIR', FILENAME
          END IF
        END DO
      END IF
    END IF
  C Open seismic coefficient output files
  C
  IF (NSLP.GT.0) THEN
    IF (RHISTMT.EQ.'C' .OR. RHISTEM.EQ.'C') THEN
      FILENAME=FILEOUT((LEN(TRIM(FILEOUT)))/'.'//KSUFFIX)
      OPEN(50,FILE=DIROUT((LEN(TRIM(OROUT)))/FILENAME))
      WRITE(50,'(A/A)') TITLE
      WRITE(50,'(A,A,Y)') ' Seismic Coefficient Surface Histories'
      IF (KV.EQ.1) THEN
        WRITE(50,'(A,Y)') ' Time.sec'
        DO N=1,NSLP
          WRITE(50,'(A8,12,V)') ' Block ' , N
          WRITE(*,1132) SURFACE , N, 'X DIR', FILENAME
          WRITE(6,1132) SURFACE , N, 'Y DIR', FILENAME
        END DO
      ELSE
        WRITE(50,'(A,V)') ' Time.sec'
        DO N=1,NSLP
          WRITE(50,'(A8,12,10H,V)') ' Block ' , N
        END DO
      END IF
    END IF
    IF (KV.GT.0) THEN
      FILENAME=FILEOUT((LEN(TRIM(FILEOUT)))/'0')/
        CHAR(47+1)//'/'//KSUFFIX
      OPEN(1+49,FILE=DIROUT((LEN(TRIM(OROUT)))/FILENAME))
      WRITE(1+49,'(A/A,A,F10.3,A)') TITLE,
      ' Seismic Coefficient Surface History',
      ' Time Step = ' , DTEQ, ' Sec'
      WRITE(1+49,'(A10,13,V)') ' Surface ' , N
      WRITE(*,1132) SURFACE , N, 'X DIR', FILENAME
      WRITE(6,1132) SURFACE , N, 'Y DIR', FILENAME
    ELSE
      WRITE(1+49,'(A13)') ' Y-dir '
      WRITE(*,1132) SURFACE , N, 'Y DIR', FILENAME
      WRITE(*,1132) SURFACE , N, 'X DIR', FILENAME
    END IF
  END IF
END IF

```

MODULE: QUAD4M1

Page A - 12

```

      WRITE(6,1132) 'SURFACE',N,'DIR',FILENAME
      END IF
    END DO
    END IF
    END IF
    END IF
    IF (KSAV.EQ.1) THEN
      WRITE(*,1133) SAVEFILE
      WRITE (6,1133) SAVEFILE
    END IF
    WRITE (6,1009) DATAIN
    DO I=1, NMPROPS
      WRITE(6,1004) 1,1DPROP(1,1),1DPROP(2,1)
      WRITE(6,1005)
      H=MPOIN(NUMPOINTS(1,1),NUMPOINTS(2,1))
      DO J=1,M
        IF (J.LE.NUMPOINTS(1,1).AND.J.LE.NUMPOINTS(2,1)) THEN
          WRITE(6,1006) SDATA(1,J),SDATA(1,J),SDATA(2,I,J),DDATA(1,J)
        ELSE IF (J.LE.NUMPOINTS(1,1)) THEN
          WRITE(6,1007) SDATA(1,J),SDATA(1,J),DDATA(1,J)
        ELSE
          WRITE(6,1008) SDATA(2,I,J),DDATA(1,J)
        END IF
      END DO
      IF (UNITS.EQ.'E') THEN
        IF (ROCKRH.NE.0) THEN
          WRITE (6,1010) ROCKRH*GRAV,'PCF',ROCKBETAS/ROCKRH,'FT/SEC'
        ELSE
          WRITE (6,1010) 0.,'PCF',0.,'FT/SEC',0.,'FT/SEC'
        END IF
      ELSE
        IF (ROCKRH.NE.0) THEN
          WRITE (6,1010) ROCKRH*GRAV,'NM^3',ROCKBETAS/ROCKRH,
     & 'M/SEC',ROCKBETAP/ROCKRH,'M/SEC'
        ELSE
          WRITE (6,1010) 0.,'NM^3',0.,'M/SEC',0.,'M/SEC'
        END IF
        IF (NSLP.GT.0) THEN
          WRITE(6,92)
        END IF
        DO N=1,NSLP
          WRITE(6,93) N,NESEG(N),ESEG(N)
          WRITE(6,94) (NESEG(N,K),K=1,NSEG(N))
          WRITE(6,95) (ESEG(N,K),K=1,ESEG(N))
        END DO
        IF (UNITS.EQ.'E') THEN
          WRITE(6,122)
        ELSE
          WRITE(6,123)
        END IF
      END IF
    END DO
    END IF
    WRITE(6,1142)
  END IF
  C Boundary condition for dynamic computation
  C
  N=0
  GINIT=0.
  DO M=1,NDPT
    CALL BOUNDARY(NO(M),M,N,NBC)
    GINIT=MAX(GINIT,ABS(NO(M)*X2(1)),ABS(NO(M)*X2(2)),
     & ABS(NO(M)*X1(1)),ABS(NO(M)*X1(2)))
  END DO
  IF (N .NE. NBP) THEN
    WRITE(*,1098)
    WRITE(6,1098)
    STOP
  END IF
  C Initial Conditions Echo
  C
  IF (GINIT.GT.0) THEN
    WRITE(*,99)
    WRITE(6,1099)
    DD=M+NDPT
    WRITE(6,1100) M,NO(M),X2(1),NO(M),X1(1),NO(M),X1(2),
     & NO(M),X2(2),NO(M),X1(2),NO(M),X1(1)
  END DO
  END IF
  RETURN
  82 FORMAT(//1'1')
  92 FORMAT(//1'1'/SEISMIC COEFFICIENT SURFACE DATA: ')
  93 FORMAT(//1'1/BLOCK: ',13X,'NSEG: ',14,X,'ESEG: ',14,'/
   & SURFACE PASSES THROUGH NODES: ')
  94 FORMAT(15$)
  95 FORMAT(' SURFACE INCLUDES ELEMENTS: ')
  99 FORMAT(//1'1'/INITIAL CONDITIONS USED AT THE NODES: ',/1'1',
   & 8X,'X2H',8X,'X1H',8X,'X1V',8X,'X2V',8X,'X1IV',8X,'X1V')
  102 FORMAT(
   & 20X,'NO. OF ELEMENTS = ',16/
   & 16X,'NO. OF NODEAL POINTS = ',16/
   & 17X,'DEGREES OF FREEDOM = ',16/
   & 21X,'HALF-BANDWIDTH = ',16/
   & 16X,'CONTROLLING ELEMENT = ',16/
   & 10X,'NO. OF FIXED BOUNDARY CONDS. = ',16/
   & 18X,'NO. OF ITERATIONS = ',16/
   & 6X,'TOTAL EQ. POINTS READ (KSMAX) = ',16/
   & 1X,'LAST EQ. PTS. USED (NLEQ TO KGEQ) = ',216/
   & 3X,'INT. EQ. PTS. USED (NSEQ TO NSEQ) = ',216/
   & 11X,'TIME INTERVAL OF RECORDS = ',1F8.4,' SECONDS '

```

MODULUS: Q11AD4M1

PROGRAM: QUAD4M

PROGRAM: QUAD4M1

Page A - 14

```

C the horizontal component and vertical component (if any) of
C the earthquake, and the soil data.
C
C Written for QUAD4M
C U.C. Davis, 1993
C by Martin Byrd Hudson, I.M. Idriss, Mohsen Beikae
C Modified from QUAD4, 1973.
C by I.M. Idriss, J. Lyssner, R. Hwang and H. Bolton Seed
C
C STRUCTURE /ELEMENT/
REAL GRX,G,E,EN,AREA,XL,TIME2,SIG(3),SIGMAX(3),EPSMAX,PO,DENS
INTEGER NODE(4),TYPE,LSR
END STRUCTURE
STRUCTURE /NODE/
REAL XRD,YRD,TRIBLEN,BETAS,BETAP,X21(2),X11(2),X1(2)
END STRUCTURE
COMMON/CONST1/TITLE,DR,NP1,NP2,NELM,UNITS,GRAN,NSLP,NF
CHARACTER TITLE*72,UNITS*
COMMON/CONST2/KGMAX,KGEQ,N1EQ,M2EQ,M3EQ,NSEQ,NUMB,KV,KSAN
COMMON/CONST3/DETEQ,EDOM,(2),PRM,UGMAX(2),EQNH,EQW,U(2),PRINPUT
CHARACTER*(72) ENH,ENW
COMMON/CONST4/SHISTEMT,AHISTEMT,KHISTEMT,SFILEOUT,AFILEOUT,
KFILEOUT,SSUFFIX,X_SUFFIX,DROUT,SAVEFILE,
DATAIN
CHARACTER SHISTEMT*1,AHISTEMT*1,KHISTEMT*1,SFILEOUT*8,AFILEOUT*8,
KFILEOUT*8,SSUFFIX*3,X_SUFFIX*3,CSUFFIX*X*3,DROUT*72,
SAVEFILE*72,DATAIN*72
COMMON/CONST5/HORI,HORV,EARTHQW,EARTHQH,EARTHQZ,
&CHARACTER EQUIMENT(2),ROCKNP,ROCKVS,
&RECORD /ELEMENT/ELEM(*)
RECORD /NODE/NO(*)*
DIMENSION U2G(10,KGMAX,*),TOPROP(2,*),SDATA(2,NUMPROPS,*),
&GDATA(NUMPROPS,*),DATA(NUMPROPS,*),NMPOLNTS(2,*),
NSEG(*),NOSEG(NSLP,*),ESEG(*),ESEG(NSLP,*)
C
IF (KV,EQ,2) THEN
  DO I=0,KGMAX
    DO J=1,2
      U2G(I,J)=0.
    END DO
  END DO
ELSE
  DO I=0,KGMAX
    UFS(I,1)=0.
  END DO
END IF
C Read Seismic Coefficient Surface Information
C

```

