

N73-15368

Ala. u. RI-8-25102-FTR, Nov 72

NASA CR-124006

51

MATHEMATICAL CHARACTERIZATION OF MECHANICAL BEHAVIOR
OF POROUS FRICTIONAL GRANULAR MEDIA

By

T. J. Chung and J. K. Lee

PROPERTY OF
NATIONAL LIBRARY
OF AERONAUTICS
& SPACE

Final Technical Report

This research work was supported by the
National Aeronautics and Space Administration
under Contract NAS8-25102

Department of Engineering Mechanics
The University of Alabama in Huntsville
Huntsville, Alabama

November, 1972

MATHEMATICAL CHARACTERIZATION OF MECHANICAL BEHAVIOR
OF POROUS FRICTIONAL GRANULAR MEDIA

By

T. J. Chung and J. K. Lee

Final Technical Report

This research work was supported by the
National Aeronautics and Space Administration
under Contract NAS8-25102

Department of Engineering Mechanics
The University of Alabama in Huntsville
Huntsville, Alabama

November, 1972

PREFACE

This report consists of two parts. Part I is concerned with the static displacement and stress fields. Part II describes the dynamic wheel-soil interaction. These studies were conducted during the period January 1, 1972 through October 31, 1972, under NASA Research Contract NAS8-25102 "Mathematical Characterization of Mechanical Behavior of Porous Frictional Granular Media," technically monitored by Dr. N. C. Costes, The Geotechnical Laboratory of the Marshall Space Flight Center, NASA, Huntsville, Alabama.

TABLE OF CONTENTS

| | | Page |
|------------|---|-------|
| PART I | STATIC DEFORMATION AND STRESS FIELDS | I-1 |
| I.1 | INTRODUCTION | I-1 |
| I.2 | YIELD CRITERIA AND PLASTIC STIFFNESS | I-2 |
| I.3 | APPLICATIONS | I-10 |
| I.3.1 | PLATE BEARING | I-10 |
| I.3.2 | CONE-PENETROMETER | I-13 |
| I.4 | CONCLUSIONS | I-19 |
| REFERENCES | | I-21 |
| APPENDIX | 1. COMPUTER PROGRAM LISTING | I-23 |
| | 2. DATA INPUT FORMAT | I-44 |
| | 3. FLOW CHART | I-47 |
| | 4. SUBROUTINE ORGANIZATION CHART | I-49 |
| | 5. DESCRIPTIONS OF SUBROUTINES | I-50 |
| PART II | DYNAMICS OF WHEEL-SOIL INTERACTION | II-52 |
| II.1 | INTRODUCTION | II-52 |
| II.2 | BALANCE OF ENERGY AND LINEAR MOMENTUM | II-55 |
| II.3 | INCREMENTAL FREE ENERGY FUNCTIONS | II-56 |
| II.4 | INELASTIC RESPONSE | II-59 |
| II.5 | FINITE ELEMENT EQUATIONS OF MOTION | II-60 |
| II.6 | SOLUTION PROCEDURE FOR INCREMENTAL EQUATIONS OF MOTION | II-63 |

TABLE OF CONTENTS (cont.)

| | Page |
|--|--------|
| II.7 EQUIVALENT DYNAMIC WHEEL LOADS | II-64 |
| II.8 DEFORMATION AND STRESS FIELDS | II-66 |
| II.9 CHARACTERIZATION OF SOIL MECHANICS PARAMETERS | II-76 |
| II.10 CONCLUSIONS | II-89 |
| REFERENCES | II-91 |
| APPENDIX 1 DERIVATION OF INTERNAL (HIDDEN) VARIABLES | II-94 |
| APPENDIX 2 CONTACT STRESSES AT WHEEL-SOIL INTERFACE | II-97 |
| APPENDIX 3 COMPUTER PROGRAM LISTING | II-101 |
| APPENDIX 4 DATA INPUT FORMAT | II-124 |
| APPENDIX 5 FLOW CHART | II-127 |
| APPENDIX 6 SUBROUTINE ORGANIZATION CHART | II-129 |
| APPENDIX 7 DESCRIPTIONS OF SUBROUTINES | II-130 |

LIST OF FIGURES

| | Page |
|---|-------|
| <u>PART I.</u> | |
| Figure 1. Yield Surface and Yield Locus (after Roscoe and Burland) | I-4 |
| Figure 2. Plate Bearing Geometry | I-11 |
| Figure 3. Deformation at Center of Plate | I-12 |
| Figure 4. Deformed Geometry | I-14 |
| Figure 5. Cone-Penetrometer Geometry | I-15 |
| Figure 6. Cone-Penetrometer Tests | I-16 |
| Figure 7. Force-Displacement Curves for Cone Penetrometer | I-17 |
| Figure 8. Deformed Configuration for Finite Element Solution (1) at $F = 25 \text{ N/cm}^2$ | I-18 |
| <u>PART II.</u> | |
| Figure 1. Wheel-Soil Interaction Geometry. | II-65 |
| Figure 2. Radial and Tangential Stress Distribution at the Interface for 3.1% Slip on Compact Sand. Ref. [6]. | II-68 |
| Figure 3. Radial and Tangential Stress Distribution at the Interface for 41.4% Slip on Compact Sand. Ref. [6]. | II-69 |
| Figure 4. Equivalent Nodal Forces as Determined from Figures 3 and 4. | II-70 |
| Figure 5. Time-Displacement Curves for 3.1% Slip at Node No. 31. | II-71 |
| Figure 6. Vector Representation of Displacements (Static Elasto- plastic Analysis for 3.1% Slip). | II-72 |
| Figure 7. Vector Representation of Displacements (Static Elasto- plastic Analysis for 41.4% Slip). | II-73 |
| Figure 8. Vector Representation of Deformation (Dynamic Visco- elastoplastic Analysis for 41.4% Slip). | II-74 |

LIST OF FIGURES (Cont.)

| | Page |
|---|-------|
| Figure 9. Deformed Configuration at .3 sec. (Dynamic Viscoplastic Analysis for 41.4% Slip). | II-75 |
| Figure 10. Isobars of Major Principal Stress (in psi) for 3.1% Slip. | II-77 |
| Figure 11. Isobars of Major Principal Stress (in psi) for 41.4% Slip. | II-78 |
| Figure 12. Isobars of Maximum Shear Stress for 3.1% Slip. | II-79 |
| Figure 13. Isobars of Maximum Shear Stress for 41.4% Slip. | II-80 |
| Figure 14. Isobars of Major Principal Stresses and Maximum Shear Stresses (psi) at $t = 0.072$ sec. (Dynamic Elastoplastic Analysis for 3.1% Slip). | II-81 |
| Figure 15. Isobars of Major Principal Stress and Maximum Shear Stress (psi) at $t=0.15$ sec. (Dynamic Elastoplastic Analysis for 3.1% Slip). | II-82 |
| Figure 16. Isobars of Principal Stress and Maximum Shear Stress (psi) at $t=0.228$ sec. (Dynamic Elastoplastic Analysis for 3.1% Slip). | II-83 |
| Figure 17. Isobars of Major Principal Stress at Maximum Shear Stress (psi) at $t=0.3$ sec. (Dynamic Elastoplastic Analysis for 3.1% Slip). | II-84 |
| Figure 18. Isobars of Major Principal Stresses and Maximum Shear Stresses at $t=0.3$ sec. (Dynamic Viscoelastoplastic Analysis for 3.1% Slip). | II-85 |
| Figure 19. Isobars of Major Principal Stresses and Maximum Shear Stresses (psi) at .6 sec. (Dynamic Viscoelastoplastic Analysis for 3.1% Slip). | II-86 |
| Figure 20. Isobars of Maximum Shear and Major Principal Stresses (psi) at $t=0.3$ sec. (Dynamic Viscoelastoplastic Analysis for 41.4% Slip). | II-87 |
| Figure 21. Isobars of Major Principal Stress and Maximum Shear Stress (psi) at $t=0.65$ sec. (Dynamic Viscoelastoplastic Analysis for 41.4% Slip). | II-88 |

ABSTRACT

A new definition of loading and unloading along the yield surface of Roscoe and Burland is introduced. This is achieved by noting that the strain-hardening parameter in the plastic potential function is deduced from the yield locus equation of Roscoe and Burland. The analytical results are compared with the experimental results for plate-bearing and cone-penetrometer problems and close agreements are demonstrated.

The second part of the reports deals with the wheel-soil interaction under dynamic loading. The rate-dependent plasticity or viscoelastoplastic behavior is considered. This is accomplished by the internal (hidden) variables associated with time-dependent viscous properties directly superimposed with inelastic behavior governed by the yield criteria of Roscoe and Burland. Effects of inertia and energy dissipation are properly accounted for. Exhaustive example problems are presented.

PART I

STATIC DEFORMATION AND STRESS FIELDS

I-1. INTRODUCTION

Recent achievements in the critical state soil mechanics advanced by Roscoe and others [1,2] have stimulated many other investigators searching for practical applications. Initial attempts have been made by Smith and Kay [3], Zienkiewicz [4], Chung and Lee [5], and Chung, Costes, and Lee [6] in the context of finite element techniques. The present study is an extension of [5,6] with some significant modifications in reference to interpretation of the yield criteria of Roscoe and Burland [1].

In the previous works [5,6], the authors considered the strain-hardening parameter to be controlled by the constant yield stress, an independent material parameter, in addition to the basic material properties M , λ and μ proposed by Roscoe and Burland [1]. However, in view of the fact that the equation of yield surface and subsequently the equation of yield locus as defined in [1] are based on the normality requirements of the plastic strain vector with strain-hardening phenomena incorporated in the plastic potential function, additional imposition of strain-hardening through a constant yield stress is unnecessary. Because the terms included in the plastic potential function [5,6] consists of deviatoric stress

invariant and the basic soil mechanics material properties (M, λ, μ) associated with the mean pressure the later contributions in the plastic potential function must provide strain-hardening behavior in the sense of classical incremental theory of plasticity. This argument leads to the standard manner of handling the plastic potential function in that the variation of the plastic potential function simply depends on the second deviatoric stress invariant and the strain-hardening parameter. If such variation is equal to zero we have a neutral loading, and the positive and negative values would indicate loading and unloading, respectively. The positive change of this potential function, therefore, shifts the yield locus in the deviatoric-mean stress space whose projection back to the void ratio - mean stress space lies entirely on the yield surface at all times.

The constitutive relationships and the finite element equations are derived as demonstrated earlier [5,6]. The plastic tangent stiffness matrix is updated for small increments of loading. The repetitive solution of the equilibrium equations continues until the total load is reached. Numerical examples for the plate-bearing and cone-penetrometer are presented to evaluate correctness of the procedure. Comparisons with test results indicate close agreements.

I-2. YIELD CRITERIA AND PLASTIC STIFFNESS

We record here the following basic assumptions of the critical state theory: (1) the soil material is continuously distributed over its whole volume with its behavior described by a macroscopic model; (2) the mechanical behavior of cohesive and cohesion-

less soil depends only on effective stresses independent of the presence or absence of pore pressures. The consequences of these assumptions lead to a complete description of soil behavior in a space of void ratio e , mean pressure p , and deviatoric pressure q . The deviatoric and volumetric strains corresponding to q and p along the yield locus are then related by means of the normality principle of plasticity theory as shown in Figure 1.

The mathematical model of pre-yield behavior may be based on the simple assumption of complete rigidity or elasticity, although some evidence exists of irrecoverable plastic shear distortion in this range [1]. For simplicity we may use the elasticity theory for the range of elastic wall (Figure 1).

To deal with irrecoverable volumetric and deviatoric strains and recoverable volumetric strains we turn to the equation of yield locus,

$$\frac{p}{p_0} - \frac{M^2}{M^2 + \eta^2} = 0 \quad (1)$$

where $\eta = q/p$; p_0 is the mean pressure corresponding to $q = 0$; and M is the slope η at the critical state line,

$$M = \frac{6 \sin \varphi}{3 - \sin \varphi} \quad (2)$$

in which φ is the angle of internal friction.

The incremental plastic (irrecoverable) volumetric strain is

$$dv^{(P)} = - \frac{de^{(P)}}{1 + e} \quad (3)$$

The overall void ratio change along the isotropic compression curve is

$$de = -\lambda \frac{dp_0}{p_0} \quad (4)$$

whereas the incremental recoverable void ratio is given by

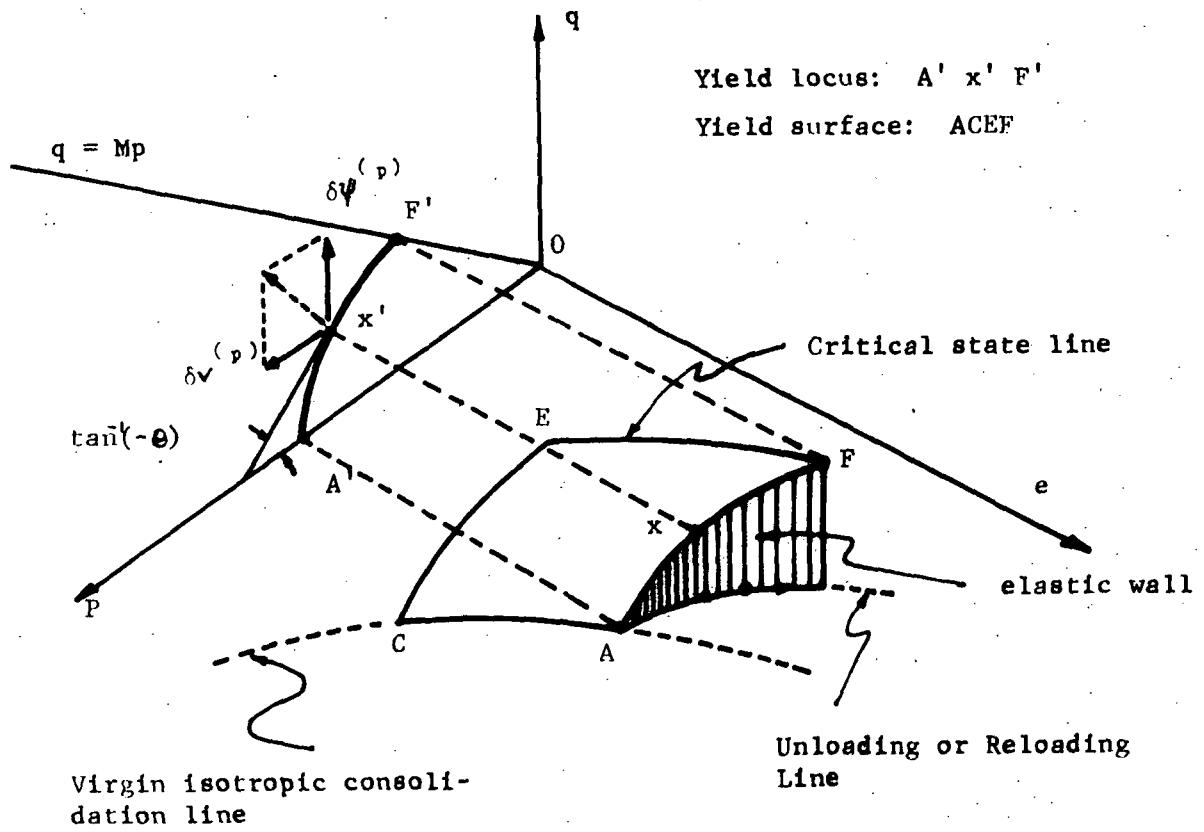


Figure 1: Yield Surface and Yield Locus (after Roscoe and Burland)

$$de^{(r)} = -\kappa \frac{dP_0}{P_0} \quad (5)$$

Here λ and κ are the compression index and swelling index, respectively.

The incremental irrecoverable void ratio is then obtained from (4) and

(5) as

$$de^{(p)} = -(\lambda - \kappa) \frac{dP_0}{P_0} \quad (6)$$

At this point we introduce the equation of yield surface in the form

[1]

$$\frac{p}{p_0} = \left(\frac{M^2}{M^2 + \eta^2} \right) \left(1 - \frac{\kappa}{\lambda} \right) \quad (7)$$

in which p_0 is the equivalent pressure corresponding to that void on the virgin isotropic consolidation line whose projection to the $p - q$ space is P_0 .