```

IF (NSLP,GT,0) THEN
  DO J=1,NSLP
    READ (5,'(1A*)')
    READ (5,'(2I5)') NSEG(J),ESEG(J)
    READ (5,'(1A*)')
    READ (5,'(1A5)') (NSEG(J,I),I=1,NSEG(J))
    READ (5,'(1A5)') READ (5,'(1A5)') (ESEG(J,I),I=1,ESEG(J))
  END DO
END IF
C Read Element Information
C
READ (5,'(1A*)')
DO J=1,NELM
  READ (5,'(6I5,5F10.0,1E5)') N,(EL(N),NODE(J),L=1,4),EL(N),TYPE,
  EL(N),DENS,EL(N),PO,EL(N),GRX,EL(N),G,
  EL(N),XL,EL(N),LSTR
  C Reorder nodes if not according to specifications
  DO I=1,2
    DO K=1,4
      IF (EL(N),NODE(I),EL(N),NODE(K)) THEN
        WRITE(6,*) 'Nodes reordered for element: ',N
        IF (I,EQ,1,AND,K,EQ,2) THEN
          EL(N),NODE(1)=EL(N),NODE(3)
          EL(N),NODE(3)=EL(N),NODE(2)
        ELSE IF (I,EQ,2,AND,K,EQ,3) THEN
          EL(N),NODE(2)=EL(N),NODE(4)
          EL(N),NODE(4)=EL(N),NODE(3)
        ELSE IF (I,EQ,3,AND,K,EQ,4) THEN
          EL(N),NODE(3)=EL(N),NODE(1)
          EL(N),NODE(1)=EL(N),NODE(4)
        ELSE
          PRINT*, 'Element ',N, ' has improperly repeating nodes'
        END IF
      END DO
    END DO
  END IF
  C Read Node Information
  C
  READ (5,'(1A*)')
  DO N=1,NODEP
    READ (5,'(15,2F10.0,2I5,6F13,0)') I,NO(I),XORD,NO(C),YORD,
    NO(I),BC,NO(D),OUT,
    NO(I),X2(I,1),NO(I),X1(I,1),NO(I),X1(I,2),
    NO(I),X2(I,2),NO(I),X1(I,2),NO(I),X1(I,2)
  END DO
END IF
C

```

MODULE: QUAD4M1

Page A - 15

```

C Verify that no seismic coefficient surfaces go through base
C
C IF (NSLP.GT.0) THEN
DO J=1,NSLP
  DO I=1,NESEG(J)
    IF (NO(EL(K),NODE(M)).EQ.4) THEN
      WRITE (*,'(A,14,A,A)') ' NODE ',NESEG(J,1),' IS NOT A',
      & ' FREE NODE AND SHOULD NOT BE USED IN A SEISMIC COEFF SURFACE!'
      STOP
    END IF
  END DO
END IF
C Find area of elements
C
DO J=1,NELM
  DO I=1,4
    X1=NO(EL(J),NODE(I,1),XORD
    Y1=NO(EL(J),NODE(I,2),YORD
    X3=NO(EL(J),NODE(I,3),XORD
    Y3=NO(EL(J),NODE(I,4),YORD
    X4=NO(EL(J),NODE(I,1),XORD
    Y4=NO(EL(J),NODE(I,2),YORD
    T1=NO(EL(J),NODE(I,1),XORD
    Y2=NO(EL(J),NODE(I,3),YORD
    V2=NO(EL(J),NODE(I,4),YORD
    EL(J).AREA=(Y2-Y4)*(X1*X2-X3*X4)
    EL(J).AREA=0.5*EL(J).AREA
    IF (EL(J).AREA.LE.0.) THEN
      PRINT *, ' ELEMENT ',J,' HAS 0 OR NEGATIVE AREA.
      WRITE (*,'(A,14,A,A)') ' ABORTED'
      STOP
    END IF
  END DO
C Find Tributary Length of base nodes
C
DO L=1,NPNT
  DO N=1,NELN0
    IF (NO(L).TRIBLEN=0.)
      DO K=1,NELH
        IF (NO(EL(K),NODE(N),EQ.L) THEN
          IF (N.EQ.1) THEN
            M=4
          ELSE
            M=N-1
          END IF
          IF (NO(EL(K),NODE(M)).BC.EQ.4) THEN
            NO(L).TRIBLEN=NO(L).TRIBLEN+
              ABS(NO(EL(K),NODE(M)).XORD-NO(L).XORD)
          END IF
          IF (N.EQ.4) THEN
            M=4
          ELSE
            M=N-1
          END IF
        END IF
      END DO
    END IF
  END DO
END IF
C Read Soil Information
C
PRINT*, ' Read Soil Properties File... '
READ (7,'(15)') NMPROPSS
DO I=1,NMPROPSS
  READ (7,'(15,A75*)') NMPOINTS(I,1),IDPROP(I,1)
  READ (7,'(15,A75*)') NMPOINTS(I,2),IDPROP(I,2)
  READ (7,'(8F10.0)') (SDATA(I,1,J),J=1,NMPOINTS(I,1,1))
  READ (7,'(8F10.0)') (SDATA(I,2,J),J=1,NMPOINTS(I,2,1))
  READ (7,'(15,A75*)') NMPOINTS(I,3),IDPROP(I,3)
  READ (7,'(8F10.0)') (SDATA(2,1,J),J=1,NMPOINTS(I,2,1))
  READ (7,'(8F10.0)') (SDATA(1,1,J),J=1,NMPOINTS(I,1,1))
END DO
C Read Acceleration Input File
C
PRINT*, ' Read Earthquake File... '
IF (HDX.LT.1) THEN
  PRINT*, ' There must be at least one header line in the
  & acceleration input files. '
  STOP
END IF
READ (8,'(A)') EQRH
DO I=2,HDX
  READ (8,'(1)') (RFLX,I)
END DO
IF (RFLX.LT.1) THEN
  IF (EQINPM(I,1).EQ.'*') THEN
    READ(8,'(1)') (U26(I,1),I=1,KGMAX)
  ELSE
    READ(8,EOFINPM(I,1)) (U26(I,1),I=1,KGMAX)
  END IF
ELSE
  NCARDS=(KGMAX-1)/NPY+1
  IF (EQINPM(I,1).EQ.'*') THEN
    DO I=1,NCARDS
      READ(8,'(1)') (U26(I,1),I=1,KGMAX)
    END IF
    IF (I.NE.NCARDS) THEN
      READ(8,'*') (U26(I,1),I=1,NPPLY)
    ELSE
      READ(8,EOFINPM(I,1)) (U26(I,1),I=1,KGMAX)
    END IF
  END IF
END IF

```

PROGRAM: QUAD4M1

Page A - 16