Therefore, setting $p_0 = P_0$ in (7) leads to

$$P_0 = p \left(1 + \frac{\eta^2}{M^2} \right)^{(1-\kappa/\lambda)} \quad (8)$$

Under triaxial compression, the second deviatoric stress invariant becomes

$$J = 1/3 (\sigma_{11} - \sigma_{33})^2 = 1/3 q^2 \quad (9)$$

which gives

$$q = \sqrt{3J} \quad (10)$$

Substituting (10) into (1) and rearranging yields

$$3J + p M^2 (P - P_0) = 0$$

or

$$3J - A^2 = 0 \quad (11)$$

where

$$A^2 = p M^2 (P - P_0) \quad (12)$$

It should be noted that (11) assumes the identical form as the plastic potential function $F(J, A)$ in the sense of classical incremental theory of plasticity,

$$F(J, A) = 3J - A^2 = 0 \quad (13)$$

The associated flow rule for the incremental plastic volumetric strain $dv^{(P)}$ and the incremental plastic deviatoric strain tensor $d\psi_{mn}^{(P)}$ may be written, respectively,

$$dv^{(P)} = \frac{\delta F}{\delta A} \frac{\delta A}{\delta p} d\lambda \quad (17)$$

$$d\psi_{mn}^{(P)} = \frac{\delta F}{\delta J} \frac{\delta J}{\delta \sigma_{mn}} d\lambda \quad (18)$$

in which $d\lambda$ is the positive constant. Here $dv^{(P)}$ may also be expressed in an alternate form from (3) and (6),

$$dv^{(P)} = \frac{\lambda - \kappa}{1 + e} \frac{dP_0}{P_0} \quad (19)$$

Equating (17) and (19) and using (16) give

$$dv^{(p)} = \frac{(\lambda - \kappa) dp_0}{(1+e)p_0} \frac{\frac{\partial F}{\partial A} \frac{\partial A}{\partial p}}{\frac{\partial F}{\partial A} \frac{\partial A}{\partial p} + \frac{\partial F}{\partial J} dJ} = \frac{-(\lambda - \kappa)}{(1+e)p_0} \left(\frac{\partial F}{\partial J} dJ + \frac{\partial F}{\partial A} \frac{\partial A}{\partial p} dp \right) \quad (20)$$

The incremental total plastic strain tensor is given by

$$d\gamma_{mn}^{(p)} = d\psi_{mn}^{(p)} + 1/3 dv_{mn}^{(p)} \delta_{mn} \quad (21)$$

in which δ_{mn} is the Kronecker delta. Using (17) through (20) in (21) yields

$$d\gamma_{mn}^{(p)} = B_{mn} R_{\alpha\beta} d\sigma^{\alpha\beta} \quad (22)$$

where

$$B_{mn} = - \frac{\left(3 \frac{\partial J}{\partial \sigma_{mn}} + \frac{1}{3} \frac{\partial F}{\partial A} \frac{\partial A}{\partial p} \delta_{mn} \right) (\lambda - \kappa)}{p_0 (1+e) \frac{\partial F}{\partial A} \frac{\partial A}{\partial p} + \frac{\partial F}{\partial J} \frac{\partial A}{\partial p_0}} \quad (23)$$

$$R_{\alpha\beta} d\sigma^{\alpha\beta} = \frac{\partial F}{\partial J} dJ + \frac{\partial F}{\partial A} \frac{\partial A}{\partial p} dp \quad (24)$$

also,

$$\frac{\partial F}{\partial J} dJ = 3 \frac{\partial J}{\partial \sigma_{mn}} d\sigma^{mn} = S_{mn} d\sigma^{mn} \quad (25a)$$

$$\frac{\partial F}{\partial A} \frac{\partial A}{\partial p} dp = (2pM^2 - p_0 M^2) dp \quad (25b)$$

$$\frac{\partial F}{\partial A} \frac{\partial A}{\partial p_0} = -M^2 p \quad (25c)$$

Substituting (25) into (23) gives

$$B_{mn} = \frac{S_{mn} + a \delta_{mn}}{b} \quad (26)$$

in which

$$a = \frac{M^2}{3} (2p - p_0) \quad (27)$$

$$b = 3a(1+e)M^2 p^2 p_0 / (\lambda - \kappa) \quad (28)$$

Similarly, $R_{\alpha\beta} d\sigma^{\alpha\beta}$ in (24) is given by

$$R_{\alpha\beta} d\sigma^{\alpha\beta} = (S_{\alpha\beta} + a\delta_{\alpha\beta}) d\sigma^{\alpha\beta} \quad (29)$$

where

$$S_{11} = 2\sigma_{11} - \sigma_{22} - \sigma_{33}$$

$$S_{22} = 2\sigma_{22} - \sigma_{11} - \sigma_{33}$$

$$S_{33} = 2\sigma_{33} - \sigma_{11} - \sigma_{22}$$

$$S_{12} = 6\sigma_{12}, S_{23} = 6\sigma_{23}, S_{31} = 6\sigma_{31}$$

The incremental total strain tensor dY_{mn} is the sum of the incremental elastic strain tensor $dY_{mn}^{(e)}$ and the incremental plastic strain tensor $dY_{mn}^{(p)}$.

Therefore,

$$dY_{mn}^{(e)} = dY_{mn} - dY_{mn}^{(p)} \quad (30)$$

The incremental total stress tensor $dY^{\alpha\beta}$ is then given by

$$d\sigma^{\alpha\beta} = D_{(e)}^{\alpha\beta mn} dY_{mn}^{(e)} \quad (31)$$

in which $D_{(e)}^{\alpha\beta mn}$ is the standard elasticity matrix. Substituting (30) and (22) into (31) yields

$$d\sigma^{\alpha\beta} = D^{\alpha\beta mn} (dY_{mn} - B_{mn} R_{1j} d\sigma^{1j}) \quad (32)$$

In view of (14b) and (24), and (32), we obtain

$$R_{rs} \left[D_{(e)}^{rs mn} (dY_{mn} - B_{mn} R_{\alpha\beta} d\sigma^{\alpha\beta}) \right] + \frac{\partial F}{\partial A} \frac{\partial A}{\partial P_0} dP_0 = 0$$

or

$$R_{rs} \left[D_{(e)}^{rs mn} (dY_{mn} - B_{mn} R_{\alpha\beta} d\sigma^{\alpha\beta}) \right] - R_{\alpha\beta} d\sigma^{\alpha\beta} = 0$$

from which

$$R_{\alpha\beta} d\sigma^{\alpha\beta} = \frac{R_{rs} D^{rs mn} dY_{mn}}{1 + R_{rs} B_{mn} D^{rs mn}} \quad (33)$$

Substituting (33) into (32) gives

$$d\sigma^{\alpha\beta} = \left(D_{(e)}^{\alpha\beta mn} + D_{(p)}^{\alpha\beta mn} \right) dY_{mn} \quad (34)$$

where

$$D_{(p)}^{\alpha\beta mn} = - \frac{D^{\alpha\beta kl} B_{kl} R_{1j} D^{1jmn}}{1 + B_{rs} R_{\alpha\beta} D^{rs mn}} \quad (35)$$

which is identical to the form obtained by the authors earlier [5,6].

Now, the yield criterion equation (14) is written as

$$dF = R_{\alpha\beta} d\sigma^{\alpha\beta} - M^2 p dp_0 \quad (36)$$

where dp_0 can be determined from (8),

$$dp_0 = g dp + h S_{ij} d\sigma_{ij} \quad (37)$$

in which

$$g = \left(1 + \frac{3J}{M^2 p^2}\right)^{(1-\kappa/\lambda)} - \frac{6J}{M^2 p^2} \left(1 - \frac{\kappa}{\lambda}\right) \left(1 + \frac{3J}{M^2 p^2}\right)^{(-\kappa/\lambda)} \quad (38a)$$

$$h = \frac{1}{M^2 p^2} \left(1 - \frac{\kappa}{\lambda}\right) \left(1 + \frac{3J}{M^2 p^2}\right)^{(-\kappa/\lambda)} \quad (38b)$$

Substituting these in (36) yields

$$dF = \left[S_{\alpha\beta} + a \delta_{\alpha\beta} - M^2 p \left(\frac{1}{3} g \delta_{\alpha\beta} + h S_{\alpha\beta} \right) \right] d\sigma^{\alpha\beta} \quad (39)$$

which is then used for determining the status of loading, neutral loading, and unloading as defined in (15 a, b, c).

I-3. APPLICATIONS

I.3.1 Plate Bearing

Based on the definition of yielding given by (15) the finite element computer program was written to solve boundary value problems. The program listing and data input format are given in Appendix 1. and Appendix 2, respectively.

Figure 2 shows the geometry of a plate bearing problem. The load-displacement curves for center of plate are shown in Figure 3 comparing the experimental results of Namiq [8]. It should be noted that the plane strain conditions of Namiq's experiments with a square box are approximated here in the analysis by an equivalent axis-symmetric cylindrical box. The material constants given by Namiq are angle of internal friction $\phi = 35^\circ$, initial void ratio $e = 0.875$, initial density $\gamma = 0.0147 \text{ N/cc}$. Other constants needed in this analysis are listed in Figure 3. It is seen that the load-displacement curve for the compression index $\lambda = 0.05$ follows very closely the experimental results whereas $\lambda = 0.13$ gives slightly larger displacements. It is interesting to note that from the void ratio-pressure curves given by Namiq the compression index can be estimated indeed to be approximately 0.05. Here the swelling index $\kappa = 0.003$ is used for both cases. For elastic behavior the soil modulus $E_s = 10 \text{ N/cm}^2$ and Poisson's ratio $\nu_s = .45$ are used.

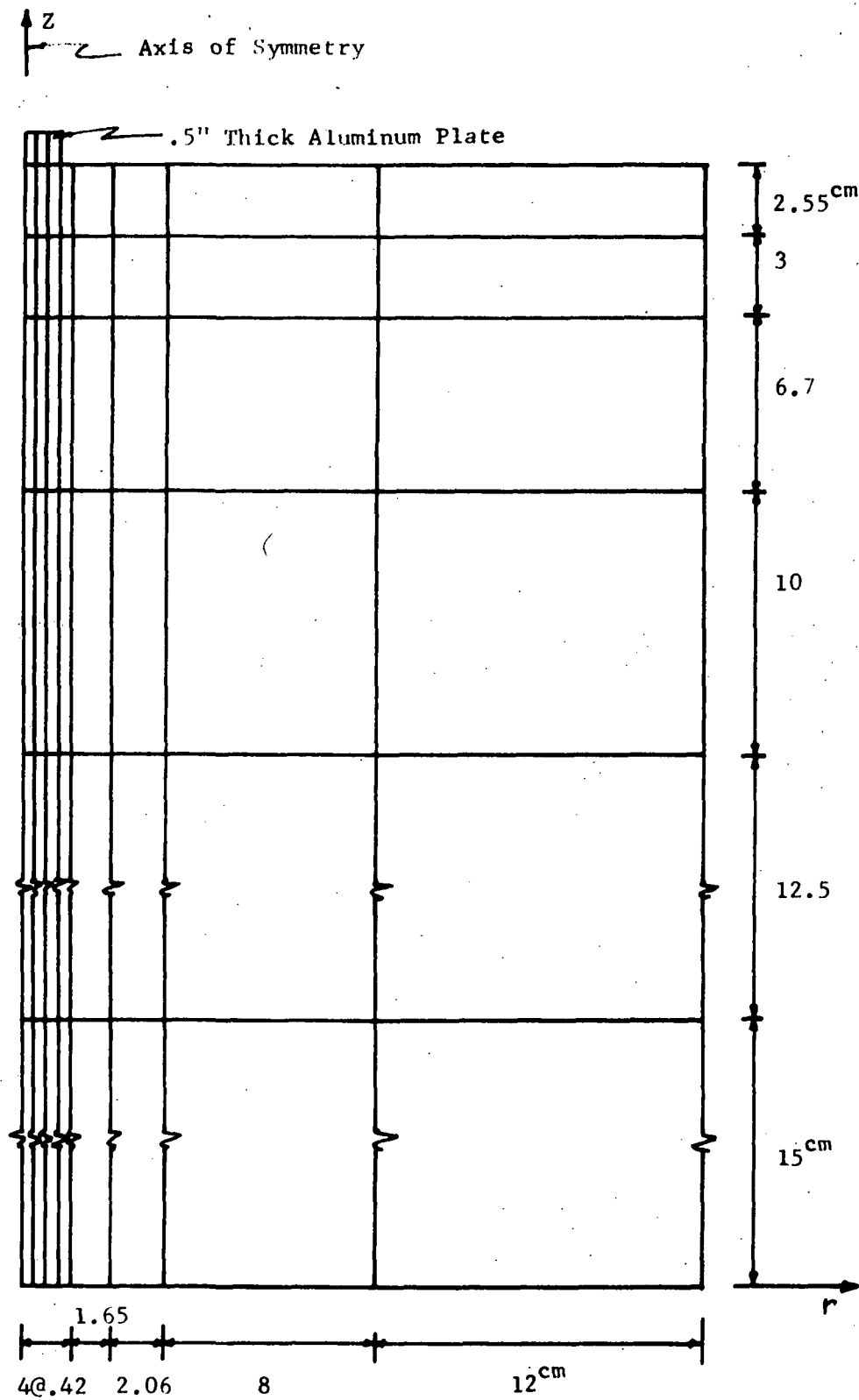


Figure 2: Plate Bearing Geometry

| Legend | ϕ | e | ν | μ | λ | E_s | ν_s |
|-----------|------------|------|------------|-------|-----------|--------------------------|---------|
| ———— | 35° | .875 | .0147 N/cc | NA | NA | NA | NA |
| ----- | " | " | " | .003 | .05 | 10^{10} N/cm^2 | .45 |
| - · - · - | " | " | " | " | .13 | " | " |

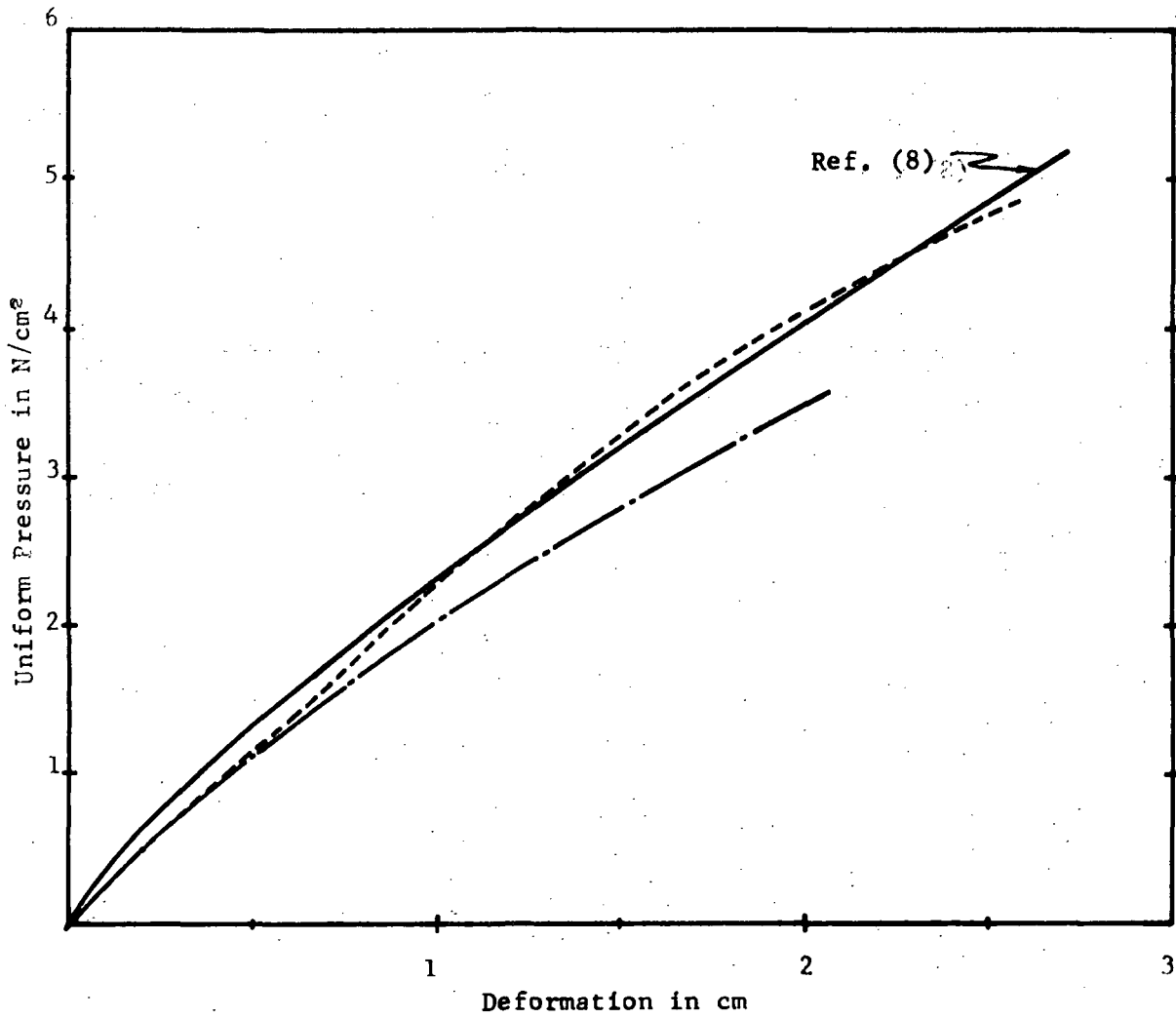


Figure 3: Deformation at Center of Plate

Deformed shapes of the finite elements adjacent to the bearing plate are shown in Figure 4 for the loading increments at $F = 2.5 \text{ N/cm}^2$ and $F = 5 \text{ N/cm}^2$. These results correspond to $\lambda = 0.05$ which gives the same displacement at the center of plate as Namiq. Unfortunately, however, no further comparison can be made as Namiq does not show such deformed shapes in his experimental results.