```

      READ(8,*), (U2G(J,1),J=(1,1)*NPLX+1,KGMAX)
      END IF
      END DO
      ELSE
      DO I=1,NCARDS
      IF (I,NE,NCARDS) THEN
      READ(8,EQINPMT(1)) (U2G(J,1),J=(1,1)*NPLX+1,1*NPLX)
      ELSE
      READ(8,EQINPMT(1)) (U2G(J,1),J=(1,1)*NPLX+1,KGMAX)
      END IF
      END DO
      END IF
      IF (KV, EQ, 2) THEN
      IF (HORY, LT, 1) THEN
      PRINT*, 'There must be at least one header line in the'.
      & acceleration input files.
      STOP
      END IF
      READ(9,'(A') EQNV
      DO I=1,HORY
      READ(9,'(/') )
      END DO
      IF (NPLY, LT, 1) THEN
      IF (EQINPMT(2)(1), EQ, '*') THEN
      READ(9,*), (U2G(I,2),I=1,KGMAX)
      ELSE
      READ(9,EQINPMT(2)), (U2G(I,2),I=1,KGMAX)
      END IF
      ELSE
      NCARDS=(KGMAX-1)*NPLY+1
      IF (EQINPMT(2)(1), EQ, '*') THEN
      DO I=1,NCARDS
      IF (I,NE,NCARDS) THEN
      READ(9,*), (U2G(J,2),J=(1,1)*NPLY+1,1*NPLY)
      ELSE
      READ(9,*), (U2G(J,2),J=(1,1)*NPLY+1,KGMAX)
      END IF
      END DO
      ELSE
      DO I=1,NCARDS
      IF (I,NE,NCARDS) THEN
      READ(9,EQINPMT(2)), (U2G(J,2),J=(1,1)*NPLY+1,1*NPLY)
      ELSE
      READ(9,EQINPMT(2)), (U2G(J,2),J=(1,1)*NPLY+1,KGMAX)
      END IF
      END DO
      END IF
      END IF
      END IF
      RETURN
      END
      ****
      C SUBROUTINE OPENSTRES(.TILENAME)
      C
      C Opens Stress History Files
      C
      C Written for QUAD4M
      C
      C U.C. Davis, 1993
      C
      C by Martin Byrd Hudson, I.M. Idriess, Mohsen Beikae
      C
      COMMON/CONST4/SHISTEM, KHISTFT, SFILLCUT, AFILCUT,
      & KFILEOUT, SSUFFIX, ASUFFIX, KSUFFIX, DIROUT, SAVEFILE,
      & DATAIN
      CHARACTER SHISTFT*, KHISTFT*, SFILLCUT*, AFILCUT*8,
      & KFILEOUT*, SSUFFIX*, ASUFFIX*, KSUFFIX*, DIROUT*72,
      & SAVEFILE*, DATAIN*72
      CHARACTER FILENAME*12
      C
      IF (I,LT,1) THEN
      FILENAME=SFILLCUT//LEN(TRIM(SFILLCUT))//'0'//CHAR(47+1)//'
      & //SSUFFIX
      ELSE IF (I,LT,21) THEN
      FILENAME=SFILLCUT//LEN(TRIM(SFILLCUT))//'1'//CHAR(37+1)//'
      & //SSUFFIX
      ELSE IF (I,LT,31) THEN
      FILENAME=SFILLCUT//LEN(TRIM(SFILLCUT))//'2'//CHAR(27+1)//'
      & //SSUFFIX
      ELSE
      PRINT*, 'NUMBER OF STRESS HIST REQUESTED TOO LARGE'
      STOP
      END IF
      OPEN(19,FILE=DIROUT,LEN=TRIM(DIROUT)//FILENAME)
      RETURN
      END
      ****
      C *****SUBROUTINE SOILSCAN(NUMPROPS,MAXPOINTS)
      C
      C Finds size of soil properties input file
      C
      C Written for QUAD4M
      C
      C U.C. Davis, 1993
      C
      C by Martin Byrd Hudson, I.M. Idriess, Mohsen Beikae
      C
      MAXPOINTS=40
      PRINT*, 'Read Soil Properties File ..'
      READ(1, '(15)') NUMPROPS
      DO J=1,NUMPROPS
      DO K=1,2
      READ(1, '(15)') NUMPOINTS
      IF (NUMPOINTS GT MAXPOINTS) MAXPOINTS=NUMPOINTS
      READ(1, '(8F10.0)') (DATA,I=1,2*NUMPOINTS)
      END DO
      END IF
      END IF
      RETURN
      END
      ****
      
```

PROGRAM: QUAD4M

MODULE: QUAD4M1

Page A - 17

```
REWIND 7  
RETURN  
END
```

```

SUBROUTINE QUAD4EL(INO,ZHSS,SMSS,R,A1,B1,X1,X2,DSMAX,TIME,NBC,
6 U2G,X,ST,DS,MASK,IHISTAC,SDATA,GOATA,
6 DDATA,NUMPROPS,NUMPOINTS,NOSEG,ESEG,ELSEG)
6 NSEG,NOSEG,ESEG,ELSEG)

  Controling Finite Element Computation Subroutine

Written for QUAD4H
  U. C. Davis, 1993
  by Martin Byrd Hudson, I. M. Idriss, Mohsen Beikae

Modified from QUAD4, 1973.
  by I. M. Idriss, J. Lysmer, R. Hwang and H. Bolton Seed

STRUCTURE /ELEMENT/
  REAL GMX, G, E, EN AREA, XL, TIME2, S1G(3), SIGMAX(3), EPSMAX, PO, DENS
  INTEGER NODE(4), TYPE, LSTR
END STRUCTURE

STRUCTURE /NODE/
  REAL XORD, YORD, TRIBLEN, BETAS, BETAP, X21(2), X11(2), X1(2)
  INTEGER BC, OUT
END STRUCTURE

COMMON /CONST1/TITLE,DRF,MDPT,NBL,NSP,NELEM,UNITS,GRAV,NSLP,NF
CHARACTER TITLE*72,UNITS1
COMMON /CONST2/KGMX,KGEQ,NIEQ,N2EQ,N3EQ,NUMB,KV,KSAV
COMMON /CONST3/ODE,ECMUL(2),PDM,UGMAX(2),EQRN,ENVN,JEC(2),PRINPUT
CHARACTER/EQ/ ENVN, EQRN
COMMON /CONST4/ KHISTMT, KHISTFT, SEFILEOUT, AFILEOUT,
& DATAIN
  KFILEOUT, SUFFIX,X,SUFFIX,X,KFILEOUT,X,OTROUT,SAVEFILE,
& SAVETIME*12, DATAIN*12,
  KFILEOUT*8, SUFFIX*X*,ASUFIX*X*,KSUFFIX*X*,DIROUT*172,
  SAVETIME*12, DATAIN*12,
  IYR1, IMON1, IDAT1, 1100TH, IYR2, IMIN2, 1SEC2,
  IYR2, IMON2, IDAT2, 1DAY2
  INTEGER ESEG(*), EL SEG(NSLP,*)
  RECORD /ELEMENT/EL(*)
  RECORD /NODE/NOC()
  DIMENSION ZMASST(*,*),R(*),AT(*,*),BT(*,*),X1(*,*),X2(*,*),DSMAX(*,*),
  TIME(*,*),NBSL(*),UZG(*,*),UGMAX(*,*),X(*,*),ST(NF,*),DSNF(*,*),
  SDATA(2,NUMPROPS,*),GOATA(2,NUMPROPS,*),DDATA(2,NUMPROPS,*),
  NUMPOINTS(2,*),NSEG(*),NOSEG(NSLP,*),
  ELSL (*ALLOCATABLE) (:,:),
  ELMASSORG (*ALLOCATABLE) (:,:),
  SMOD (*ALLOCATABLE) (:,:),
  SELSMAK (*ALLOCATABLE) (:,:),
  WSILP (*ALLOCATABLE) (:,:)

  ALLOCATE (ELSTNEUM(8,8) ELMASSORG(NEHL), SMOD(NSLP,NSLP))
  IF (NSLP GT 0) THEN
    ALLOCATE (SELSSMAX(2MKV*NSLP),WSILP(NSLP))
  END IF
  CALL GETIM(CHRL,IMINI,1SEC1,1100TH)

```

PROGRAM: QIADAM

```

      AT(N) = 4.*X(N)/(DTEQ*DTEQ) + X1(N)/DTEQ + X2(N)/4.
      BT(N) = X1(N) + X(N)*2./DTEQ
      C
      C Modified Force Vector:
      C
      R(N) = SMAX(N)*AT(N) + DS(N,1)*BT(N)
      C
      C Add on Forces due to relative acceleration of Points
      C
      IF (MOD(N,2).NE.0) THEN
        R(N) = R(N) - SMAX(N)*UG(1)
      ELSE IF (KV.EQ.2) THEN
        R(N) = R(N) - SMAX(N)*UG(2)
      END IF
      DO N = 1, NF
        L = N - 1
        DO J = 2, MIN0 (MAXL,NF-N+1)
          K = L + J
          R(N) = R(N) + DS(N,J)*BT(K)
          R(K) = R(K) + DS(N,J)*BT(N)
        END DO
      END DO
      C
      C Solve for displacement
      C
      CALL SYMBOL(2,STMOD,R,NF,MAXL)
      DO N = 1, NF
        U0=X(N)
        U1=X1(N)
        U2=U2(N)
        X(N)=R(N)
        X2(N)=X(N)*4./((DTEQ*DTEQ) - AT(N))
        X1(N)=U0 + DTEQ/2.*((U20 + X2(N))
      END DO
      DO L = 1, 2
        DO N = 1, NF, 2
          A1 = ABS(X2(N) + UGL(1))           ! Disp at last step
          UG(1) = ABS(X2(N) + UGL(1))         ! Velocity at last step
          IF (OSMAX(N).GE. A1) CYCLE
          DSMAX(N)=A1
          TIME(N)=ELAT(KTIME)*DTEQ
        END DO
      END DO
      DO N = 1, NLM
        CALL STRESSEL(N),NO,KTIME,X)
      END DO
      IF (LOOP.NE.1) CYCLE
      C SAVE STATE OF SYSTEM IN FILE FOR RESTART
      IF (KTIME.EQ.M3.AND.KSAV.EQ.1) THEN
        DO N = 1, NF/2
          NO(N).X21(1) = X2(C2*N-1)
          NO(N).X21(2) = X2(C2*N)
          NO(N).X11(1) = X1(C2*N-1)
          NO(N).X11(2) = X1(C2*N)
        END IF
      END IF
      IF (ABS(X(N)).GE.100.) THEN ! If disp too large
        WRITE(*,905) LOOP,KTIME,X(N)
        WRITE(*,905) KTIME,X(N)
        WRITE(6,82) KTIME,X(N)
        WRITE(6,82) KTIME,X(N)
        WRITE(6,899) KTIME,X(N)
        WRITE(6,899) KTIME,X(N)
        STOP
      END IF
    END DO
    C
    C Start dynamic computation
    C
    WRITE(*,9405)
    LINE=0
    DO KTIME = N2, N3
      IF (LINE.LT.8) THEN
        LINE=LINE+1
      ELSE
        LINE=1
      END IF
      TIME=DTEQ*FLOAT(KTIME)
      WRITE(*,9505) LOOP,KTIME,TIME
      UG(1) = UG(KTIME,1)
      UG(2) = 0.0
      XIBASFORZ=XIBASFORZ + DTEQ/2.*((UG(KTIME,1,1)+UG(KTIME,2))
      IF (KV.EQ.2) THEN
        UG(2) = UG(KTIME,2)
        XIBASFORZ=XIBASFORZ + DTEQ/2.*((UG(KTIME,1,2)+UG(KTIME,2)))
      END IF
      DO N = 1, NF
        IF (ABS(X(N)).GE.100.) THEN ! If disp too large
          WRITE(*,82)
          WRITE(*,775) KTIME,X(N)
          WRITE(6,82)
          WRITE(6,775) KTIME,X(N)
          WRITE(6,899) KTIME,X(N)
          WRITE(6,899) KTIME,X(N)
        END IF
      END DO
    END DO
  END DO
END PROGRAM QUAD4M2

```