I.3.2 Cone-Penetrometer

The geometry for a cone-penetrometer problem is shown in Figure 5. Experiments for the cone-penetrometer were undertaken and the test set-up is shown in Figure 6. Both smooth and rough aluminum cones were used and loaded through the lunar soil simulants under the strain-controlled loading devices. These measurements are plotted in Figure 7 and compared with analytical results. The axisymmetric interface elements developed by Chung and Lee [5] are used to model contact areas between the cone and soil. Because of the lunar soil simulants being extremely soft compared with the metal cone the shear modulus and rotational modulus for the interface elements were set equal to zero. Experimentally determined material constants for the lunar soil simulants used in the tests are also given in Figure 7. The same material constants were used in the analysis with the exceptions of soil modulus $E_s = 10 \text{ N/cm}^2$ and Poisson's ratio $\nu_s = .45$. The analytical solution gives results somewhere between the rough and smooth cones.

The deformed geometry of soil is shown in Figure 8. For excessive alterations of finite elements in shape it would appear that

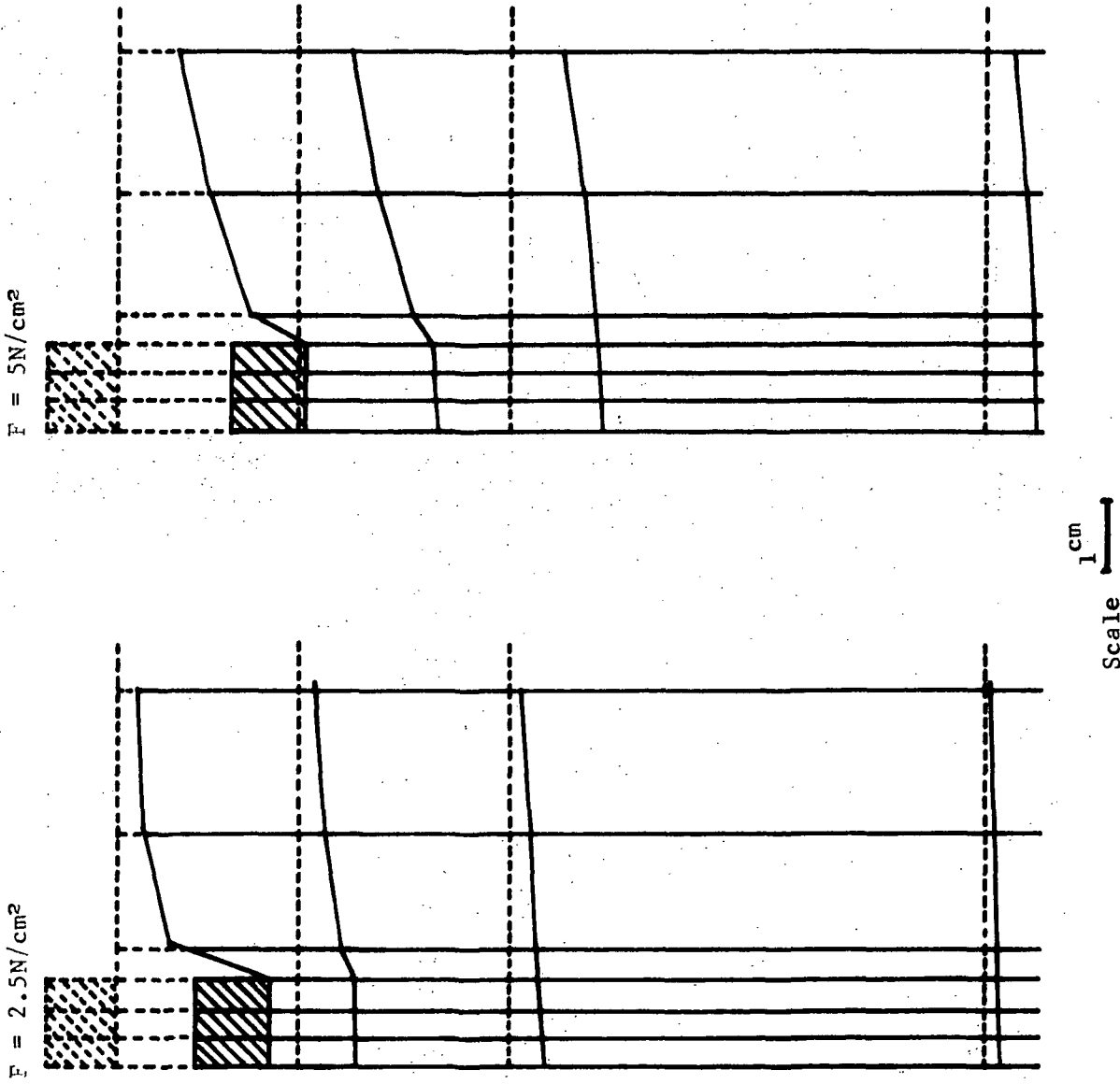


Figure 4: Deformed Geometry

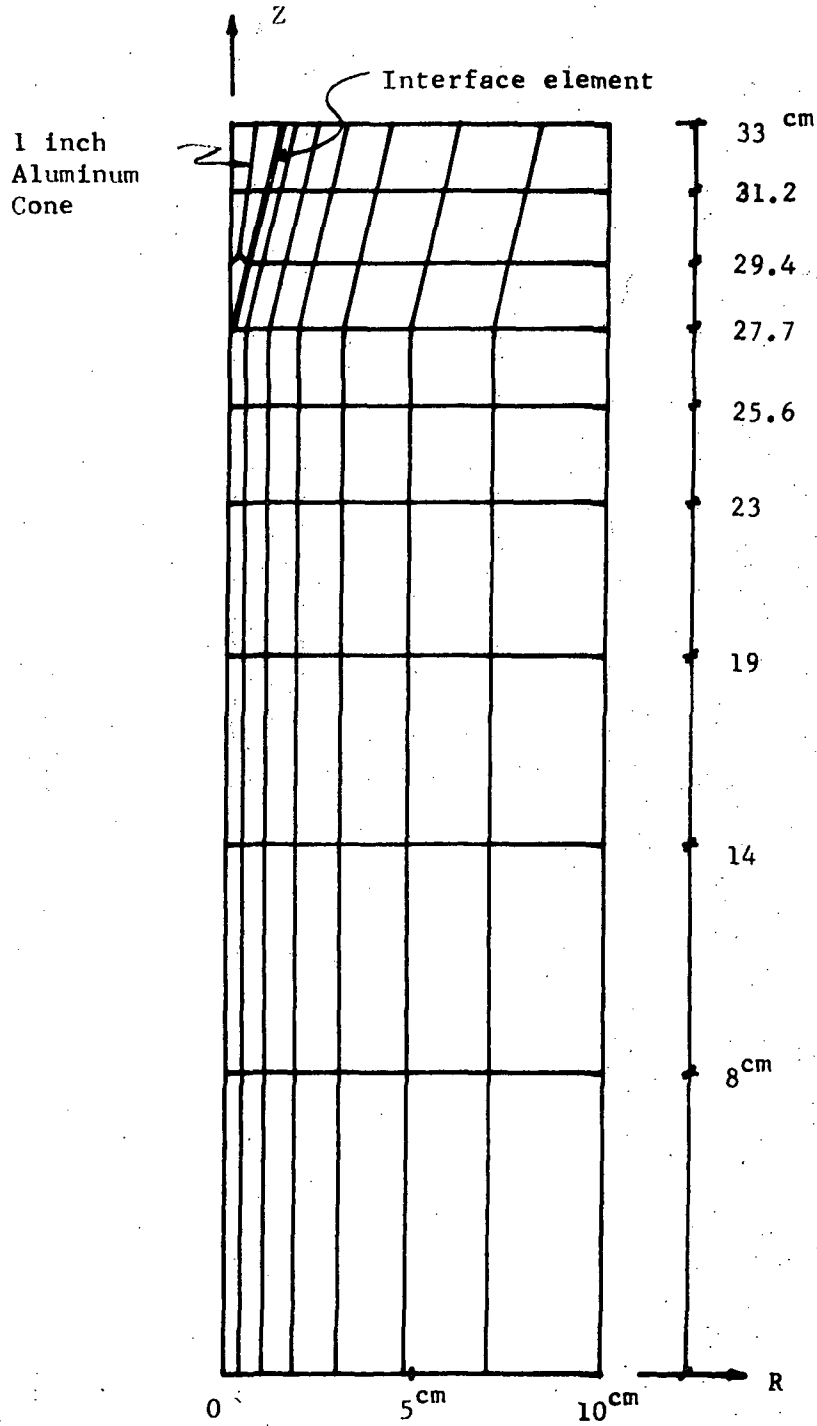
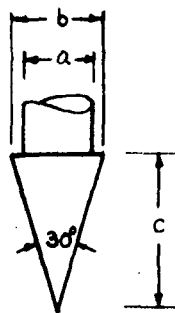
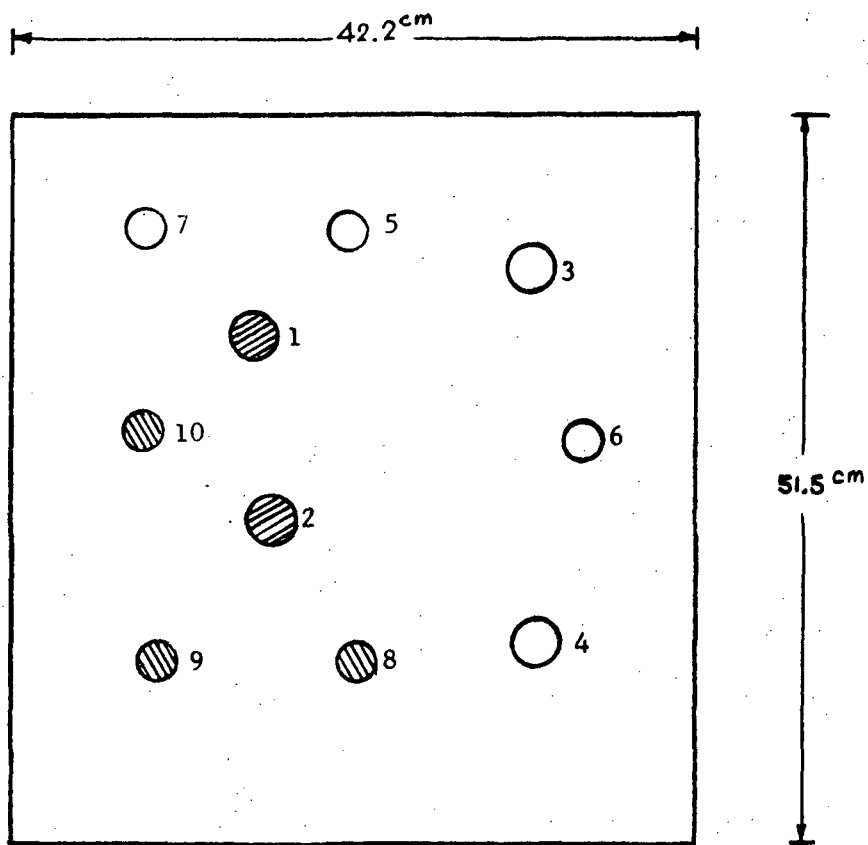


Figure 5: Cone-Penetrometer Geometry



- | | | |
|--------|--------------------------------|--|
| 1,2 | 1 inch Rough Cone | } $a=2.54^{cm}$, $b=2.82^{cm}$, $c=5.3^{cm}$ |
| 3,4 | 1 inch Smooth Cone | |
| 5,6,7 | $\frac{1}{2}$ inch Smooth Cone | } $a=1.7^{cm}$, $b=2^{cm}$, $c=3.9^{cm}$ |
| 8,9,10 | $\frac{1}{2}$ inch Rough Cone | |

Figure 6: Cone-Penetrometer Tests

| Legend | Description |
|--------|--|
| —◇— | Average of 1/2" Rough Cone Tests |
| —○— | " " " " " |
| —▽— | " " 1/2" Smooth Cone Tests |
| —△— | " " " " " |
| | Finite Element Solution (1), $\lambda = .07$ * |
| ----- | " " (2), $\lambda = .13$ * |

* Other constants used are:
 $\alpha = .006$, $\phi = 35^\circ$, $e = .76$, $\nu = .0157$ N/cc

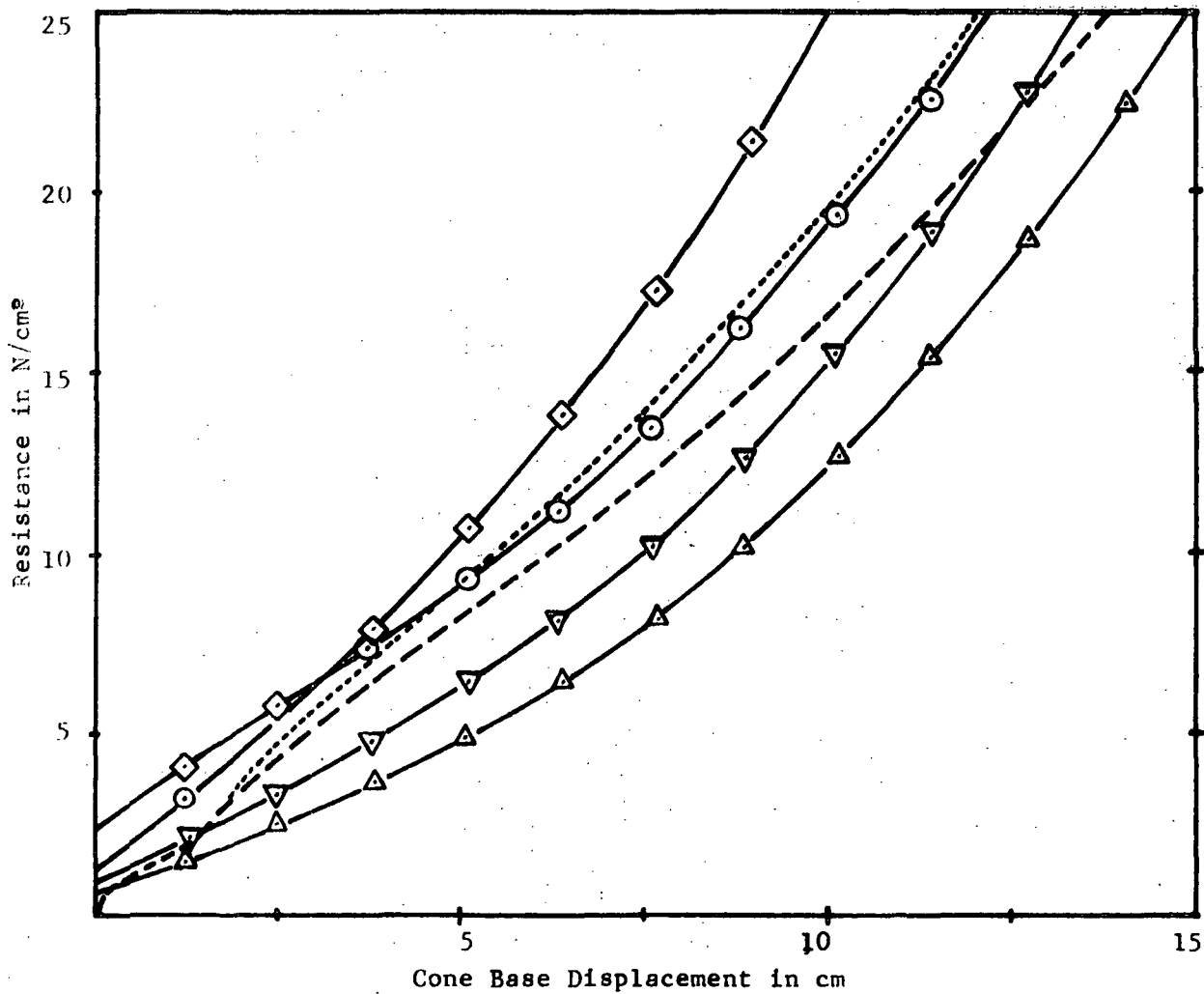


Figure 7: Force-Displacement Curves for Cone Penetrometer

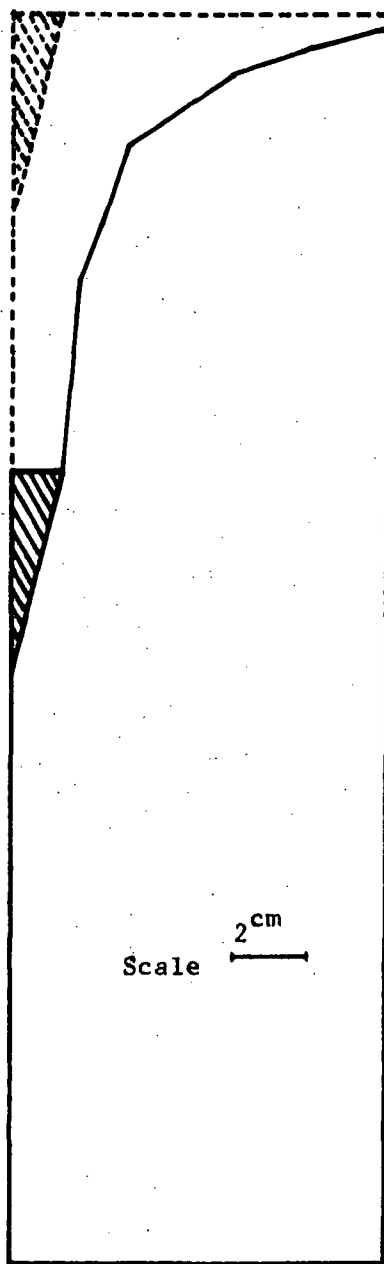


Figure 8: Deformed Configuration for Finite Element Solution (1) at $F = 25 \text{ N/cm}^2$

renumbering of nodes is necessary to update the stiffness matrix based on new geometry. It is believed that such treatment would improve the solution considerably.

I-4. CONCLUSIONS

A new definition of loading and unloading along the yield surface of Roscoe and Burland is introduced. This is done by noting the strain-hardening parameter in the plastic potential function. With the differential of the plastic potential function with respect to the second deviatoric stress invariant and the strain-hardening parameter being positive or negative the manner of loading and unloading is clearly determined. This is an improvement from the previous definition of yielding through a constant yield stress.

The forms of plastic stiffness matrix and the finite element equations, however, are unchanged. Applications of the present analytical formulation to a number of boundary value problems are presented. The analytical results for the plate bearing and cone-penetrometer problems indicate good agreements with the experimental results.

Our ultimate goal is to characterize the material parameters of the lunar soil. Such a task depends on correct constitutive relationships and a computational scheme which provides the results of load-deformation. With this facility available exhaustive computer runs

for various combinations of material constants are to be compared with the data brought back from the lunar exploration. To this end the present study has provided the basic analytical tool to prepare for such an undertaking.

REFERENCES

1. Roscoe, K. H. and Burland, J. B. "On the Generalized Stress-Strain Behavior of Wet Clay", Symposium on Engineering Plasticity, Cambridge University Press, 1968, pp. 535-609.
2. Schofield, A. N., and Wroth, C. P., Critical State Soil Mechanics, McGraw-Hill Book Co., Inc., New York, 1968.
3. Smith, I. M. and Kay, S., "Stress Analysis of Contractive or Dilative Soil", Journal of the Soil Mechanics and Foundations Division, ASCE, Vol. 97, No. SM7, 1971, pp. 981-997.
4. Zienkiewicz, O. C. "The Finite Element Method Applied to Soil and Other Porous Media", NATO Advanced Study Institute on Finite Element Methods in Continuum Mechanics, Lisbon, September, 1971.
5. Chung, T. J. and Lee, J. K. "Incremental Plasticity Theory Applied to Boundary Value Problems in Soil", Proc. of W. E. S. Symposium on Finite Element Applications to Geotechnical Engineering, Vicksburg, April 1-3, 1972.
6. Chung, T. J., Costes, N. C., and Lee, J. K., "Boundary Value Problems with Incremental Plasticity in Granular Media", Discussions at International Symposium on Foundations of Plasticity, Warsaw, Poland, August 27 - September 1, 1972, also submitted to Archives of Mechanics.
7. Drucker, D. C., "Some Implications of Work Hardening and Ideal Plasticity", Quarterly of Applied Mathematics, Vol. 7, No. 4, January, 1950, pp. 411-418.

REFERENCES (cont.)

8. Namiq, L. I., "Stress Deformation Study of Simulated Lunar Soil", Ph.D. thesis, University of California, Berkeley, 1970.

APPENDIX 1

COMPUTER PROGRAM LISTING
(Static Analysis - Axisymmetric)

WELT,IDL DECK,,SYSTOP
ELT 005-11/24-14:39

```
000001 000 WELT,SIH NASA*TPF$.MAIN,,132656133010
000002 000 C-----00000100
000003 000 C 00000200
000004 000 C THE FINITE ELEMENT ANALYSIS OF AXISYMMETRIC SOIL MEDIUM 00000300
000005 000 C BY A SOIL PLASTICITY THEORY 00000400
000006 000 C 00000500
000007 000 C-----00000600
000008 000 C 00000700
000009 000 PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 00000800
000010 000 C 00000900
000011 000 COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,NCYCL,EPSON 00001000
000012 000 COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4), 00001100
000013 000 * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),YPEC(4,4),TYPEE(4,4), 00001200
000014 000 * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL 00001300
000015 000 COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DEI,DE2 00001400
000016 000 COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHR,IHRI,LT,LAST 00001500
000017 000 COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP 00001600
000018 000 COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS, 00001700
000019 000 1 FINC,FN,ULOAD,FEL,PMAX,DLMAX 00001800
000020 000 COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8), 00001900
000021 000 1 R(NODS),Z(NODS),TUDIS(MAX) 00002000
000022 000 COMMON /BLK7/ USTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS), 00002100
000023 000 1 AOJ(NELS) 00002200
000024 000 COMMON/BLK8/INCR,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,PP 00002300
000025 000 COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRIGD 00002400
000026 000 COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20), 00002500
000027 000 1 POR(20),DELTS,EC,XNUC,DELD 00002600
000028 000 COMMON /BLK11/ VOID(NELS),DGAM(NELS,4) 00002700
000029 000 COMMON /BLK12/ SIGMX(NELS),DEP(NELS),EP(NELS),DEGST2(NELS),ES 00002800
000030 000 COMMON /GMTRY/ RO(NODS),ZO(NODS) 00002900
000031 000 C-----00003000
000032 000 C 00003100
000033 000 NTAPE = 2 00003200
000034 000 C 00003300
000035 000 C-----00003400
000036 000 C 00003500
000037 000 CALL SETUP 00003600
000038 000 C 00003700
000039 000 C INITIALIZES NECESSARY CONSTANTS FOR INTEGRATION SCHEME. 00003800
000040 000 C-----00003900
000041 000 C 00004000
000042 000 CALL INPUT(NTAPE) 00004100
000043 000 ISHEAR = 1 00004200
000044 000 C 00004300
000045 000 C 00004400
000046 000 FN = FEL 00004500
000047 000 C 00004600
000048 000 C-----00004700
000049 000 C 00004800
000050 000 C START MAIN ITERATION LOOP. 00004900
000051 000 C 00005000
000052 000 C-----00005100
000053 000 DO 990 NI = 1,NINCR 00005200
000054 000 C 00005300
000055 000 FN = FN + FINC 00005400
```


| | | | |
|--------|-----|---|----------|
| 000056 | 000 | ITER = 0 | 00005500 |
| 000057 | 000 | | 00005600 |
| 000058 | 000 | C IF(NI.EQ.1) GO TO 950 | 00005700 |
| 000059 | 000 | ISHEAR = NI | 00005800 |
| 000060 | 000 | CALL ZERO(XK,NF,1) | 00005900 |
| 000061 | 000 | REWIND NTAPE | 00006000 |
| 000062 | 000 | C | 00006100 |
| 000063 | 000 | DO 940 NEL = 1,NELEM | 00006200 |
| 000064 | 000 | IC = IJKL(NEL,1) | 00006300 |
| 000065 | 000 | JC = IJKL(NEL,2) | 00006400 |
| 000066 | 000 | KC = IJKL(NEL,3) | 00006500 |
| 000067 | 000 | LC = IJKL(NEL,4) | 00006600 |
| 000068 | 000 | NF1 = NEL - NRIGD | 00006700 |
| 000069 | 000 | IF(NEL.LE.NRIGD) GO TO 938 | 00006800 |
| 000070 | 000 | SIGZ = STRS(NEL,1) + DSTRS(NEL,1) / 2. | 00006900 |
| 000071 | 000 | SIGX = STRS(NEL,2) + DSTRS(NEL,2) / 2. | 00007000 |
| 000072 | 000 | SIGY = STRS(NEL,3) + DSTRS(NEL,3) / 2. | 00007100 |
| 000073 | 000 | TAUZR = STRS(NEL,4) + DSTRS(NEL,4) / 2. | 00007200 |
| 000074 | 000 | IF(NBC.NE.U.AND.NF1.LE.NBC) GO TO 899 | 00007300 |
| 000075 | 000 | C----- | 00007400 |
| 000076 | 000 | C | 00007500 |
| 000077 | 000 | CALL DMATRX(NI) | 00007600 |
| 000078 | 000 | C | 00007700 |
| 000079 | 000 | C CALCULATE STRESS DEPENDENT MATERIAL PROPERTY MATRIX (D). | 00007800 |
| 000080 | 000 | C----- | 00007900 |
| 000081 | 000 | C | 00008000 |
| 000082 | 000 | GO TO 837 | 00008100 |
| 000083 | 000 | C | 00008200 |
| 000084 | 000 | 938 DO 400 I = 1,4 | 00008300 |
| 000085 | 000 | DO 400 J = 1,4 | 00008400 |
| 000086 | 000 | DMAT(NEL,I,J) = DE(NEL,I,J) | 00008500 |
| 000087 | 000 | 400 D(I,J) = DMAT(NEL,I,J) | 00008600 |
| 000088 | 000 | 837 CALL STIFF2(NI,NTAPE) | 00008700 |
| 000089 | 000 | GO TO 838 | 00008800 |
| 000090 | 000 | C | 00008900 |
| 000091 | 000 | 899 CALL FRICTN(IC,KC,NEL,NF1,ISHEAR,VOIDI) | 00009000 |
| 000092 | 000 | C | 00009100 |
| 000093 | 000 | 838 CALL ASSEMB(NEL,NF1) | 00009200 |
| 000094 | 000 | 940 CONTINUE | 00009300 |
| 000095 | 000 | 950 CONTINUE | 00009400 |
| 000096 | 000 | ITER = ITER + 1 | 00009500 |
| 000097 | 000 | INCR = ITER | 00009600 |
| 000098 | 000 | C | 00009700 |
| 000099 | 000 | CALL DISPL(NFREE,NI,INODE,INCR) | 00009800 |
| 000100 | 000 | C | 00009900 |
| 000101 | 000 | DO 340 I = 1,INODE | 00010000 |
| 000102 | 000 | JJ = I * 2 | 00010100 |
| 000103 | 000 | II = JJ - 1 | 00010200 |
| 000104 | 000 | Z(1) = Z0(1) + TOTDIS(II) + XK(LAST+II) | 00010300 |
| 000105 | 000 | 340 R(1) = R0(1) + TOTDIS(JJ) + XK(LAST+JJ) | 00010400 |
| 000106 | 000 | C | 00010500 |
| 000107 | 000 | CALL STRAIN(NI,ITER) | 00010600 |
| 000108 | 000 | C | 00010700 |
| 000109 | 000 | C SUMMING OF STRESSES AND DISPLACEMENTS FOR EACH INCREMENTAL STEP | 00010800 |
| 000110 | 000 | DO 310 IIT = 1,NFREE | 00010900 |
| 000111 | 000 | 310 TOTDIS(IIT) = TOTDIS(IIT) + XK(IIT+LAST) | 00011000 |
| 000112 | 000 | DO 329 I = 1,NELEM | 00011100 |

```

000113      000      DO 329 J = 1,4
000114      000      329 STRS(I,J) = STRS(I,J) + DSTRS(I,J)
000115      000      C
000116      000      WRITE(6,689) NINCR,NI, FN
000117      000      DO 320 I = 1,INODE
000118      000      II = (I-1) * 2 + 1
000119      000      JJ = II + 1
000120      000      320 WRITE(6,690) I,(TOTDIS(LX),LX = II,JJ)
000121      000      WRITE(6,691)
000122      000      DO 330 I = 1,NELEM
000123      000      330 WRITE(6,692) I,(STRS(I,J),J=1,4)
000124      000      IF(FN.GT.PMAX) STOP LOADMX
000125      000      990 CONTINUE
000126      000      C
000127      000      C
000128      000      500 FORMAT(10I5)
000129      000      600 FORMAT(// ' ITER',15,' DMAX',E12.5,' DE2',E12.5)
000130      000      689 FORMAT(1H1,10X,' TOTAL DISPLACEMENT NO. OF INCREMENTAL STEPS
000131      000      . *',2I5//5X,'NODE',5X,'Z - DISPL',20X,'R - DISPL',5X,'FN =',E12.5/)
000132      000      690 FORMAT(4X,15,E15.7,10X,E15.7)
000133      000      691 FORMAT(///,10X,' TOTAL STRESSES//5X,'ELEM',5X,'SIGMA - Z',T28,
000134      000      1'SIGMA - R',T42,'TANGENTIAL',T58,'TAU - ZR')
000135      000      692 FORMAT(18,4F14.6)
000136      000      693 FORMAT(// ' TAU, SIG, RAT, DELTS, ISHEAR, ',4E12.5,I5//)
000137      000      694 FORMAT(//20X,'NEW GEOMETRY AT THE END OF LOAD INCR.',I5//)
000138      000      695 FORMAT(110,2F15.6)
000139      000      STOP
000140      000      END
000141      000      WELT,SIH NASA*TPFS,INPUT,,132661133010
000142      000      SUBROUTINE INPUT(NTAPE)
000143      000      C-----
000144      000      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
000145      000      C-----
000146      000      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,NCYCL, EPSLON
000147      000      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000148      000      * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
000149      000      * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL
000150      000      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2
000151      000      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHR,IHBI,LT, LAST
000152      000      COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP
000153      000      COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS,
000154      000      1 FINC, FN, ULOAD, FEL, PMAX, ULMAX
000155      000      COMMON /BLK6/ SIGR, SIGZ, SIGT, TAUZR, D(4,4), STIFF(8,8), KK(NELS,8),
000156      000      1 R(NODS), Z(NODS), TOTDIS(MAX)
000157      000      COMMON /BLK8/ INCR, PDEPTH(NELS), VOIDI, ALAMDA, DEPTH, XMS
000158      000      COMMON /BLK9/ PI, SMALLK, CK, BETA, PO, NFREE, NELST, ICASE, NRIGD
000159      000      COMMON /BLK10/ FRCK(20,8,8), TR(20,8,8), XXL(20), DZZ(20), DRR(20),
000160      000      1 POR(20), DELTS, EC, XNUC, DELD
000161      000      COMMON /BLK11/ VOID(NELS), DGAM(NELS,4)
000162      000      COMMON /BLK12/ SIGMX(NELS), DEP(NELS), EP(NELS), DEQST2(NELS), ES
000163      000      COMMON /GMTRY/ RO(NODS), ZO(NODS)
000164      000      C-----
000165      000      REWIND NTAPE
000166      000      READ(5,510) (TITLE(I), I=1,20)
000167      000      READ(5,500) INODE,NELEM,NAPC,NBC,NINCR,NCYCL, ICASE,NRIGD,NULOAD
000168      000      READ(5,530) YSTRS,DELL,ZETA,PMAX,DI,MAX
000169      000      READ(5,511) DZI,DRI

```

```

00011200
00011300
00011400
00011500
00011600
00011700
00011800
00011900
00012000
00012100
00012200
00012300
00012400
00012500
00012600
00012700
00012800
00012900
00013000
00013100
00013200
00013300
00013400
00013500
00013600
00013700
00013800
00013900
00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800