```

N(0).X(1) = X(2*N-1)
N(0).X(2) = X(2*N)
END DO
CALL WRITEFILE(NSEG,NSSEG,ESEG,ELSEG,EL,N)
END IF
Saves requested stress histories
IF (1HISTR.GT.0) THEN
  ISTRU=1
  IF (SHISTMT.EQ.'C' OR. SHISTMT.EQ.'C') THEN
    WRITE(10,'(FB.3.,1)') TIMEF
  END IF
  DO N=1, NELM
    IF (EL(N).LSTR.EQ.1) OR. EL(N).LSTR.EQ.3, OR. EL(N).LSTR
      .EQ.5, OR. EL(N).LSTR.EQ.7) THEN
      CALL STRESSOUT(SHISTMT,EL(N),SIG(1),ILINE,1STRES)
      1STRES=1STRES+1
    END IF
    IF (EL(N).LSTR.EQ.2, OR. EL(N).LSTR.EQ.3, OR. EL(N).LSTR
      .EQ.6, OR. EL(N).LSTR.EQ.7) THEN
      CALL STRESSOUT(SHISTMT,EL(N),SIG(2),ILINE,1STRES)
      1STRES=1STRES+1
    END IF
    IF (EL(N).LSTR.EQ.4, OR. EL(N).LSTR.EQ.5, OR. EL(N).LSTR
      .EQ.6, OR. EL(N).LSTR.EQ.7) THEN
      CALL STRESSOUT(SHISTMT,EL(N),SIG(3),ILINE,1STRES)
      1STRES=1STRES+1
    END IF
    IF (SHISTMT.EQ.'C' OR. SHISTMT.EQ.'C') THEN
      WRITE(10,'(1A,1)') )
    END IF
  END DO
  IF (SHISTMT.EQ.'C' OR. SHISTMT.EQ.'C') THEN
    WRITE(40,'(F10.3.,1)') TIMEF
  END IF
  DO N=1, NELM/2
    IF (ND(N).OUT.EQ.1) THEN
      CALL ACCOUNT(AHISTMT,(X2(N**2-1)+H6(1))/GRAV,ILINE,1ACC)
    ELSE IF (ND(N).OUT.EQ.2) THEN
      CALL ACCOUNT(AHISTMT,(X2(N**2)-H6(2))/GRAV,ILINE,1ACC)
    ELSE IF (ND(N).OUT.EQ.3) THEN
      CALL ACCOUNT(AHISTMT,(X2(N**2-1)+H6(1))/GRAV,ILINE,1ACC)
      CALL ACCOUNT(AHISTMT,(X2(N**2)-H6(2))/GRAV,ILINE,1ACC)
    END IF
  END DO
  IF (AHISTMT.EQ.'C' OR. AHISTMT.EQ.'C') THEN
    WRITE(40,'(1A,1)') )
  END IF
  END DO
  Computes and Saves requested seismic coefficient histories
  IF (NSLP.GT.0) THEN
    N(0).X(1) = X(2*N-1)
    N(0).X(2) = X(2*N)
  END IF
END IF

```

CALL SEISCOFF(EL,NO,KV,NOSEG,NSSEG,ELSEG,ESEG,WSLP,KHISTMT,
 ILINE,SEISMAX,TIMEF)
 & END IF
 END DO
 DO N = 1, NF
 DSA4X(N)=DSMAX(N)/GRAV
 END DO
 DO N = 1, NELM
 EL(N)=EPSMAX*100.*EL(N).EPSMAX
 END DO
 C Prints output and Calculates strain compatible properties
 C Calculates and prints strain compatible properties
 C Calculates and prints strain compatible properties
 C Writes(6,612)
 AR = 0.
 Z = 0.
 DO M = 1, NELM
 A = 2.*EL(M).PO
 S = PBM*EL(M).EPSMAX
 EL(M)=EL(M).E/A
 N = EL(M).TYPE
 IF (N.EQ.0) THEN
 DE=(EL(M).XI
 GEL(M).E
 ELSE
 CALL CMRP (N,S,D,G,DDATA,DATA,NUMPOINTS,NUMPROPS)
 D = 0.100.
 G=GEL(M).GX*1000
 END IF
 DE=100.*EL(M).E/G
 DZ = 100.*EL(M).XI/D
 AR = AR + EL(M).AREA
 Z = Z + D*EL(M).AREA
 WRITE(6,622) M,EL(M).E,G,DE,EL(M).XI,D,DZ
 EL(M)=EL(M).E/A
 EL(M).EN=G/A
 EL(M).XI = 0
 END DO
 IF (LOOP.EQ.NMB) THEN
 Prints Nodal Peak Acc.
 WRITE(6,552)
 DO N = 1, NF/2
 WRITE(6,562) N,NO(N).YORD,NO(N).YORD,DSMAX(2**N-1),
 TIME1(2**N-1).DSMAX(2**N).TIME1(2**N)
 END DO
 Prints Element Peak Stresses
 WRITE(6,592)
 DO N = 1, NELM
 WRITE(6,602) N,EL(N).SIGMAX(L),L=1,3),EL(N).EPSMAX,
 EL(N).TIME2
 END DO
 Prints Seismic Coefficient Max and Min
 IF (NSLP.GT.0) THEN
 END IF
 END IF

```

      WRITE(6,625)
      IF (KV.EQ.2) THEN
        WRITE(6,626)
      END IF
      WRITE(6,'(18(LH))')
      WRITE(6,627)
      IF (KV.EQ.2) THEN
        WRITE(6,628)
      END IF
      WRITE(6,'(//)')
      END IF
      DO M = 1, NSLP
        WRITE(6,623) M,JS1(JM),SE1SMAX(1,M),SE1SMAX(2,M)
        IF (KV.EQ.2) THEN
          WRITE(6,624) SE1SMAX(3,M),SE1SMAX(4,M)
        END IF
        WRITE(6,'(//)')
      END DO
      END IF
      AJ = Z/LAR
      WRITE(*,632) LOOP,AJ
      WRITE(6,632) LOOP,AJ
      CALL GETTIME(IHR2,IMIN2,ISEC2,1100TH)
      CALL GETDATE(YR2,IMON2,10AY2)
      TIM =60*(60*(24*(365*(1HR2-1YR1)+MODAY(1MON2)-MODAY(1MON1))+*
     & (1DAY2-1DAY1)*(1HR2-1HR1))+(IMIN2-IMIN1)+(1SEC2-1SEC1))
      IHR1=IHR2
      ISEC1=ISEC2
      IMON1=IMON2
      IYR1=IYR2
      IDAY1=IDAY2
      END DO
      N=1
      WRITE(*,7676)
      WRITE(6,7676)
      RETURN
      C
      2 FORMAT(112A6,12)
      82 FORMAT(1//1')
      552 FORMAT(1'/)
      & 5X,'PEAK NODAL ACCELERATION VALUES (G**S)',//,10X,'NODE',5X,
      & 'XORD',5X,'YORD',5X,'X-ACC',7X,'AT TIME',11X,'Y-ACC',/
      & 7X,'AT TIME',11X,'/')

      562 FORMAT(114.2F9.1,4F14.4)
      592 FORMAT(1//1.5X,'PEAK ELEMENTS STRESSES (ENG. PSF OR SI: N/MM^2),
      & AND STRAINS',//,10X,'ELM',11X,'SIG-X',10X,'SIG-Y',9X,'SIG-XY',
      & 9X,'EPS-MAX',8X,'AT TIME //')
      602 FORMAT(114.3F15.1,F15.3,F15.3)
      612 FORMAT(1/5X,'MODULI (ENGS: KSF or SI: KN/MM^2) AND DAMPING',/
      & 2X,'ELM',6X,'G-US'D,TX,'G-NEW',5X,'DIF-G',3X,
      & 'DAMP-US'D,A,X,'DAMP-NEW',2X,'DIF-DAMP',//)

```

```

SUBROUTINE ACCOUNT(HISTMT,ACC,ILINE,IACC)
C Outputs acceleration history to proper files
C
C Written for QUAD4M
C U.C. Davis, 1993
C by Martin Byrd Hudson, I.M. Idriss, Mohsen Beikae
C
CHARACTER*1 HISTMT
C
IF(HISTMT,EQ.'C',OR,HISTMT,ED.'C') THEN
  WRITE(IACC+39,'(F10.6)') ACC
ELSE
  IF(ILINE,LT,8) THEN
    WRITE(IACC+39,'(F10.6,\')') ACC
  ELSE
    WRITE(IACC+39,'(F10.6)') ACC
  END IF
  IACC=IACC+1
END IF
RETURN
END
C *****
C
SUBROUTINE CMPPR(N,S,D,G,DATA,SDATA,SDATA,NUMPOINTS,NUMPROPS)
C
C Calculates Strain Compatible Modulus and Damping
C
C Written for QUAD4M
C U.C. Davis, 1993
C by Martin Byrd Hudson, I.M. Idriss, Mohsen Beikae
C
C Modified from QUAD4, 1973
C by I.H. Idriss, J. Lyshier, R. Hwang and H. Bolton Seed
C
DIMENSION SDATA(2,NUMPROPS,*),GDATA(NUMPROPS,*),ODATA(NUMPROPS,*),
     & NUMPOINTS(2,*),J(2),FRACTION(2)
C
S = ABS(S)
DO K=1,2
  DO L=1,NUMPOINTS(K,N)
    J(K)=L
    IF(S,LE, SDATA(K,N,L)) EXIT
  END DO
END DO
DO K=1,2
  IF(S,GE, SDATA(K,N,1)) THEN
    FRACTION(K) = LOG10(S/SDATA(K,N,J(K)-1))/LOG10(SDATA(K,N,J(K))-1)
  END IF
END DO
IF(S,LT, SDATA(2,N,1)) THEN
  D = ODATA(N,1)
  MARRI=R(1)

```

MODUL E: OIIAD4M3

PROGRAM: QIAD4M

```

C Sum Mass around each node
C
C DO I=1,4
C   ZMASS(EL(N),NODE(I))=ZMASS(EL(N),NODE(I))+ELMASS
C END DO
C
C Preserve original element stiffness matrices and masses
C
C DO I=1,8
C   DO J=1,8
C     ELST(N,I,J)=ST(I,J)
C   END DO
C   END DO
C   ELMASSORG(N)=ELMASS
C END DO
C
C Store same mass in both x and y directions for each node
C
C DO M = 1, NF/2
C   SMASS(2*M+1) = ZMASS(M)
C   SMASS(2*M) = ZMASS(M)
C END DO
C
C Preserve nodal masses
C
C IF (I.EQ.1) THEN
C   AJ = ELMASSORG(N)
C ELSE
C   AJ = ELMASSORG(N)
C END IF
C
C Assemble global stiffness, mass, and damping matrices
C
C DO N = 1, NF/N
C   Restore element stiffness matrices and masses
C   EL(N).E=EL(N).EN
C   AK = EL(N).EN*EL(N).E
C   DO L = 1, 8
C     DO M = 1, 8
C       ELST(N,L,M) = AK*ELST(N,L,M)
C     END DO
C   END DO
C
C Assemble global stiffness matrix
C
C DO L = 1, 4
C   IF (EL(N).NODE(L).GE.NB1) CYCLE ! Skip if on rigid base
C   I=2*EL(N).NODE(L)-1 ! Global Row position
C   I=2*L-1 ! Element Row position
C   DO M = 1, 4
C     IF (EL(N).NODE(M).GE.NB1) CYCLE ! Skip if on rigid base
C     IF (EL(N).NODE(L).EQ.EL(N).NODE(M)) THEN
C       J=2*(EL(N).NODE(M)-EL(N).NODE(L))+1 ! Global Col posn
C       J=2*N-1
C       ST(I,J) = ST(I,J) + ELST(N,I,L,M)
C       ST(I,J-1) = ST(I,J-1) + ELST(N,I,L,M+1)
C       IF (J.NE.1) ST(I+1,J-1)=ST(I+1,J-1)+ELST(N,I+1,L,M)
C       ST(I+1,J) = ST(I+1,J) + ELST(N,I+1,J+1)
C     END IF
C   END DO
C END DO
C
C Preserve stiffness without b.c.'s temporarily in DS
C
C DO L = 1, NF
C   DO N = 1, NF
C     DS(L,N) = ST(L,N)
C   END DO
C END DO
C
C Essential boundary conditions for mass and stiffness
C
C DO N = 1, NF/2
C   IF (NO(0).BC.EQ.1.OR.NO(N).BC.EQ.3.OR.NO(N).BC.EQ.4) THEN
C     ST(2*N-1,1) = 1.
C     SMASS(2*N-1) = 0.
C   DO J = 2, MAXM
C     ST(2*N-1,J) = 0.
C     L = 2*N-J
C     IF (L.GT.0) THEN
C       ST(L,J) = 0.
C     END IF
C   END DO
C   IF (NO(0).BC.EQ.2.OR.NO(N).BC.EQ.3.OR.NO(N).BC.EQ.4) THEN
C     ST(2*N,1) = 1.
C     SMASS(2*N) = 0.
C   DO J = 2, MAXM
C     ST(2*N,J) = 0.
C     L = 2*N-J+1
C     IF (L.GT.0) THEN
C       ST(L,J) = 0.
C     END IF
C   END DO
C END IF
C
C Find first frequency of matrix pencil
C
C CALL EIGEN(EV,SMASS,ST,R,NF,MAXM)
C W1 = SORT(EV) ! 1st mode frequency (circular)
C PRL = 4.*PI*IN(1,)/W1 ! Period
C
C Find next higher odd integer to PRL/PRINPUT
C IF (PRINPUT.EQ.0) THEN

```

```

      WIMULT = 1
      ELSE
        WINMULT = INT((PRL/PRINPUT-1)/2* 99999)*2+1
      END IF
      W2 = W1*WIMULT ! Higher frequency
      PR2 = 4.*ASIN(1./W2)
      WRITE(*,302) W1,PRL,WIMULT,W2,PR2
      WRITE(6,302) W1,PRL,WIMULT,W2,PR2
      C Restore stiffness and mass without b.c.'s: initialize damping
      C
      DO M = 1, NF/2
        SMASS(2*M-1) = ZMASS(M)
        SMAS(2*M) = ZMAS(M)
      END DO
      DO L = 1, NF
        DO N = 1, NELEM
          ST(L,N) = DS(L,N)
          DS(L,N) = 0.
        END DO
      END DO
      C Set up damping matrix
      C
      C AJ = ELMASORG(N)
      C
      C Form element damping matrix;
      C Rayleigh damping = alpha*mass + beta*stiffness
      C
      C To minimize damping at w1:
      C if (wimult.eq.1) then
        AI = EL(N).WL*WL
        BI = EL(N).WL/WL
      else
        To set damping at w1 & w2:
        AI = 2.*EL(N).WL**WL*W2/(WL+W2)
        BI = 2.*EL(N).WL/(WL+W2)
      end if
      BJ = AI*AJ
      DO M = 1, 8
        DUL(M) = BI*ELST(N,L,M)
      END DO
      END IF
      RETURN
      302 FORMAT(//,
      &           'DAMPING SET AT THE FOLLOWING TWO FREQUENCIES: ',/
      &           'THE FIRST NATURAL FREQUENCY: CIRC FREQ= ',F12.3,
      &           'PERIOD= ',F10.3,' SEC ',/14,
      &           'TIMES THE NATURAL FREQ.: CIRC FREQ= ',F12.3,
      &           'PERIOD= ',F10.3,' SEC ',/)