```

| | | | | |
|--------|-----|---|--|----------|
| 000170 | 000 | C | DZI = SHEAR MOD. FOR INTERFACE ELEMENTS. | 00002900 |
| 000171 | 000 | C | DKI = NORMAL FOR INTERFACE ELEMENTS. | 00003000 |
| 000172 | 000 | | READ(5,511) EC,XNUC,ES,XNUS | 00003100 |
| 000173 | 000 | | READ(5,530) PI,SMALLK,XI,VOIDI,PO,DEPTH,ALAMDA,EPSLON | 00003200 |
| 000174 | 000 | | WRITE(6,530)PI,SMALLK,XI,VOIDI,PO,DEPTH,ALAMDA,EPSLON | 00003300 |
| 000175 | 000 | | WRITE(6,630) YSTRS,DELL | 00003400 |
| 000176 | 000 | | WRITE(6,631) ZETA,DZI,DRI | 00003500 |
| 000177 | 000 | | 631 FORMAT(// ZETA, MOD. FOR INTERFACE ELEM., SHEAR, NORMAL, | 00003600 |
| 000178 | 000 | | 1 3F15.6) | 00003700 |
| 000179 | 000 | | POP = EXP(1.-SMALLK/ALAMDA) | 00003800 |
| 000180 | 000 | | II = 0 | 00003900 |
| 000181 | 000 | | DO 101 I = 1,INODE | 00004000 |
| 000182 | 000 | | DO 101 J = 1,2 | 00004100 |
| 000183 | 000 | | II = II + 1 | 00004200 |
| 000184 | 000 | | 101 ID(I,J) = II | 00004300 |
| 000185 | 000 | | BETA=ALAMDA-SMALLK | 00004400 |
| 000186 | 000 | | PI=PI*3.14159/180. | 00004500 |
| 000187 | 000 | | EPSLON = EPSLON * 3.14159 / 180. | 00004600 |
| 000188 | 000 | | DELTS = TAN(EPSLON) | 00004700 |
| 000189 | 000 | | PI=SIN(PI) | 00004800 |
| 000190 | 000 | C | PI IS SINE(PI) | 00004900 |
| 000191 | 000 | | XM = 6.*PI/(3.-PI) | 00005000 |
| 000192 | 000 | | XMS = XM * XM | 00005100 |
| 000193 | 000 | | WRITE(6,600) (TITLE(I),I=1,20) | 00005200 |
| 000194 | 000 | | DO 100 I = 1,INODE | 00005300 |
| 000195 | 000 | | READ(5,520) Z(1),R(1),IZ,IR | 00005400 |
| 000196 | 000 | | ZO(I) = Z(I) | 00005500 |
| 000197 | 000 | | RO(I) = R(I) | 00005600 |
| 000198 | 000 | | IF(IZ.NE.0) ID(I,1) = 0 | 00005700 |
| 000199 | 000 | | IF(IR.NE.0) ID(I,2) = 0 | 00005800 |
| 000200 | 000 | | 100 WRITE(6,620) I,Z(1),R(1) ,IZ,IR | 00005900 |
| 000201 | 000 | | WRITE(6,501) INODE,NELEM,NAPC,NINCR,NCYCL | 00006000 |
| 000202 | 000 | | NFREE = INODE * 2 | 00006100 |
| 000203 | 000 | C | | 00006200 |
| 000204 | 000 | | WRITE(6,651) | 00006300 |
| 000205 | 000 | | READ(5,540) ((IJKL(NEL,J),J=1,4),NEL=1,NELEM) | 00006400 |
| 000206 | 000 | | WRITE(6,650) (NEL,(IJKL(NEL,J),J=1,4),NEL=1,NELEM) | 00006500 |
| 000207 | 000 | C | | 00006600 |
| 000208 | 000 | C | FIND HALF BAND WIDTH AND ACTUAL SIZE OF MATRIX (XK) | 00006700 |
| 000209 | 000 | C | | 00006800 |
| 000210 | 000 | | IMAX = 0 | 00006900 |
| 000211 | 000 | | DO 800 NEL = 1,NELEM | 00007000 |
| 000212 | 000 | | DO 700 I = 1,4 | 00007100 |
| 000213 | 000 | | IN = IJKL(NEL,I) | 00007200 |
| 000214 | 000 | | KK(NEL,I) = ID(IN,1) | 00007300 |
| 000215 | 000 | | 700 KK(NEL,I+4) = ID(IN,2) | 00007400 |
| 000216 | 000 | | DO 7999 I = 1,2 | 00007500 |
| 000217 | 000 | | II = I + 1 | 00007600 |
| 000218 | 000 | | DO 7999 J = II,4 | 00007700 |
| 000219 | 000 | | IDIF = IJKL(NEL,I) - IJKL(NEL,J) | 00007800 |
| 000220 | 000 | | IF(IDIF.LT.0) IDIF = -IDIF | 00007900 |
| 000221 | 000 | | 7999 IF(IDIF.GT.IMAX) IMAX = IDIF | 00008000 |
| 000222 | 000 | | 800 CONTINUE | 00008100 |
| 000223 | 000 | | IF(NBC.NE.0) READ(5,500) ((ID(I,J),J=1,2),I=1,NBC) | 00008200 |
| 000224 | 000 | C | IMAX = MAX DIFFERENCE IN ADJACENT NODE NO. | 00008300 |
| 000225 | 000 | | IHB = (IMAX + 1) * 2 | 00008400 |
| 000226 | 000 | | IHB1 = IHB - 1 | 00008500 |

```

000227 000      LT = IHB * IHBI / 2                      00008600
000228 000      LAST = LT + (NFREE - IHRI) * IHB      00008700
000229 000      WRITE(6,640) IMAX,IHB,LT,LAST        00008800
000230 000      C-----                                00008900
000231 000      IF(NRIGD.NE.0) CALL ELASTC(D,NRIGD,EC,XNUC) 00009000
000232 000      C-----                                00009100
000233 000      NLST = NELEM - NRIGD - NBC            00009200
000234 000      DO 900 NEL = 1,NELEM                  00009300
000235 000      IC = IJKL(NEL,1)                     00009400
000236 000      JC = IJKL(NEL,2)                     00009500
000237 000      KC = IJKL(NEL,3)                     00009600
000238 000      LC = IJKL(NEL,4)                     00009700
000239 000      IF(NEL.LE.NRIGD) GO TO 898            00009800
000240 000      NFT = NEL - NRIGD                      00009900
000241 000      IF(NFT.GT.NRC) GO TO 896              00010000
000242 000      DZL(NFT) = DZI                        00010100
000243 000      DRK(NFT) = DRI                        00010200
000244 000      CALL FRICTN(IC,KC,NEL,NFT,0,VOIDI)    00010300
000245 000      GO TO 899                              00010400
000246 000      896 IF(ICASE. NE. 0) GO TO 897         00010500
000247 000      C-----                                00010600
000248 000      CALL AREA(A,IC,JC,KC,LC,AREA)         00010700
000249 000      VOIDR = VOIDI                          00010800
000250 000      VOID(NEL) = VOIDR                      00010900
000251 000      DELNM1(NEL) = AREA                    00011000
000252 000      C-----                                00011100
000253 000      897 NELST = NRIGD + NBC + 1           00011200
000254 000      IF(NEL.EQ.NELST)CALL ELASTC(D,NLST,ES,XNUS) 00011300
000255 000      C-----                                00011400
000256 000      898 DO 111 I = 1,4                     00011500
000257 000      DO 111 J = 1,4                     00011600
000258 000      DE(NEL,I,J) = D(I,J)                 00011700
000259 000      111 DMAT(NEL,I,J) = D(I,J)           00011800
000260 000      C-----                                00011900
000261 000      CALL STIFF1(NTAPE)                     00012000
000262 000      CALL STIFF2(0,NTAPE)                   00012100
000263 000      899 CALL ASSEMB(NEL,NFT)                00012200
000264 000      C-----                                00012300
000265 000      C-----                                00012400
000266 000      900 CONTINUE                            00012500
000267 000      IF(NAPC.NE.0) CALL PTLOAD(NAPC,ULOAD)  00012600
000268 000      901 IF(NULOAD.NE.0) CALL EGLOAD(ULOAD) 00012700
000269 000      C-----                                00012800
000270 000      SCAL = NINCR                            00012900
000271 000      DO 200 I = 1,NFREE                    00013000
000272 000      200 APF(I) = APF(I) / SCAL             00013100
000273 000      FEL = 0.                               00013200
000274 000      FINC = ULOAD / SCAL                   00013300
000275 000      C-----                                00013400
000276 000      500 FORMAT(10I5)                       00013500
000277 000      501 FORMAT(//'      NUMBER OF NODES =',I5//  NUMBER OF ELEMENTS =',I00013600
000278 000      15//'      NUMBER OF APLIED CONCENTRATED LOADS =',I5//'      NUMBER OF 00013700
000279 000      2INCREMENTAL LOAD STEPS =',I5//'      NUMBER OF ITERATIONS PER EACH I00013800
000280 000      3INCREMENTAL LOADING =',I5//')          00013900
000281 000      510 FORMAT(20A4)                       00014000
000282 000      511 FORMAT(4F20.5)                     00014100
000283 000      520 FORMAT(2F10.4,2I5)                 00014200

```

```

000284 000 530 FORMAT(8F10.4) 00014300
000285 000 531 FORMAT(/' SINE PHI =',F10.4,' SMALLK =',F10.4,' KI =',00014400
000286 000 *F10.4,' PORCUSITY =',F10.4/' BETA =',F10.4,' OVERBURD00014500
000287 000 *EN PRESSURE =',F10.4,' DESIRED ACCURACY =',F10.4,' PERCENT.'/00014600
000288 000 * ' DEPTH OF SOIL MEDIA =',F10.4//) 00014700
000289 000 540 FORMAT(4I5) 00014800
000290 000 600 FORMAT(1H1,20X,20A4//30X,'COORDINATE VALUES'//T11,'NODE',T30,'Z-C00014900
000291 000 *ORD',T50,'R-COORD',T65,'0,IF FREE TO Z',3X,'0,IF FREE TO R'//) 00015000
000292 000 620 FORMAT(T10,I5,T25,F10.4,T45,F10.4,T65,2(I8,5X)) 00015100
000293 000 630 FORMAT(/' YIELD STRESS =',E12.5,' DELL =',E12.5//) 00015200
000294 000 640 FORMAT(/' IMAX,IMB,LT,LAST',4I10//) 00015300
000295 000 650 FORMAT(5I7) 00015400
000296 000 651 FORMAT(1H1,10X,'CONNECTIVITY'//) 00015500
000297 000 RETURN 00015600
000298 000 END 00015700
000299 000 WELT,SIH NASA*IPFS,DMATRX,,,132671133010
000300 000 SUBROUTINE DMATRX(NI) 00000100
000301 000 C----- 00000200
000302 000 PARAMETER NOUS=300,NELS=260,NF=20000,MAX=600 00000300
000303 000 C----- 00000400
000304 000 COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON 00000500
000305 000 COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4), 00000600
000306 000 * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4), 00000700
000307 000 * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL 00000800
000308 000 COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP 00000900
000309 000 COMMON /BLK5/ DE(NELS,4,4),SIGRA(NELS),DSIGRA(NELS),DELL,YSTRS, 00001000
000310 000 1 FINC,FN,ULOAD,FEL,PMAX,DLMAX 00001100
000311 000 COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8), 00001200
000312 000 1 R(NODS),Z(NODS),TOTOIS(MAX) 00001300
000313 000 COMMON /BLK7/ USTRS(NELS,4),DUM(NELS,10) 00001400
000314 000 COMMON/BLK8/INCR,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,XMS 00001500
000315 000 COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRIGD 00001600
000316 000 COMMON /BLK11/ VOID(NELS),DGAM(NELS,4) 00001700
000317 000 COMMON/BLK12/ SIGMX(NELS),DEP(NELS),EP(NELS),DEQST2(NELS),ES 00001800
000318 000 DIMENSION RLR(4),DR(4),RD(4) 00001900
000319 000 C----- 00002000
000320 000 VOIDR = VOID(NEL) 00002100
000321 000 P = (SIGZ + SIGT + SIGR) / 3. 00002200
000322 000 TJ = ((SIGZ-SIGR)**2+(SIGR-SIGT)**2+(SIGT-SIGZ)**2)/6.+TAUZR**2 00002300
000323 000 PSQ = P*P 00002400
000324 000 ETS = 3.*TJ/PSQ 00002500
000325 000 POW = 1.-SMALLK/ALAMDA 00002600
000326 000 SIGMX(NEL) = P * ((XMS+ETS) /XMS) ** POW 00002700
000327 000 POP = SIGMX(NEL) 00002800
000328 000 AA = XMS * (2.*P-POP) / 3. 00002900
000329 000 BB = 3. * AA * XMS * P * POP * (1.+VOIDR) / BETA 00003000
000330 000 DO 100 I = 1,4 00003100
000331 000 DO 100 J = 1,4 00003200
000332 000 100 D(I,J) = DE(NEL,I,J) 00003300
000333 000 SZZ = 2.*SIGZ-SIGR-SIGT 00003400
000334 000 SRR = 2.*SIGR-SIGZ-SIGT 00003500
000335 000 STT = 2.*SIGT-SIGZ-SIGR 00003600
000336 000 SZR = 6.*TAUZR 00003700
000337 000 RLB(1) = SZZ + AA 00003800
000338 000 RLB(2) = SRR + AA 00003900
000339 000 RLB(3) = STT + AA 00004000
000340 000 RLB(4) = SZR 00004100

```

```

000341      000      CELP = (JSTRS(NEL,1)+DSTRS(NEL,2)+DSTRS(NEL,3)) / 3.      00004200
000342      000      DB(1) = SZZ      00004300
000343      000      DB(2) = SRR      00004400
000344      000      DB(3) = STT      00004500
000345      000      DB(4) = SZR      00004600
000346      000      DF1 = 0.      00004700
000347      000      DFK = 0.      00004800
000348      000      DO 200 I = 1,4      00004900
000349      000      DF1 = DF1 + RLB(I) * DSTRS(NEL,I)      00005000
000350      000      200 DFK = DFK + DB(I) * DSTRS(NEL,I)      00005100
000351      000      DFJ = DELP * POP * XMS      00005200
000352      000      POW = -SMALLK / ALAMDA      00005300
000353      000      DEN = (1.+ETS/XMS) ** POW      00005400
000354      000      DFK = DEN * (DFK-6.*TJ/P*DELP)      00005500
000355      000      DF = DFI - DFJ - DFK * (1.-SMALLK/ALAMDA)      00005600
000356      000      ASQ = XMS * P * (POP-P)      00005700
000357      000      TTJ = TJ * 3.      00005800
000358      000      SIGBA(NEL) = ASQ      00005900
000359      000      EP(NEL) = DF      00006000
000360      000      WRITE(6,620) NEL,VOIDR,TTJ,ASQ,POP,DF,XMS,ETS      00006100
000361      000      620 FORMAT(15,' VOIDR=',F10.4,' 3J=',F10.4,' ASQ=',F10.4,' PO=',      00006200
000362      000      'F10.4,' DF=',E12.5,' XMS=',E12.5,' ETS=',E12.5)      00006300
000363      000      C      00006400
000364      000      IF(DF.LT.0.) GO TO 764      00006500
000365      000      C      00006600
000366      000      DO 110 I = 1,4      00006700
000367      000      DR(I) = 0.      00006800
000368      000      RD(I) = 0.      00006900
000369      000      DO 110 J = 1,4      00007000
000370      000      DR(I) = DR(I) + D(1,J) * RLR(J)      00007100
000371      000      110 RD(I) = RD(I) + RLB(J) * D(J,I)      00007200
000372      000      DEN = 0.      00007300
000373      000      DO 120 I = 1,4      00007400
000374      000      120 DEN = DEN + RLB(I) * DB(I)      00007500
000375      000      DEN = DEN + BR      00007600
000376      000      DO 130 I = 1,4      00007700
000377      000      DO 130 J = 1,4      00007800
000378      000      130 D(1,J) = D(I,J) - DR(I) * RD(J) / DEN      00007900
000379      000      764 DO 111 I = 1,4      00008000
000380      000      DO 111 J = 1,4      00008100
000381      000      111 DMAT(NEL,I,J) = D(I,J)      00008200
000382      000      600 FORMAT(4E20.7)      00008300
000383      000      RETURN      00008400
000384      000      END      00008500
000385      000      WELT,SIH NASA*TPF$.STIFF1,,113762121110
000386      000      SUBROUTINE STIFF1(NTAPE)      00000100
000387      000      C-----      00000200
000388      000      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600      00000300
000389      000      C-----      00000400
000390      000      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON      00000500
000391      000      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),      00000600
000392      000      * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),YPEC(4,4),TYPEE(4,4),      00000700
000393      000      * TYPEF(4,4),YPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL      00000800
000394      000      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),      00000900
000395      000      1 R(NODS),Z(NODS),TUTDIS(MAX)      00001000
000396      000      COMMON /BLK7/ DSTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS),      00001100
000397      000      1 AOJ(NELS)      00001200

```

| | | | |
|--------|-----|-------------------------------------|----------|
| 000398 | 000 | COMMON /BLKA/M,N | 00001300 |
| 000399 | 000 | ----- | 00001400 |
| 000400 | 000 | C | 00001500 |
| 000401 | 000 | C | 00001600 |
| 000402 | 000 | C | 00001700 |
| 000403 | 000 | C | 00001800 |
| 000404 | 000 | | 00001900 |
| 000405 | 000 | AZ(1) = Z(LC) - Z(JC) | 00002000 |
| 000406 | 000 | AZ(2) = Z(IC) - Z(KC) | 00002100 |
| 000407 | 000 | AZ(3) = -AZ(1) | 00002200 |
| 000408 | 000 | AZ(4) = -AZ(2) | 00002300 |
| 000409 | 000 | BZ(1) = Z(KC) - Z(LC) | 00002400 |
| 000410 | 000 | BZ(2) = -BZ(1) | 00002500 |
| 000411 | 000 | BZ(3) = Z(JC) - Z(IC) | 00002600 |
| 000412 | 000 | BZ(4) = -BZ(3) | 00002700 |
| 000413 | 000 | CZ(1) = Z(KC) - Z(JC) | 00002800 |
| 000414 | 000 | CZ(2) = Z(IC) - Z(LC) | 00002900 |
| 000415 | 000 | CZ(3) = -CZ(2) | 00003000 |
| 000416 | 000 | CZ(4) = -CZ(1) | 00003100 |
| 000417 | 000 | AR(1) = R(JC) - R(LC) | 00003200 |
| 000418 | 000 | AR(2) = R(KC) - R(IC) | 00003300 |
| 000419 | 000 | AR(3) = -AR(1) | 00003400 |
| 000420 | 000 | AR(4) = -AR(2) | 00003500 |
| 000421 | 000 | BR(1) = R(LC) - R(KC) | 00003600 |
| 000422 | 000 | BR(2) = -BR(1) | 00003700 |
| 000423 | 000 | BR(3) = R(IC) - R(JC) | 00003800 |
| 000424 | 000 | BR(4) = -BR(3) | 00003900 |
| 000425 | 000 | CR(1) = R(JC) - R(KC) | 00004000 |
| 000426 | 000 | CR(2) = R(LC) - R(IC) | 00004100 |
| 000427 | 000 | CR(3) = -CR(2) | 00004200 |
| 000428 | 000 | CR(4) = -CR(1) | 00004300 |
| 000429 | 000 | AO = -AR(3)*AZ(2) + AR(4)*AZ(1) | 00004400 |
| 000430 | 000 | BO = -BR(2)*BZ(4) + BR(3)*BZ(1) | 00004500 |
| 000431 | 000 | CO = CR(3)*CZ(1) - CR(4)*CZ(2) | 00004600 |
| 000432 | 000 | C | 00004700 |
| 000433 | 000 | RT = R(IC) + R(JC) + R(KC) + R(LC) | 00004800 |
| 000434 | 000 | RA = -R(KC) + R(IC) - R(LC) + R(JC) | 00004900 |
| 000435 | 000 | RB = R(JC) - R(IC) + R(KC) - R(LC) | 00005000 |
| 000436 | 000 | RC = -R(IC) + R(JC) - R(KC) + R(LC) | 00005100 |
| 000437 | 000 | AOJ(NEL) = AO | 00005200 |
| 000438 | 000 | RTT(NEL) = RT | 00005300 |
| 000439 | 000 | C | 00005400 |
| 000440 | 000 | DO 200 M = 1,4 | 00005500 |
| 000441 | 000 | ARM(NEL,M) = AR(M) | 00005600 |
| 000442 | 000 | AZM(NEL,M) = AZ(M) | 00005700 |
| 000443 | 000 | DO 200 N = 1,4 | 00005800 |
| 000444 | 000 | CALL GAUSS(1,AA) | 00005900 |
| 000445 | 000 | CALL GAUSS(2,BB) | 00006000 |
| 000446 | 000 | CALL GAUSS(3,CC) | 00006100 |
| 000447 | 000 | CALL GAUSS(4,EE) | 00006200 |
| 000448 | 000 | CALL GAUSS(5,FF) | 00006300 |
| 000449 | 000 | CALL GAUSS(6,GG) | 00006400 |
| 000450 | 000 | TYPEA(M,N) = AA | 00006500 |
| 000451 | 000 | TYPEB(M,N) = BB | 00006600 |
| 000452 | 000 | TYPEC(M,N) = CC | 00006700 |
| 000453 | 000 | TYPEE(M,N) = EE | 00006800 |
| 000454 | 000 | TYPEF(M,N) = FF | 00006900 |
| | | TYPEG(M,N) = GG | |

```

000455 000 200 CONTINUE 00007000
000456 000 WRITE (NTAPE) TYPEA,TYPEB,TYPEC, TYPEE,TYPEF,TYPEG 00007100
000457 000 RETURN 00007200
000458 000 END 00007300
000459 000 WELT,SIH NASA*TPFS,STIFF2,,,113771121110
000460 000 SUBROUTINE STIFF2(NI,NTAPE) 00000100
000461 000 -----00000200
000462 000 PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 00000300
000463 000 -----00000400
000464 000 COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON 00000500
000465 000 COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4), 00000600
000466 000 * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4), 00000700
000467 000 * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RE,RA,RC,IC,JC,KC,LC,NEL 00000800
000468 000 COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8), 00000900
000469 000 1 R(NODS),Z(NODS),TOTOIS(MAX) 00001000
000470 000 -----00001100
000471 000 C 00001200
000472 000 C FORM STIFFNESS MATRIX. 00001300
000473 000 C 00001400
000474 000 IF (NI.NE.0) 00001500
000475 000 1 READ (NTAPE) TYPEA,TYPEB,TYPEC, TYPEE,TYPEF,TYPEG 00001600
000476 000 DO 200 I = 1,4 00001700
000477 000 DO 200 J = 1,4 00001800
000478 000 C 00001900
000479 000 STIFF(I,J) = TYPEA(I,J)*D(1,1)/8.+TYPEB(I,J)*D(4,1)/ 8.+ 00002000
000480 000 1 TYPEC(J,I)*D(1,4)/ 8.+TYPEE(I,J)*D(4,4)/ 8. 00002100
000481 000 STIFF(J+4,I) = TYPEC(J,I)*D(2,1)/8.+TYPEC(J,I)*D(3,1)+TYPEA(J,I)* 00002200
000482 000 1 D(4,1)/ 8.+TYPEE(J,I)*D(2,4)/ 8.+TYPEF(J,I)*D(3,4)*2+TYPEB(I,J)* 00002300
000483 000 2 D(4,4)/ 8. 00002400
000484 000 STIFF(I,J+4) = STIFF(J+4,I) 00002500
000485 000 STIFF(I+4,J+4) = TYPEE(I,J)*D(2,2)/8.+(TYPEB(I,J)+TYPEB(J,I))* 00002600
000486 000 1 D(2,4)/ 8.+TYPEA(I,J)*D(4,4)/ 8.+2.*(TYPEF(I,J)+TYPEF(J,I))* 00002700
000487 000 2 D(2,3)+(TYPEC(I,J)+TYPEC(J,I))*D(3,4) +TYPEG(I,J)*D(3,3) 00002800
000488 000 200 CONTINUE 00002900
000489 000 RETURN 00003000
000490 000 END 00003100
000491 000 WELT,SIH NASA*TPFS,FRICTN,,,132675133010
000492 000 SUBROUTINE FRICTN(IC,KC,NEL,NFT,ISHEAR,VOIDI) 00000100
000493 000 PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 00000200
000494 000 COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2 00000300
000495 000 COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP 00000400
000496 000 COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS, 00000500
000497 000 1 F1NC,FN,ULOAD,FEE,PMAX,DLMAX 00000600
000498 000 COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8), 00000700
000499 000 1 R(NODS),Z(NODS),TOTOIS(MAX) 00000800
000500 000 COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRI6D 00000900
000501 000 COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20), 00001000
000502 000 1 POR(20),DELTS,EC,XNUC,DELD 00001100
000503 000 DIMENSION TS(8,8) 00001200
000504 000 DZ = DZZ(NFT) 00001300
000505 000 DR = DRR(NFT) 00001400
000506 000 PPI = 3.14159 00001500
000507 000 RB = (R(IC)+R(KC)) / 2. 00001600
000508 000 PIR = PPI * RB / 3. 00001700
000509 000 BASE = R(KC) - R(IC) 00001800
000510 000 HIGH = Z(IC) - Z(KC) 00001900
000511 000 XL = (BASE*BASE + HIGH*HIGH)**.5 00002000

```


| | | | |
|--------|-----|--|----------|
| 000512 | 000 | XXL(NFT) = XL | 00002100 |
| 000513 | 000 | SINT = BASE / XL | 00002200 |
| 000514 | 000 | COST = HIGH / XL | 00002300 |
| 000515 | 000 | WRITE(6,600) NEL,XL,SINT,COST | 00002400 |
| 000516 | 000 | IF(ISHEAR.NE.0) GO TO 160 | 00002500 |
| 000517 | 000 | 600 FORMAT(' NEL,XL,SINT,COST',I5,3F12.4) | 00002600 |
| 000518 | 000 | DO 120 I = 1,8 | 00002700 |
| 000519 | 000 | 120 TR(NFT,I,I) = COST | 00002800 |
| 000520 | 000 | DO 130 I = 1,4 | 00002900 |
| 000521 | 000 | J = I + 4 | 00003000 |
| 000522 | 000 | TR(NFT,I,J) = -SINT | 00003100 |
| 000523 | 000 | 130 TR(NFT,J,I) = SINT | 00003200 |
| 000524 | 000 | DII = DZ | 00003300 |
| 000525 | 000 | DIJ = 0. | 00003400 |
| 000526 | 000 | DJJ = DR | 00003500 |
| 000527 | 000 | GO TO 170 | 00003600 |
| 000528 | 000 | 160 CONTINUE | 00003700 |
| 000529 | 000 | NAMELIST/NAME1/ SIGZZ,SIGRR,TAURZ,SIGN,SHEAR,TAUF,DII,DZ | 00003800 |
| 000530 | 000 | II = IO(NFT,1) | 00003900 |
| 000531 | 000 | JJ = IO(NFT,2) | 00004000 |
| 000532 | 000 | SIGZZ = (STRS(II,1)+STRS(JJ,1)) / 2. | 00004100 |
| 000533 | 000 | SIGRR = (STRS(II,2)+STRS(JJ,2)) / 2. | 00004200 |
| 000534 | 000 | TAURZ = (STRS(II,4)+STRS(JJ,4)) / 2. | 00004300 |
| 000535 | 000 | SIGN = SIGZZ*COST*COST+SIGRR*SINT*SINT-TAURZ*COST*SINT | 00004400 |
| 000536 | 000 | SHEAR = ABS(STRS(NEL,1)) | 00004500 |
| 000537 | 000 | TAUF = CK + SIGN*DELTS | 00004600 |
| 000538 | 000 | DII = DZ * (1.-SHEAR/TAUF) | 00004700 |
| 000539 | 000 | DZZ(NFT) = DII | 00004800 |
| 000540 | 000 | DIJ = 0. | 00004900 |
| 000541 | 000 | WRITE(6,NAME1) | 00005000 |
| 000542 | 000 | 170 CONTINUE | 00005100 |
| 000543 | 000 | DIIF = DII * PIR / XXL(NFT) | 00005200 |
| 000544 | 000 | DIJF = DIJ * PIR / XXL(NFT) | 00005300 |
| 000545 | 000 | DJJF = DJJ * PIR / XXL(NFT) | 00005400 |
| 000546 | 000 | DO 190 I = 1,4 | 00005500 |
| 000547 | 000 | STIFF(I,1) = 2.*DIIF | 00005600 |
| 000548 | 000 | STIFF(I+4,I+4) = 2.*DJJF | 00005700 |
| 000549 | 000 | STIFF(I,1+4) = 2.*DIJF | 00005800 |
| 000550 | 000 | 190 STIFF(I,9-1) = DIJF | 00005900 |
| 000551 | 000 | DO 191 I = 1,2 | 00006000 |
| 000552 | 000 | STIFF(I,1+2) = -DIIF | 00006100 |
| 000553 | 000 | STIFF(I,5-1) = DIIF | 00006200 |
| 000554 | 000 | STIFF(1,7-1) = -2.*DIJF | 00006300 |
| 000555 | 000 | STIFF(I,1+6) = -DIJF | 00006400 |
| 000556 | 000 | STIFF(I+4,1+6) = -DJJF | 00006500 |
| 000557 | 000 | STIFF(I+4,9-1) = DJJF | 00006600 |
| 000558 | 000 | STIFF(I+2,1+4) = -DIJF | 00006700 |
| 000559 | 000 | STIFF(I+2,9-1) = -2.*DIJF | 00006800 |
| 000560 | 000 | STIFF(2+1-1,2+1) = -2.*DIIF | 00006900 |
| 000561 | 000 | 191 STIFF(2+1+3,2+1+4) = -2.*DJJF | 00007000 |
| 000562 | 000 | DO 210 I = 1,8 | 00007100 |
| 000563 | 000 | DO 210 J = 1,8 | 00007200 |
| 000564 | 000 | 210 STIFF(J,I) = STIFF(I,J) | 00007300 |
| 000565 | 000 | DMAT(NEL,1,1) = DII | 00007400 |
| 000566 | 000 | DMAT(NEL,1,2) = DIJ | 00007500 |
| 000567 | 000 | DMAT(NEL,2,2) = DJJ | 00007600 |
| 000568 | 000 | WRITE(6,630) NEL,DZ,DR,DII,DIJ,DJJ,XXL(NFT) | 00007700 |

| | | | | |
|--------|-----|-----|--|----------|
| 000569 | 000 | 630 | FORMAT(' NEL DZ DR DII DIJ DJJ LENGTH',IS,6E12.5) | 00007800 |
| 000570 | 000 | | WRITE(6,610) STIFF | 00007900 |
| 000571 | 000 | 610 | FORMAT(8E15,6) | 00008000 |
| 000572 | 000 | | DO 140 I = 1,8 | 00008100 |
| 000573 | 000 | | DO 140 J = 1,8 | 00008200 |
| 000574 | 000 | | SUM = 0. | 00008300 |
| 000575 | 000 | | DO 140 K = 1,8 | 00008400 |
| 000576 | 000 | 144 | SUM = SUM + TR(NFT,K,I) * STIFF(K,J) | 00008500 |
| 000577 | 000 | 140 | TS(I,J) = SUM | 00008600 |
| 000578 | 000 | | DO 150 I = 1,8 | 00008700 |
| 000579 | 000 | | DO 150 J = 1,8 | 00008800 |
| 000580 | 000 | | SUM = 0. | 00008900 |
| 000581 | 000 | | DO 155 K = 1,8 | 00009000 |
| 000582 | 000 | 155 | SUM = SUM + TS(I,K) * TR(NFT,K,J) | 00009100 |
| 000583 | 000 | 150 | STIFF(I,J) = SUM | 00009200 |
| 000584 | 000 | | DO 200 I = 1,8 | 00009300 |
| 000585 | 000 | | DO 200 J = 1,8 | 00009400 |
| 000586 | 000 | 200 | FRCK(NFT,I,J) = STIFF(I,J) | 00009500 |
| 000587 | 000 | | WRITE(6,610) STIFF | 00009600 |
| 000588 | 000 | | RETURN | 00009700 |
| 000589 | 000 | | END | 00009800 |
| 000590 | 000 | | WELT,SIH NASA*TPFS.ASSEMB,,,114006121110 | |
| 000591 | 000 | | SUBROUTINE ASSEMB(NEL,NFT) | 00000100 |
| 000592 | 000 | C | ----- | 00000200 |
| 000593 | 000 | | PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 | 00000300 |
| 000594 | 000 | C | ----- | 00000400 |
| 000595 | 000 | | COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON | 00000500 |
| 000596 | 000 | | COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2 | 00000600 |
| 000597 | 000 | | COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST | 00000700 |
| 000598 | 000 | | COMMON /BLK6/ SIGH,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8), | 00000800 |
| 000599 | 000 | | 1 R(NODS),Z(NODS),TOTDIS(MAX) | 00000900 |
| 000600 | 000 | | COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20), | 00001000 |
| 000601 | 000 | | 1 POR(20),DELTS,EC,XNUC,DELD | 00001100 |
| 000602 | 000 | C | ----- | 00001200 |
| 000603 | 000 | C | | 00001300 |
| 000604 | 000 | | NFREE = INODE * 2 | 00001400 |
| 000605 | 000 | | IF(NFT.LT.1.OR.NF1.GT.NBC) GO TO 120 | 00001500 |
| 000606 | 000 | | DO 130 I = 1,8 | 00001600 |
| 000607 | 000 | | DO 130 J = 1,8 | 00001700 |
| 000608 | 000 | 130 | STIFF(I,J) = FRCK(NFT,I,J) | 00001800 |
| 000609 | 000 | 120 | CONTINUE | 00001900 |
| 000610 | 000 | C | | 00002000 |
| 000611 | 000 | | DO 110 I = 1,8 | 00002100 |
| 000612 | 000 | | II = KK(NEL,I) | 00002200 |
| 000613 | 000 | | DO 110 J = 1,8 | 00002300 |
| 000614 | 000 | | JJ = KK(NEL,J) | 00002400 |
| 000615 | 000 | | IF(I1.EQ.0.OR.JJ.EQ.0) GO TO 110 | 00002500 |
| 000616 | 000 | | IF(II.LT.JJ) GO TO 110 | 00002600 |
| 000617 | 000 | | IF(II.GT.IHBI) GO TO 104 | 00002700 |
| 000618 | 000 | | L = JJ + (II-1) * II / 2 | 00002800 |
| 000619 | 000 | | GO TO 105 | 00002900 |
| 000620 | 000 | 104 | L = JJ + LT + (II-IHB) * IHBI | 00003000 |
| 000621 | 000 | 105 | XK(L) = -XK(L) + STIFF(I,J) | 00003100 |
| 000622 | 000 | 110 | CONTINUE | 00003200 |
| 000623 | 000 | | RETURN | 00003300 |
| 000624 | 000 | | END | 00003400 |
| 000625 | 000 | | WELT,SIH NASA*TPFS.DISPL,,,11401121110 | |