      C Assemble global damping matrix
      C

```

MODULE: QUAD4M3

PROGRAM: QUAD4M

```

C *****
C ***** SUBROUTINE GL5T(X1,X2,X3,X4,Y1,Y2,Y3,Y4,S,C)
C ***** Numerical Integration of Element Matrix
C ***** Written for QUAD4H
C ***** by Martin Byrd Hudson, I.M. Idriess, Mohsen Beikae
C ***** Modified from QUAD4, 1973.
C ***** U.C. Davis, 1993
C ***** I.M. Idriess, Mohsen Beikae
C ***** DIMENSION ST(3,8),S(8,8),C(3,3)
C ***** DIMENSION VH(6),VS(6),VT(6),AP(2,4),AB(3,8),AS(8,8)
C ***** DATA ST/24/0.,/
C
C     VH(1)=0.5555555555555556 ! weighting factors for quad pts
C     VH(2)=0.8888888888888889
C     VH(3)=0.774596669241483
C     VS(1)=0. ! xi quadrature coordinates
C     VS(2)=0.
C     VS(3)=VS(1)
C     VT(1)=VS(1)
C     VT(2)=0.
C     VT(3)=VS(3)
C
C     Numerically integrate using 9 point quadrature
C
C     DO I=1,3
C       DO J=1,3
C         AP(1,1)=1.+VT(J)
C         AP(2,1)=1.+VS(1)
C         AP(1,2)=1.-VT(J)
C         AP(2,2)=1.-VS(1)
C         AP(1,3)=1.+VT(J)
C         AP(2,3)=1.+VS(1)
C         AP(1,4)=1.-VT(J)
C         AP(2,4)=1.-VS(1)
C
C         dy/dEta4:=
C           AP(2,1)*Y1+AP(2,2)*Y2+AP(2,3)*Y3+AP(2,4)*Y4
C           AP(1,1)*Y1-AP(1,2)*Y2-AP(1,3)*Y3-AP(1,4)*Y4
C           -dx/dEta4;
C           AP(2,1)*Y1-AP(2,2)*Y2-AP(2,3)*Y3-AP(2,4)*Y4
C           dx/dXi4;
C
C         AJ1=AP(1,1)*X1+AP(1,2)*X2+AP(1,3)*X3+AP(1,4)*X4
C         AJ2=AP(1,1)*Y1-AP(1,2)*Y2-AP(1,3)*Y3-AP(1,4)*Y4
C         AJ3=AP(1,1)*X1-AP(2,2)*X2-AP(2,3)*X3-AP(2,4)*X4
C
C         AJ4=AP(1,1)*Y1+AP(1,2)*Y2+AP(1,3)*Y3+AP(1,4)*Y4
C
C         AJ4=AJ1*AJ4-AJ2*AJ3
C
C         C0**=AJ1*AJ4-AJ2*AJ3
C
C         Prepare matrix [AB]
C
C         DO K=1,4
C           K1=2*K-1
C           K2=2*K
C           AB(1,K1)=AB(1,K1)*AJ1*AP(1,K)+AJ2*AP(2,K) ! det(J)*dR/dX*16
C           AB(3,K2)=AB(1,K1) ! det(J)*dR/dX*16
C           AB(1,K2)=0.
C           AB(2,K2)=AB(3)*AP(1,K)+AJ4*AP(2,K) ! det(J)*dR/dY*16
C           AB(3,K1)=AB(2,K2) ! det(J)*dR/dY*16
C           AB(2,K1)=0. ! det(J)*dN/dY*16
C
C           END DO
C
C           [ST]= [C]*[AB] = det(J)*[C]*[B]*16
C
C           DO L=1,3
C             ST(L,K)=ST(L,K)+C(L,K)*AB(N,K)
C           END DO
C
C           END DO
C
C           DO K=1,8
C             ST(L,K)=0.
C             DO N=1,3
C               ST(L,K)=ST(L,K)+C(L,N)*AB(N,K)
C             END DO
C           END DO
C
C           END DO
C
C           [AS]=[A8] ! det(J)*[ST]= det(J)*[C]*[B]*16^2
C
C           DO L=1,8
C             AS(L,K)=0.
C             DO M=1,3
C               AS(K,L)=AS(K,L)+AB(M,K)*ST(M,L)
C             END DO
C           END DO
C
C           END DO
C
C           DO K=1,8
C             DO L=1,8
C               S(K,L)=S(K,L)+(VH(1)*VH(3))*AS(K,L)/(16.*C0**)
C             END DO
C           END DO
C
C           RETURN
C           END
C
C           *****
C           FUNCTION MODAY(TMON)
C
C

```

MONDII E. ONTADAM

PROGRAM: QUADAM

PROGRAM: QUAD4M

Page A - 28

```

STY=(A13*T3+A4*T4)/A
GAM=(A3*T1+A4*T2+A1*T3+A2*T4)/A
T=EL.PO
Y = EL.E((1.+Z)*(1.-Z))
EL.SIG(1) = Y*((1.-Z)*ST*Z*ST)
EL.SIG(2) = Y*(Z*ST*(1.-Z)*ST)
EL.SIG(3) = 0.5*Y*(1.-Z)*T*GAM
EPS=SQRT(GAM**GAM+(STY-STY)*(STY-STY))
EL.EPSMAX = MAX(ABS(EPS),EL.EPSMAX)

DO L=1,3
IF (EL.SIGMAX(L).LT.ABS(EL.SIG(L))) THEN
  EL.SIGMAX(L) = ABS(EL.SIG(L))
  IF (L.EQ.3) EL.TIME2=FLOAT(KTIME)*10E0
END IF
END DO
RETURN
END

C *****
C ***** SUBROUTINE STRESSOUT(SHIFTMT,STRESS,ILINE,ISTRES)
C
C   subroutine stressout to stress output file
C
C   Written for QUAD4M
C   U.C. Davis, 1993
C   by Martin Byrd Hudson, I.M. Idriess, Mohsen Beikae
C
CHARACTER*1 SHIFTMT
IF(SHIFTMT.EQ.'C'.OR.SHIFTMT.EQ.'C') THEN
  ARITEL0,'(FB1.\v)' STRESS
ELSE
  IF(ILINE.LT.8) THEN
    WRITE(STRESS+9,'(FB1.\v)') STRESS
  ELSE
    WRITE(STRESS+9,'(FB1.)') STRESS
  END IF
END IF
RETURN
END

C *****
C ***** SUBROUTINE SYMBOL(KKKK,A,B,NM,MM)
C
C   SOLVE SET OF ALGEBRAIC EQUATIONS
C   KKK = 1: decomposes A
C   KKK = 2: reduces and backsubstitutes B
C
C   Written for QUAD4M
C   U.C. Davis, 1993
C
C *****
C
C   by Martin Byrd Hudson, I.M. Idriess, Mohsen Beikae
C
C   Modified from QUAD4, 1973,
C   by I.M. Idriess, J. Lysmer, R. Hwang and H. Bolton Seed
C
DIMENSION A(NN,*),B(*)
C
IF (KKK.EQ.1) THEN
  C
  TRIANGULARIZE A : A=L*T0*T
  C
  DO N=1,NN ! Row Position
    DO L=2,NN ! Column Position
      C=A(N,L)/A(N,1)
      I=N-1
      IF (NN-1.GE.0) THEN
        J=0
        DO K=L,NN
          A(I,J)=A(I,J)-C*A(N,K)
        END DO
        I=I+1
      END IF
    END DO
  END DO
  RETURN
C
C *****
C
C   Reduces B
C
ELSE
  DO N=1,NN
    DO L=2,NN
      I=N-1
      IF (NN-1.GE.0) EXIT
      B(I)=B(I)-A(N,I)*B(N)
    END DO
    B(N)=B(N)/A(N,1)
  END DO
  C
  Backsubstitutes B
  C
  N=NN-1
  DO WHILE (N.NE.0)
    DO K=2,NN
      L=N-1
      IF (NN.GE.L) B(N)=B(N)-A(N,K)*B(L)
    END DO
    N=N-1
  END DO
  RETURN
END IF
END

C *****
C
C *****

```

MODULE: Q11AD4M3

PROGRAM: QUAD4M

PROGRAM: QUAD4M

Page A - 30

```

FACTOR=FACTNUM/FACTDENOM
SIGX=EL*(NEL1).SIG(1)-EL*(NEL2).SIG(1)*FACTOR*EL*(NEL2).SIG(1)
SIGY=(EL*(NEL1).SIG(1)-EL*(NEL2).SIG(2))*FACTOR*EL*(NEL2).SIG(2)
TAUXY=EL*(NEL1).SIG(3)-EL*(NEL2).SIG(3)*FACTOR*EL*(NEL2).SIG(3)
FORCEN=FORCEV+SIGX*HGT+SIGY*WDT
IF (KV EQ. 2) THEN
  FORCEV=FORCEV*SIGY*WDT+TAUXY*HGT
END IF

END DO
ACCANE=FORCEH*(NSLIP*)
IF (ACCANE LT. SELSHAK(1,J)) SELSHAK(1,J)=ACCANE
IF (ACCANE GT. SELSHAK(2,J)) SELSHAK(2,J)=ACCANE
CALL ACCOUNT(KSHTFT,ACCANE,ILINE,N)
IF (KV EQ. 2) THEN
  AWAVE=FORCEV*NSLIP*
  IF (AWAVE LT. SELSHAK(3,J)) SELSHAK(3,J)=AWAVE
  IF (AWAVE GT. SELSHAK(4,J)) SELSHAK(4,J)=AWAVE
  CALL ACCOUNT(KSHTFT,AWAVE,ILINE,N)
ENDIF
END DO
IF (KSHTFT EQ. 'C' OR. KSHTFT EQ. 'C') THEN
  WRITE(N+39,'(I\')')
END IF
RETURN
C *****
C SUBROUTINE WRITETFILE(NSSEG,NSSEG,ESEG,ELSEG,EL,NO)
C Produces file for restart
C Written for QUAD4M
C U.C. Davis, 1993
C by Martin Byrd Hudson, I.M. Doriss, Mohsen Beikae
C
STRUCTURE /ELEMENT/
REAL GRX,G,E,EN,AREA,XL,TIME2,SIGMAX(3),EPSMAX,PO,DENS
INTEGER NODE(4),TYPE,LSTR
END STRUCTURE
REAL XORD,YORD,TRIBLEN,BETAS,BETA2,X21(2),X1(2)
INTEGER BG,OUT
C
END STRUCTURE
CHARACTER TITLE*72,UNITS*1
COMMON/CONST1/KGMAX,KGEQ,NIEQ,NEDQ,NELP,NELM,UNITS,GRAY,NSLP,NF
COMMON/CONST2/KGMAX,KGEQ,NIEQ,NEDQ,NELP,KV,KSAV
COMMON/CONST3/DTEQ,EDQL(2),PRN,DSMAX(2),EQW,DEG(2),PRINPUT
CHARACTER*72 EARTH,FONN
COMMON/CONST4/SHTFTFT,AHLSHTFT,KHLSHTFT,SHLHTFT,SHLHTFT,SHLHTFT,SHLHTFT
KFILEOUT,SSUFFIX,ASUFFIX,KSUFFIX,DIROUT,SAVEFILE,
DATAIN
CHARACTER SHLHTFT*1,AHLSHTFT*1,KHLSHTFT*1,SHLHTFT*1,ASUFFIX*3,KSUFFIX*3,DIROUT*72,
KFILEOUT*8,SSUFFIX*3,ASUFFIX*3,KSUFFIX*3,DIROUT*72,
& WRITE(4,'(A,A,13X,A)' ) 'ACCELERATION OUTPUT FORMAT (K or C)', ***
& FILE PREFIX, AND SUFFIX: *** (A), ***
& WRITE(4,'(A,A,13X,A)' ) AHLSHTFT

```