| | | | |
|--------|-----|---|----------|
| 000626 | 000 | SUBROUTINE DISPL(NFREE,NI,INODE,INCR) | 00000100 |
| 000627 | 000 | C----- | 00000200 |
| 000628 | 000 | PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 | 00000300 |
| 000629 | 000 | C----- | 00000400 |
| 000630 | 000 | COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2 | 00000500 |
| 000631 | 000 | COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST | 00000600 |
| 000632 | 000 | COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8), | 00000700 |
| 000633 | 000 | 1 R(NODS),Z(NODS),TUTDIS(MAX) | 00000800 |
| 000634 | 000 | C----- | 00000900 |
| 000635 | 000 | C | 00001000 |
| 000636 | 000 | N = NFREE | 00001100 |
| 000637 | 000 | ZTEST = 0.000001 | 00001200 |
| 000638 | 000 | C | 00001300 |
| 000639 | 000 | DO 100 J = 1,NFREE | 00001400 |
| 000640 | 000 | IF(J.GT.IHBI) GO TO 108 | 00001500 |
| 000641 | 000 | L = (J+1) * J / 2 | 00001600 |
| 000642 | 000 | GO TO 109 | 00001700 |
| 000643 | 000 | 108 L = LT + IHB * (J - IHBI) | 00001800 |
| 000644 | 000 | 109 XTEST = ABS(XK(L)) | 00001900 |
| 000645 | 000 | IF(XTEST.LT.ZTEST) XK(L) = 1. | 00002000 |
| 000646 | 000 | 100 CONTINUE | 00002100 |
| 000647 | 000 | C | 00002200 |
| 000648 | 000 | DO 110 I = 1,NFREE | 00002300 |
| 000649 | 000 | 110 XK(LAST+I) = APF(I) | 00002400 |
| 000650 | 000 | C | 00002500 |
| 000651 | 000 | C | 00002600 |
| 000652 | 000 | CALL FACTOR(NFREE) | 00002700 |
| 000653 | 000 | C | 00002800 |
| 000654 | 000 | C | 00002900 |
| 000655 | 000 | CALL SOLTN(NFREE) | 00003000 |
| 000656 | 000 | C | 00003100 |
| 000657 | 000 | C | 00003200 |
| 000658 | 000 | WRITE(6,600) INCR,NI | 00003300 |
| 000659 | 000 | NN = 10 | 00003400 |
| 000660 | 000 | DO 280 J = 1,NN | 00003500 |
| 000661 | 000 | II = (J-1) * 2 + 1 + LAST | 00003600 |
| 000662 | 000 | JJ = II + 1 | 00003700 |
| 000663 | 000 | 280 WRITE(6,610) J,XK(II),XK(JJ) | 00003800 |
| 000664 | 000 | 600 FORMAT(/// 20X,'DISPLACEMENTS FOR CYCLE NO.',I4//18X,'Z - DISPL', | 00003900 |
| 000665 | 000 | * 15X,'R - DISPL',15X,'LOAD INCREMENT STEP =',I5//) | 00004000 |
| 000666 | 000 | 610 FORMAT(I7,2E20.7) | 00004100 |
| 000667 | 000 | RETURN | 00004200 |
| 000668 | 000 | END | 00004300 |
| 000669 | 000 | WELT,SIH NASA*TPFS.FACTOR,,,114013121110 | |
| 000670 | 000 | SUBROUTINE FACTOR(NFREE) | 00000100 |
| 000671 | 000 | C THIS SUBROUTINE PERFORMS FACTORING | 00000200 |
| 000672 | 000 | PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 | 00000300 |
| 000673 | 000 | COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST | 00000400 |
| 000674 | 000 | N = NFREE | 00000500 |
| 000675 | 000 | IHB1 = IHB | 00000600 |
| 000676 | 000 | DO 8 J=1,N | 00000700 |
| 000677 | 000 | IF(I.GT.IHB1) GO TO 2 | 00000800 |
| 000678 | 000 | K=1 | 00000900 |
| 000679 | 000 | M=K+(I-1)*1/2 | 00001000 |
| 000680 | 000 | GO TO 3 | 00001100 |
| 000681 | 000 | 2 K=I-IHB1 | 00001200 |
| 000682 | 000 | M=K+LT+(I-IHB)*IHB1 | 00001300 |

| | | | |
|--------|-----|--|----------|
| 000683 | 000 | 3 J=I+IHB1 | 00001400 |
| 000684 | 000 | IF(J.GT.N) GO TO 4 | 00001500 |
| 000685 | 000 | JJ=I+IHB1 | 00001600 |
| 000686 | 000 | GO TO 5 | 00001700 |
| 000687 | 000 | 4 JJ=N | 00001800 |
| 000688 | 000 | 5 B=0.0 | 00001900 |
| 000689 | 000 | LA=I-1 | 00002000 |
| 000690 | 000 | LB=I+1 | 00002100 |
| 000691 | 000 | IF(LA.EQ.0) GO TO 6 | 00002200 |
| 000692 | 000 | DO 7 L=K,LA | 00002300 |
| 000693 | 000 | IF(L.GT.IHB1)GO TO 50 | 00002400 |
| 000694 | 000 | J = (L+1)*L/2 | 00002500 |
| 000695 | 000 | GO TO 51 | 00002600 |
| 000696 | 000 | 50 J = LT + IHB*(L-IHB1) | 00002700 |
| 000697 | 000 | 51 A = XK(M) | 00002800 |
| 000698 | 000 | R = B+A*A*XK(J) | 00002900 |
| 000699 | 000 | 7 M=M+1 | 00003000 |
| 000700 | 000 | 6 A=XK(M) | 00003100 |
| 000701 | 000 | XK(M)=A-B | 00003200 |
| 000702 | 000 | IF(I.EQ.N) GO TO 8 | 00003300 |
| 000703 | 000 | DO 9 J=LB,JJ | 00003400 |
| 000704 | 000 | SUM=0.0 | 00003500 |
| 000705 | 000 | IF(J.GT.IHB1) GO TO 10 | 00003600 |
| 000706 | 000 | K=1 | 00003700 |
| 000707 | 000 | MM=K+(J-1)*J/2 | 00003800 |
| 000708 | 000 | GO TO 11 | 00003900 |
| 000709 | 000 | 10 K=J-IHB1 | 00004000 |
| 000710 | 000 | MM=K+LT+(J-IHB)*IHB1 | 00004100 |
| 000711 | 000 | 11 IF(LA.EQ.0) GO TO 9 | 00004200 |
| 000712 | 000 | IF(K.GT.LA) GO TO 9 | 00004300 |
| 000713 | 000 | DO 12 JA=K,LA | 00004400 |
| 000714 | 000 | L=M-1+JA | 00004500 |
| 000715 | 000 | IF(JA.GT.IHB1) GO TO 13 | 00004600 |
| 000716 | 000 | L1=(JA+1)*JA/2 | 00004700 |
| 000717 | 000 | GO TO 14 | 00004800 |
| 000718 | 000 | 13 L1=LT+IHB*(JA-IHB1) | 00004900 |
| 000719 | 000 | 14 SUM=SUM+XK(MM)*XK(L)*XK(L1) | 00005000 |
| 000720 | 000 | 12 MM=MM+1 | 00005100 |
| 000721 | 000 | 9 XK(MM)=(XK(MM)-SUM)/XK(M) | 00005200 |
| 000722 | 000 | 8 CONTINUE | 00005300 |
| 000723 | 000 | RETURN | 00005400 |
| 000724 | 000 | END | 00005500 |
| 000725 | 000 | WELT,SIH NASA*TPFS,SOLTN,,,114016121110 | |
| 000726 | 000 | SUBROUTINE SOLTN(NFREE): | 00000100 |
| 000727 | 000 | PARAMETER NOUS=300,NELS=260,NT=20000,MAX=600 | 00000200 |
| 000728 | 000 | COMMON /BLK3/ XK(N1),APF(MAX),IMAX,IHB,IHB1,LT,LAST | 00000300 |
| 000729 | 000 | C THIS PORTION OF SUBROUTINE PERFORMS FORWARD-SUBSTITUTION | 00000400 |
| 000730 | 000 | C | 00000500 |
| 000731 | 000 | N = NFREE | 00000600 |
| 000732 | 000 | IHB1 = IHB | 00000700 |
| 000733 | 000 | NF = LAST + 1 | 00000800 |
| 000734 | 000 | C | 00000900 |
| 000735 | 000 | 14 DO 1 K = 2,N | 00001000 |
| 000736 | 000 | C | 00001100 |
| 000737 | 000 | IF(K.GT.IHB1) GO TO 2 | 00001200 |
| 000738 | 000 | M=0 | 00001300 |
| 000739 | 000 | MM=K-1 | 00001400 |

| | | | |
|--------|-----|---|----------|
| 000740 | 000 | M1=MM*K/2 | 00001500 |
| 000741 | 000 | GO TO 3 | 00001600 |
| 000742 | 000 | 2 N=K-IHB | 00001700 |
| 000743 | 000 | MM=IHB1 | 00001800 |
| 000744 | 000 | M1=M*IHB1+LT | 00001900 |
| 000745 | 000 | 3 SUM=0.0 | 00002000 |
| 000746 | 000 | DO 4 L=1,MM | 00002100 |
| 000747 | 000 | LL=L+M | 00002200 |
| 000748 | 000 | JJ=LL+M1 | 00002300 |
| 000749 | 000 | LL=LL+NF-1 | 00002400 |
| 000750 | 000 | 4 SUM=SUM+XK(JJ)*XK(LL) | 00002500 |
| 000751 | 000 | 1 XK(LL+1)=XK(LL+1)-SUM | 00002600 |
| 000752 | 000 | J = NF+N-1 | 00002700 |
| 000753 | 000 | C THIS PORTION OF SUBROUTINE PERFORMS BACK-SUBSTITUTION | 00002800 |
| 000754 | 000 | NF=NF+N-1 | 00002900 |
| 000755 | 000 | XK(NF)=XK(NF)/XK(LAST) | 00003000 |
| 000756 | 000 | DO 5 K=2,N | 00003100 |
| 000757 | 000 | L=N-K+1 | 00003200 |
| 000758 | 000 | IF(L.GT.IHB1) GO TO 6 | 00003300 |
| 000759 | 000 | I=L+(L-1)*L/2 | 00003400 |
| 000760 | 000 | GO TO 7 | 00003500 |
| 000761 | 000 | 6 I=L+(L-IHB)*IHB1+LT | 00003600 |
| 000762 | 000 | 7 IR=N-IHB | 00003700 |
| 000763 | 000 | IF(L.GT.IR) GO TO 8 | 00003800 |
| 000764 | 000 | J=IHB1 | 00003900 |
| 000765 | 000 | GO TO 9 | 00004000 |
| 000766 | 000 | 8 J=K-1 | 00004100 |
| 000767 | 000 | 9 SUM=0.0 | 00004200 |
| 000768 | 000 | DO 10 M=1,J | 00004300 |
| 000769 | 000 | MM=L+M | 00004400 |
| 000770 | 000 | IF(MM.GT.IHB1) GO TO 11 | 00004500 |
| 000771 | 000 | NN=L+(MM-1)*MM/2 | 00004600 |
| 000772 | 000 | GO TO 12 | 00004700 |
| 000773 | 000 | 11 NN=L+(MM-IHB)*IHB1+LT | 00004800 |
| 000774 | 000 | 12 MM=NF-N+MM | 00004900 |
| 000775 | 000 | 10 SUM=SUM+XK(NN)*XK(MM) | 00005000 |
| 000776 | 000 | MM=NF-N+L | 00005100 |
| 000777 | 000 | 5 XK(MM)=XK(MM)/XK(1)-SUM | 00005200 |
| 000778 | 000 | RETURN | 00005300 |
| 000779 | 000 | END | 00005400 |
| 000780 | 000 | WELT,SIH NASA*TPFS,STRAIN,,,132705133010 | |
| 000781 | 000 | SUBROUTINE STRAIN(NI,INCR) | 00000100 |
| 000782 | 000 | C----- | 00000200 |
| 000783 | 000 | PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 | 00000300 |
| 000784 | 000 | C----- | 00000400 |
| 000785 | 000 | COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSON | 00000500 |
| 000786 | 000 | COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2 | 00000600 |
| 000787 | 000 | COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHR,IHB1,LT,LAST | 00000700 |
| 000788 | 000 | COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELMN1(MAX),POP | 00000800 |
| 000789 | 000 | COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS, | 00000900 |
| 000790 | 000 | 1 FINC,FN,ULOAD,FEL,PMAX,DLMAX | 00001000 |
| 000791 | 000 | COMMON /BLK7/ DSTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS), | 00001100 |
| 000792 | 000 | 1 AQQ(NELS) | 00001200 |
| 000793 | 000 | COMMON /BLK8/ INCNN,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,XMS | 00001300 |
| 000794 | 000 | COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NEREE,NELST,ICASE,NRIGN | 00001400 |
| 000795 | 000 | COMMON /BLK11/ VOID(NELS),UGAM(NELS,4) | 00001500 |
| 000796 | 000 | COMMON /BLK12/ SIGMX(NELS),DEP(NELS),EP(NELS),DEGST2(NELS),ES | 00001600 |

```

000797      000      -DIMENSION      U(4),V(4),EPS(4),SIG(4)      00001700
000798      000      C      00001800
000799      000      C      00001900
000800      000      SUMA = 0.      00002000
000801      000      SUMB = 0.      00002100
000802      000      IF(NBC.NE.0) CALL INTFAC(DEF,NI)      00002200
000803      000      WRITE(6,866) NI,INCR      00002300
000804      000      666 FORMAT(///10X,'STRAINS AND STRESSES FOR INCREMENT NO. =',I4,      00002400
000805      000      * ' NO. OF CYCLES = ',I4,///5X,'ELEM',T14,'SIG-Z',T30,'SIG-R',T45,      00002500
000806      000      1'SIG-T',T60,'TAU-ZR',T75,'AREA',T90,'VOID RATIO')      00002600
000807      000      DO 900 NEL = 1, NELEM      00002700
000808      000      NFI = NEL - NRIGD      00002800
000809      000      IF(NFI.GT.0.AND.NFI.LE.NBC) GO TO 900      00002900
000810      000      DO 100 I = 1,4      00003000
000811      000      IN = IJKL(NEL,I)      00003100
000812      000      II = (IN-1)*2 + LAST + 1      00003200
000813      000      JJ = II + 1      00003300
000814      000      V(I) = XK(II)      00003400
000815      000      100 U(I) = XK(JJ)      00003500
000816      000      C      FIND STRAINS AND STRESSES AT THE CENTROID.      00003600
000817      000      EZ = 0.      00003700
000818      000      ER = 0.      00003800
000819      000      GM = 0.      00003900
000820      000      SUM = 0.      00004000
000821      000      DO 111 I = 1,4      00004100
000822      000      EZ = EZ + ARM(NEL,I) * V(I)      00004200
000823      000      ER = ER + AZM(NEL,I) * U(I)      00004300
000824      000      GM = GM + ARM(NEL,I) * U(I) + AZM(NEL,I) * V(I)      00004400
000825      000      111 SUM = SUM + U(I)      00004500
000826      000      C      COMPRESSION POSITIVE.      00004600
000827      000      EPS(1) = -EZ / A0J(NEL)      00004700
000828      000      EPS(2) = -ER / A0J(NEL)      00004800
000829      000      EPS(3) = -SUM / RIT(NEL)      00004900
000830      000      EPS(4) = -GM / A0J(NEL)      00005000
000831      000      C      EPS(1) = STRAIN IN Z - DIRECTION.      00005100
000832      000      C      EPS(2) = STRAIN IN R - DIRECTION.      00005200
000833      000      C      EPS(3) = TANGENTIAL STRAIN.      00005300
000834      000      C      EPS(4) = SHEAR STRAIN      00005400
000835      000      DO 200 I = 1,4      00005500
000836      000      DGAM(NEL,I) = EPS(I)      00005600
000837      000      SIG(I) = 0.      00005700
000838      000      DO 200 J = 1,4      00005800
000839      000      SIG(I) = SIG(I) + DMAT(NEL,I,J) * EPS(J)      00005900
000840      000      200 CONTINUE      00006000
000841      000      I11=NEL      00006100
000842      000      DO 210 I = 1,4      00006200
000843      000      DSTRS(I11,I)=SIG(I)      00006300
000844      000      210 CONTINUE      00006400
000845      000      IF(NEL.LE.NRIGD) GO TO 890      00006500
000846      000      CALL AREA(IJKL(NEL,1),IJKL(NEL,2),IJKL(NEL,3),IJKL(NEL,4),AREA)      00006600
000847      000      RATE = AREA / DELNM1(NEL)      00006700
000848      000      VOID(NEL) = RATE * (1.+VOIDI) - 1.      00006800
000849      000      890 CONTINUE      00006900
000850      000      WRITE(6,600) NEL,SIG,AREA,VOID(NEL)      00007000
000851      000      900 CONTINUE      00007100
000852      000      C      00007200
000853      000      600 FORMAT(I9,7F15.6)      00007300