```

      WRITE (4,'(A)') AFILEOUT
      WRITE (4,'(A)') ASUFFIX
      END IF
      IF (NSLP.GT.0) THEN
        WRITE (4,'(A,13X,A)') 'SEISMIC COEFF OUTPUT FORMAT (H or C).',
        & ' FILE PREFIX AND SUFFIX: *** (A) ***'
        WRITE (4,'(A,L1)') KHISFM
        WRITE (4,'(A)') KFILEOUT
        WRITE (4,'(A,L1)') KSUFFIX
        DO I=1,NSLP
          WRITE (4,'(A,(5X,A),NSEG ESEG)',*** '(215)'           ***,
                '(215)',NSEG(1),ESEG(1))
          WRITE (4,'(A,(5X,A),NSEG)',*** '(1515)'           ***,
                '(1515)',NSEG(1))
          WRITE (4,'(1515)',<NSEG(1,J),J=1,SEG(1))           ***
          WRITE (4,'(A,(5X,A),ELSEG)',*** '(1515)'           ***,
                '(1515)',ELSEG(1))
          WRITE (4,'(1515)',<ELSEG(1,J),J=1,SEG(1))           ***
        END DO
      END IF
      WRITE (4,'(A,A1)') ' N NP1 NP2 NP3 NP4 TYPE',
      & ' DENS PU GX G XL LSTR',
      & ' ***(615,5F10.0,15) ***'
      DO N=1,NELM
        WRITE (4,'(615,5G10.5E115)') N,EL(N),NODE(1),I=1,4),
        & EL(N),TYPE,EL(N),DENS,EL(N),PO,EL(N),GX,
        & EL(N),E/(2.*((1.+EL(N),PO)),EL(N),XL,EL(N),LSTR
      END DO
      WRITE (4,1)
      DO N=1,NOPT
        WRITE (4,'(15.2G10.4E1215,6E13.6)',N,NO(N),XORD,
        & NO(N),YORD,NO(N),BC,NO(N),OUT,NO(N),X21(1),
        & NO(N),X11(1),NO(N),X11(2),NO(N),X21(2),NO(N),X11(2),
        & NO(N),X11(2))
      END DO
      RETURN
1   FORMAT (' N XORD YORD 8C',
      & ' OUT 9X,X21H',9X,X11H',10X,X11H',9X,X21V',9X,X11V',10X,
      & 'XIV *** (15.2F10.0,215,6E13.0) ***')
      END

```


Appendix B - Sample Input

FILE: EXAMPLE.Q4I

```

Sliding Block Example Problem
UNITS (E for English, S for SI): *** (A1) ***
E
    DRF      PRM      ROCKVP      ROCKVS      ROCKRHO      *** (6F10.0)      ***
    1        0.65
    NELM NDPT NSLP      *** (3I5)      ***
    330 388 4
    KGMAX KGEQ N1EQ N2EQ N3EQ NUMB KV KSAV      *** (8I5)      ***
    2000 2000 1 1 2000 3 2 1
    DTEQ EQMUL1 EQMUL2 UGMAX1 UGMAX2 HDRX HDRY NPLX NPLY PRINPUT *** (5F10.0,4I5,F10.0) ***
    0.02 1 1 3 3 8 8 0.153
EARTHQUAKE INPUT FILE NAME(S) & FORMAT(S) (* for FREE FORMAT) *** (A) ***
SC_0.ACC
*
SC_V.ACC
*
SOUT ADOUT KOUT      *** (3I5)      ***
1 1 1
STRESS OUTPUT FORMAT (M or C), FILE PREFIX, AND SUFFIX: *** (A) ***
COMBINED
EXAMPLE
Q4S
ACCELERATION OUTPUT FORMAT (M or C), FILE PREFIX, AND SUFFIX: *** (A) ***
COMBINED
EXAMPLE
Q4A
SEISMIC COEFF OUTPUT FORMAT (M or C), FILE PREFIX, AND SUFFIX: *** (A) ***
COMBINED
EXAMPLE
Q4K
SYSTEM STATE OUTPUT FILE: *** (A)
EXAMPLE.Q4R
NSEG ESEG      *** (2I5)      ***
5 3
NOSEG      *** (15I5)      ***
122 130 139 138 137
ELSEG      *** (15I5)      ***
100 108 107
NSEG ESEG      *** (2I5)      ***
8 9
NOSEG      *** (15I5)      ***
122 130 139 148 157 166 165 164
ELSEG      *** (15I5)      ***
100 108 107 116 115 124 123 132 131
NSEG ESEG      *** (2I5)      ***
11 18
NOSEG      *** (15I5)      ***
111 117 124 132 141 150 159 158 157 156 155
ELSEG      *** (15I5)      ***
89 94 95 100 101 102 107 108 109 110 115 116 117 118 123
124 125 126
NSEG ESEG      *** (2I5)      ***
14 30
NOSEG      *** (15I5)      ***
111 117 124 132 141 150 159 168 177 186 185 184 183 182
ELSEG      *** (15I5)      ***
89 94 95 100 101 102 107 108 109 110 115 116 117 118 123
124 125 126 131 132 133 134 139 140 141 142 147 148 149 150
    N NP1 NP2 NP3 NP4 TYPE      DENS      PO      GMX      G      XL LSTR *** (6I5.5F10.0,15) ***
    1 2 7 6 1 120 0.45 345 249. .08198
    2 2 3 8 7 1 120 0.45 345 208. .11187
    3 3 4 9 8 1 120 0.45 345 207. .11235
    4 4 5 10 9 1 120 0.45 345 186. .12847
    5 6 7 12 11 1 120 0.45 345 220. .10279
    6 7 8 13 12 1 120 0.45 345 157. .15053

```

FULL INPUT NOT SHOWN

314	360	361	370	369	1	120	0.45	345	171.	.13981
315	362	363	372	371	1	120	0.45	345	271.	.06606
316	363	364	373	372	1	120	0.45	345	207.	.11287
317	364	365	374	373	1	120	0.45	345	197.	.12035
318	365	366	375	374	1	120	0.45	345	196.	.12108
319	366	367	376	375	1	120	0.45	345	176.	.13583
320	367	368	377	376	1	120	0.45	345	177.	.13538

321	368	369	378	378	377	1	120	0.45	345	171.	.13995		
322	369	370	379	379	378	1	120	0.45	345	169.	.14123		
323	371	372	381	380	1	120	0.45	345	266.	.06966			
324	372	373	382	381	1	120	0.45	345	243.	.08622			
325	373	374	383	382	1	120	0.45	345	241.	.08720			
326	374	375	384	383	1	120	0.45	345	243.	.08638			
327	375	376	385	384	1	120	0.45	345	228.	.09700			
328	376	377	386	385	1	120	0.45	345	225.	.09885			
329	377	378	387	386	1	120	0.45	345	214.	.10760			
330	378	379	388	387	1	120	0.45	345	210.	.11063			
N	XORD	YORD	BC	OUT	X2IH	X1IH	XIH	X2IV	X1IV	XIV	*** (I5.2F10.0.2I5.6F10.0) ***		
1	-2100	50		3									
2	-2100	37.5		3									
3	-2100	25		3									
4	-2100	12.5		3									
5	-2100	0		3									
6	-1860	50											
7	-1860	37.5											
8	-1860	25											

FULL INPUT NOT SHOWN

367	3400	37.5										
368	3400	25										
369	3400	12.5										
370	3400	0	3									
371	3770	100										
372	3770	87.5										
373	3770	75										
374	3770	62.5										
375	3770	50										
376	3770	37.5										
377	3770	25										
378	3770	12.5										
379	3770	0	3									
380	4250	100	3									
381	4250	87.5	3									
382	4250	75	3									
383	4250	62.5	3									
384	4250	50	3									
385	4250	37.5	3									
386	4250	25	3									
387	4250	12.5	3									
388	4250	0	3									

FILE: NEWSOIL.DAT

```

4
10 #1 MODULUS FOR CLAY (Idriss, 1990; Seed & Sun 1989) upper range
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    1.0     1.0     1.0    0.981   0.941   0.847   0.656   0.438
  0.238   0.144
10 DAMPING FOR CLAY (Idriss, 1990)
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    0.24    0.42    0.8     1.4     2.8     5.1     9.8    15.5
   21.     25.
10 #2 MODULUS FOR SAND (Idriss, 1990; Seed & Idriss 1970) - upper Range
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    1.0     1.0     0.99   0.96    0.85   0.64    0.37   0.18
  0.08   0.050
10 DAMPING FOR SAND (Idriss, 1990) - (about LRng from SI 1970)
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    0.24    0.42    0.8     1.4     2.8     5.1     9.8    15.5
   21.     25.
10 #3 MODULUS FOR CLAY (Idriss, 1990; Seed & Sun 1989) upper range
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    1.0     1.0     1.0    0.981   0.941   0.847   0.656   0.438
  0.238   0.144
10 damping for sand & refuse (Seed & Idriss 1970) - avg-
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    1.00   1.50    2.5     3.8     6.1     8.6    12.4    16.9
   21.     26.
10 #4 MODULUS FOR SAND (Idriss, 1990; Seed & Idriss 1970) - upper Range
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    1.0     1.0     0.99   0.96    0.85   0.64    0.37   0.18
  0.08   0.050
10 damping for sand & refuse (Seed & Idriss 1970) - avg-
  .0001   .0003   .001   .003   .01    .03    0.1    0.3
    1.      3.
    1.00   1.50    2.5     3.8     6.1     8.6    12.4    16.9
   21.     26.

```

FILE: SC_0.ACC (partial listing)

```

" Loma P. Eqk", "Santa Cruz", "H2 O", "init. vel:", "1.369 c/s", "disp: -1.066 cm"
" Total No. of Points : ", 2000, "@ DT = ", .02
" Peak Acceleration (g) = ", .4413005, "@ Time (sec) : ", 7.54
-0.002839 0.003273 0.004088 0.002397 -0.004423 -0.003957 0.002609 0.003254
-0.002257 -0.002117 0.005400 0.000527 -0.005296 -0.005056 -0.000565 0.001540
0.003347 0.010519 0.011245 0.000712 -0.002901 -0.006159 -0.011535 -0.015913
-0.008552 -0.008890 -0.000954 0.011002 0.021674 0.022078 0.016705 0.003633
-0.019671 -0.025850 -0.029108 -0.027187 -0.011781 0.020643 0.027084 0.026296

```

FILE: SC_V.ACC (partial listing)

"Loma P. Eqk", "Santa Cruz", "Vert", "init. vel:", ".051 c/s", "disp: -0.143 cm"
"Total No. of Points :".2000, "@ DT =".02
"Peak Acceleration (g) =".3307056, "@ Time (sec) :".7.42
-0.000521 -0.001011 0.000503 0.002590 -0.008624 -0.002452 0.032474 0.034005
-0.002525 -0.028430 -0.015743 0.007411 0.024565 -0.012625 -0.033764 -0.004639
0.007550 0.011532 0.006985 0.012103 0.012117 0.011107 -0.000131 0.002244
0.004372 -0.005036 -0.013182 -0.010066 -0.000011 -0.003768 -0.005271 -0.000947
0.005143 -0.003727 -0.009742 -0.009777 0.015026 0.014614 -0.001033 -0.019850

FILE: EXAMPLE.Q40

Appendix C - Sample Output

A. COMPARISON OF EVALUATING THE USELESS	SEISMIC RISK ASSESSMENT IN SULI SEDIMENTARY L. L. Davis, 1993	by Martin Borchsenius I. M. Ulrichs, and Michael Behre	MCULLY, C.R., QUAR., 1973	by J. M. Ulrichs, J. Lyman, R. Young, and H. Bolton, Sed.
SILICON BLOCK LAYERED PROBLEMS				
HORIZONTAL ACCELERATION 1980 FILE:				
SC_0100	WITH 1980 FILE	Lens 1, Err = "Santa Cruz", "NEC", "Wet", "1", "369 c/s", "dip": -1, 066	Vertical Acceleration 1980 FILE	WITH FIRST 1000 m of the Layer 1, Err = "Santa Cruz", "Wet", "100 vol.", "0.05 c/s", "dip": -0, 147

NO. OF ELEMENTS	200
NO. OF MODEL POINTS	388
DEGREES OF FREEDOM	756
INITIAL BOUNDARY	22
CONTINUOUS BOUNDARY	108
NO. OF FIXED BOUND. CONDITIONS	112
LAST TOTAL NO. OF PITS USED IN EACH CYCLE	2000
INIT. TO PTS. USED IN EACH CYCLE	1
INT. TO PTS. USED IN EACH CYCLE	1
INT. TO INT. OF RELOADS	0.020
STRAIN CONVERGENCE FACTOR	-0.6500
UNLOADING RATE CONVERGENCE FACTOR	1.000
UNLOADING RATE INITIALIZATION FACTOR	-1.5500
END. MAT. T. FACTOR	1.0000

10. M.R.J. FAVOR (VER) CDP 1-1 1-9000
 MATHMATICAL INSTRUMENTS (VER) CDP 1-1 3507
 4 LURK HISTORY RECORDED.
 5 ACCURATE RECORDS.
 6 SETS OF FAITH HISTORY RECORDS.
 7 OUTPUT REPORT FILES ARE AS TDA-045.
 ELEMENT 10 SIGHT X IN FILE ELEMENT 045
 ELEMENT 10 SIGHT X IN FILE ELEMENT 045
 ELEMENT 10 TAU XY IN FILE ELEMENT 045
 ELEMENT 14B TAU XY IN FILE ELEMENT 045
 ELEMENT 37 X IN FILE ELEMENT 045

MATERIAL	TYPE NO.	MODULUS:	E3 MODULUS:	DAMPING:
STRAIN	G/INCH			
.0001	1,000			
.0003	1,000			
.0010	1,000			
.0030	.981			

SAMPLE OUTPUT

EIE: EXAMPIE 040

SAMPLE OUTPUT

FILE: EXAMPLE.Q40

FULL OUTPUT NOT PRINTED FOR FIRST ITERATIONS

ITERATION CYCLE NO. 2 AND DEFAULT DUMP - 124

TIME REQUIRED FOR 2000 STEPS = 184. SEC

1

DEFINING SET AT THE FOLLOWING TWO FREQUENCIES
FOR THE FIRST NATURAL FREQUENCY, CIRC FREQU.
13 TIMES THE NATURAL FREQU.: CIRC FREQU.

43 SEC : 44 SEC

TIME REQUIRED FOR 1000,1000 TRIANGULATION OF MATRICES = 8. SEC

MATRIX (INC. KSF OR ST. (K0*W2)) AND LUMPING

CUR GUSTO G-NORM DIFF-6 DAMP-NON DAMP-UND

FULL OUTPUT NOT SHOWN

1 245.3 244.5 .3 .19446 .08504 .7

2 210.0 .0 .11039 .11039 .0

3 204.5 204.5 .0 .11039 .11039 .0

4 172.9 172.1 .4 .13852 .13909 .4

5 218.3 218.7 .0 .10416 .10416 .4

FULL OUTPUT NOT SHOWN

314 171.0 170.9 .1 .13991 .13999 .1

315 269.5 269.7 .1 .08716 .08717 .2

316 208.7 .2 .11115 .11115 .2

317 192.8 192.5 .2 .12337 .12337 .2

318 193.0 193.1 .0 .12325 .12325 .0

319 174.1 174.0 .0 .13163 .13163 .0

320 175.5 175.4 .0 .13653 .13653 .0

321 169.2 169.2 .0 .14264 .14264 .0

322 169.8 169.8 .0 .14935 .14935 .0

323 227.7 227.8 .1 .06494 .06494 .2

324 246.8 246.1 .1 .06496 .06496 .2

325 242.1 242.1 .0 .08917 .08917 .0

326 2100.0 50.0 .4113 .4113 .0

3 2100.0 37.5 .4413 .4413 .1

4 2100.0 25.0 .4413 .4413 .1

5 2100.0 12.5 .4413 .4413 .1

6 1860.0 50.0 .4413 .4413 .0

7 1860.0 37.5 .2716 .2716 .0

8 1860.0 25.0 .3209 .3209 .0

9 1860.0 12.5 .3369 .3369 .0

10 1860.0 0 .4413 .4413 .0

11 1675.0 50.0 .3938 .3938 .0

12 1675.0 37.5 .3359 .3359 .0

13 1675.0 25.0 .3827 .3827 .0

14 1675.0 12.5 .4075 .4075 .0

15 1675.0 0 .4413 .4413 .0

16 -1597.0 50.0 .3884 .3884 .0

17 -1597.0 37.5 .3418 .3418 .0

SAMPLE OUTPUT

FULL OUTPUT NOT SHOWN

TIME REQUIRED FOR 2000 STEPS = 184. SEC

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

1