```

| | | | | |
|--------|-----|----------|--|----------|
| 000854 | 000 | 620 | FORMAT(I10,2F15.6) | 00007400 |
| 000855 | 000 | | RETURN | 00007500 |
| 000856 | 000 | | END | 00007600 |
| 000857 | 000 | WELT,SIH | NASA*TPFS,INTFAC,,,132711133010 | |
| 000858 | 000 | | SUBROUTINE INTFAC(UEI,NI) | 00000100 |
| 000859 | 000 | | PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 | 00000200 |
| 000860 | 000 | COMMON | /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,NCYCL,EPSON | 00000300 |
| 000861 | 000 | COMMON | /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2 | 00000400 |
| 000862 | 000 | COMMON | /BLK3/ XK(NF),APF(MAX),IMAX,IHR,IHBI,LT,LAST | 00000500 |
| 000863 | 000 | COMMON | /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP | 00000600 |
| 000864 | 000 | COMMON | /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS, | 00000700 |
| 000865 | 000 | 1 | FINC,FN,ULOAD,FEL,PMAX,DLMAX | 00000800 |
| 000866 | 000 | COMMON | /BLK7/ DSTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS), | 00000900 |
| 000867 | 000 | 1 | AOJ(NELS) | 00001000 |
| 000868 | 000 | COMMON | /BLK8/ INCR,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,PP | 00001100 |
| 000869 | 000 | COMMON | /BLK9/ PIS,SMALLK,CK,RETA,PO,NFREE,NELST,ICASE,NRIGD | 00001200 |
| 000870 | 000 | COMMON | /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRP(20), | 00001300 |
| 000871 | 000 | 1 | POR(20),DELTS,EC,XNUC,DELD | 00001400 |
| 000872 | 000 | | DIMENSION UG(8),UL(8) | 00001500 |
| 000873 | 000 | | DE1 = 0. | 00001600 |
| 000874 | 000 | | DO 900 NEL = 1,NBC | 00001700 |
| 000875 | 000 | | III = NEL + NRIGD | 00001800 |
| 000876 | 000 | | DO 100 I = 1,4 | 00001900 |
| 000877 | 000 | | II = IJKL(III,1) * 2 + LAST | 00002000 |
| 000878 | 000 | | UG(I) = XK(II-1) | 00002100 |
| 000879 | 000 | 100 | UG(I+4) = XK(II) | 00002200 |
| 000880 | 000 | | DO 110 I = 1,8 | 00002300 |
| 000881 | 000 | | UL(I) = 0. | 00002400 |
| 000882 | 000 | | DO 110 J = 1,8 | 00002500 |
| 000883 | 000 | 110 | UL(I) = UL(I) + TR(NEL,I,J) * UG(J) | 00002600 |
| 000884 | 000 | | WRITE(6,620) UG | 00002700 |
| 000885 | 000 | | WRITE(6,620) UL | 00002800 |
| 000886 | 000 | | EZ = (-UL(1)+UL(2)+UL(3)-UL(4)) / (2.*XXL(NEL)) | 00002900 |
| 000887 | 000 | | ER = (-UL(5)+UL(6)+UL(7)-UL(8)) / (2.*XXL(NEL)) | 00003000 |
| 000888 | 000 | | SIGZ = DMAT(III,1,1) * EZ + DMAT(III,1,2) * ER | 00003100 |
| 000889 | 000 | | SIGR = DMAT(III,1,2) * EZ + DMAT(III,2,2) * ER | 00003200 |
| 000890 | 000 | | SIGZ = -SIGZ | 00003300 |
| 000891 | 000 | | DSTRS(III,1) = SIGZ | 00003400 |
| 000892 | 000 | | DSTRS(III,2) = SIGR | 00003500 |
| 000893 | 000 | | NAMelist/NAME2/ III,SIGZ,SIGR | 00003600 |
| 000894 | 000 | | WRITE(6,NAME2) | 00003700 |
| 000895 | 000 | 900 | CONTINUE | 00003800 |
| 000896 | 000 | 620 | FORMAT(HF15.5) | 00003900 |
| 000897 | 000 | | RETURN | 00004000 |
| 000898 | 000 | | END | 00004100 |
| 000899 | 000 | WELT,SIH | NASA*TPFS.SETUP,,,114055121110 | |
| 000900 | 000 | | SUBROUTINE SETUP | 00000100 |
| 000901 | 000 | COMMON | /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4), | 00000200 |
| 000902 | 000 | * | HN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4), | 00000300 |
| 000903 | 000 | * | TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RE,RA,RC,IC,JC,KC,LC,NEL | 00000400 |
| 000904 | 000 | | W(1) = .1713244924 | 00000500 |
| 000905 | 000 | | W(2) = .3607615730 | 00000600 |
| 000906 | 000 | | W(3) = .4679139346 | 00000700 |
| 000907 | 000 | | W(4) = W(3) | 00000800 |
| 000908 | 000 | | W(5) = W(2) | 00000900 |
| 000909 | 000 | | W(6) = W(1) | 00001000 |
| 000910 | 000 | | H(1) = .9324695142 | 00001100 |

```

000911      000          H(2) = .6612093865      00001200
000912      000          H(3) = .2386191861      00001300
000913      000          H(4) = -H(3)           00001400
000914      000          H(5) = -H(2)           00001500
000915      000          H(6) = -H(1)           00001600
000916      000          BN(1) = -1.            00001700
000917      000          BN(2) = 1.             00001800
000918      000          BN(3) = 1.             00001900
000919      000          BN(4) = -1.            00002000
000920      000          CN(1) = 1.             00002100
000921      000          CN(2) = 1.             00002200
000922      000          CN(3) = -1.            00002300
000923      000          CN(4) = -1.            00002400
000924      000          DN(1) = -1.            00002500
000925      000          DN(2) = 1.             00002600
000926      000          DN(3) = -1.            00002700
000927      000          DN(4) = 1.             00002800
000928      000          RETURN                  00002900
000929      000          END                      00003000
000930      000      WELT,SIH NASA*TPF$.ELASTC,,,114060121110
000931      000      SUBROUTINE ELASTC(U,NRIGD,E,XNU)      00000100
000932      000      DIMENSION D(4,4)              00000200
000933      000      WRITE(6,600) NRIGD,E,XNU          00000300
000934      000      CONST = E*XNU / ((1.+XNU)*(1.-XNU*2.)) 00000400
000935      000      SHEAR = E / (2.*(1.+XNU))         00000500
000936      000      D(1,1) = CONST + SHEAR*2.        00000600
000937      000      D(2,2) = D(1,1)                 00000700
000938      000      D(3,3) = D(1,1)                 00000800
000939      000      D(4,4) = SHEAR                  00000900
000940      000      D(1,2) = CONST                  00001000
000941      000      D(1,3) = CONST                  00001100
000942      000      D(2,3) = CONST                  00001200
000943      000      DO 100 I = 1,4                  00001300
000944      000      DO 100 J = I,4                  00001400
000945      000      100 D(J,I) = D(I,J)              00001500
000946      000      600 FORMAT(/' FOR FIRST',I4,' ELEMENTS, THE FOLLOWING MATERIAL PRO 00001600
000947      000      * PERTIES ARE USED TO FORM ELASTIC MATRIX (D)'/) E =',F20.7,
000948      000      * ' XNU =',F10.3//)              00001700
000949      000      RETURN                            00001800
000950      000      END                              00001900
000951      000      WELT,SIH NASA*TPF$.GAUSS,,,114063121110
000952      000      SUBROUTINE GAUSS(I1,AA)          00000100
000953      000      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4), 00000200
000954      000      * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4), 00000300
000955      000      * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL 00000400
000956      000      TWOPI = 6.28318531              00000500
000957      000      IPT = 6                          00000600
000958      000      AA = 0.                          00000700
000959      000      DO 100 I = 1,IPT                 00000800
000960      000      X = H(I)                          00000900
000961      000      DO 100 J = 1,IPT                 00001000
000962      000      Y = H(J)                          00001100
000963      000      AA = AA + W(I) * W(J) * F(X,Y,I1) 00001200
000964      000      100 CONTINUE                      00001300
000965      000      AA = AA * TWOPI                    00001400
000966      000      RETURN                            00001500
000967      000      END                              00001600

```



```

000968 000 WELT,SIH NASA*TPF$.F,,,114066121110
000969 000 FUNCTION F(X,Y,IT) 00000100
000970 000 COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),A7(4),HZ(4),CZ(4), 00000200
000971 000 * RN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4), 00000300
000972 000 * TYPEF(4,4),TYPEG(4,4),AU,B0,CO,RT,RE,RA,RC,IC,JC,KC,LC,NFL 00000400
000973 000 COMMON /BLKA/M,N 00000500
000974 000 C X STANDS FOR ZHAI IN ZHAI - EITA COORD. 00000600
000975 000 C Y STANDS FOR EITA IN ZHAI - EITA COORD. 00000700
000976 000 C FB = DET. OF JACOBI. 00000800
000977 000 C FC = (N) * (R) 00000900
000978 000 FB = AO + B0 * X + CO * Y 00001000
000979 000 FC = ( RT + RB * X + RA * Y + RC * X * Y) / 4. 00001100
000980 000 GO TO (10,20,30,40,50,60),IT 00001200
000981 000 10 F = (AR(M)+BR(M)*A+CR(M)*Y) * (AR(N)+BR(N)*X+CR(N)*Y) /FB 00001300
000982 000 F = F * FC 00001400
000983 000 RETURN 00001500
000984 000 20 F = (AZ(M)+BZ(M)*X+CZ(M)*Y) * (AR(N)+BR(N)*X+CR(N)*Y) /FB 00001600
000985 000 F = F * FC 00001700
000986 000 RETURN 00001800
000987 000 30 F = (1.+BN(M)*X+CN(M)*Y+DN(M)*X*Y) * (AR(N)+BR(N)*X+CR(N)*Y) / 32.00001900
000988 000 RETURN 00002000
000989 000 40 F = (AZ(M)+BZ(M)*X+CZ(M)*Y) * (AZ(N)+BZ(N)*X+CZ(N)*Y) /FB 00002100
000990 000 F = F * FC 00002200
000991 000 RETURN 00002300
000992 000 50 F = (1.+BN(M)*X+CN(M)*Y+DN(M)*X*Y) * (AZ(N)+BZ(N)*X+CZ(N)*Y) / 64.00002400
000993 000 RETURN 00002500
000994 000 60 F = (1.+BN(M)*X+CN(M)*Y+DN(M)*X*Y) * 00002600
000995 000 1 (1.+RN(N)*X+CN(N)*Y+DN(N)*X*Y) * FB / (128.*FC) 00002700
000996 000 RETURN 00002800
000997 000 END 00002900
000998 000 WELT,SIH NASA*TPF$.AREA,,,114072121110
000999 000 SUBROUTINE AREA(IC,JC,KC,LC,AREA) 00000100
001000 000 PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 00000200
001001 000 COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8), 00000300
001002 000 1 R(NODS),Z(NODS),TUTDIS(MAX) 00000400
001003 000 AI = (R(JC)-R(IC)) * (Z(LC)-Z(IC)) - (R(LC)-R(IC)) * (Z(JC)-Z(IC)) 00000500
001004 000 AJ = (R(KC)-R(JC)) * (Z(LC)-Z(JC)) - (R(LC)-R(JC)) * (Z(KC)-Z(JC)) 00000600
001005 000 IF(AI.LT.0) AI = -AI 00000700
001006 000 IF(AJ.LT.0) AJ = -AJ 00000800
001007 000 AREA = (AI + AJ) / 2. 00000900
001008 000 RETURN 00001000
001009 000 END ? 00001100
001010 000 WELT,SIH NASA*TPF$.PTLOAD,,,114074121110
001011 000 SUBROUTINE PTLOAD(NAPC,ULOAD) 00000100
001012 000 PARAMETER NODS=300,NELS=260,NF=20000,MAX=600 00000200
001013 000 COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHR,IHBI,LT,LAST 00000300
001014 000 C 00000400
001015 000 C 00000500
001016 000 C GET CONCENTRATED LOAD IF THERE IS ANY 00000600
001017 000 WRITE(6,677) 00000700
001018 000 DO 920 NC = 1,NAPC 00000800
001019 000 READ(5,540) NODE,PZ,PK 00000900
001020 000 WRITE(6,688)NODE,PZ,PK 00001000
001021 000 II = (NODE-1) * 2 00001100
001022 000 APF(II+1) = PZ 00001200
001023 000 APF(II+2) = PR 00001300
001024 000 ULOAD = PZ 00001400

```

```

001025      000      920 CONTINUE                                00001500
001026      000      540 FORMAT(I5,2F15.6)                    00001600
001027      000      677 FORMAT(///10X,'APPLIED C. LOAD'//5X,'NODE',10X,'FORCE TO Z',5X,'  00001700
001028      000      * FORCE TO R'//)                            00001800
001029      000      688 FORMAT(5X,15,2(5X,E12.4))            00001900
001030      000      RETURN                                      00002000
001031      000      END                                          00002100
001032      000      WELT,SIH NASA*TPFS,EGLoad,,,114076121110
001033      000      SUBROUTINE EGLoad(ULOAD)                    00000100
001034      000      C                                          00000200
001035      000      C      THIS SUBROUTINE IS TO CALCULATE THE EQUIVALENT NODAL LOADS.  00000300
001036      000      C                                          00000400
001037      000      PARAMETER N0US=300,NELS=260,NF=20000,MAX=600  00000500
001038      000      COMMON /BLK2/ ID(N0US,2),IJKL(NELS,4),DF1,DE2  00000600
001039      000      COMMON /BLK3/ XK(NF),APF(MAX),TMAX,IHR,IHBI,LT,LAST  00000700
001040      000      COMMON /BLK6/ SIGX,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),  00000800
001041      000      1 R(N0US),Z(N0US),TOTDIS(MAX)                00000900
001042      000      PI = 3.1415926                             00001000
001043      000      XXI = 0.                                     00001100
001044      000      XXJ = 0.                                     00001200
001045      000      READ(5,500) NLDEL,ULOAD                    00001300
001046      000      WRITE(6,600) NLDEL                         00001400
001047      000      DO 200 I = 1,NLDEL                          00001500
001048      000      READ(5,510) NOL,NLFT,NRHT                  00001600
001049      000      KC = NRHT                                    00001700
001050      000      LC = NLFT                                    00001800
001051      000      EGL = (R(KC)**2-R(LC)**2)*PI*ULOAD/2.      00001900
001052      000      II = (KC-1) * 2 + 1                        00002000
001053      000      JJ = (LC-1) * 2 + 1                        00002100
001054      000      APF(II) = APF(II) + EGL                    00002200
001055      000      APF(JJ) = APF(JJ) + EGL                    00002300
001056      000      200 WRITE(6,620) NOL,LC,KC,EGL,ULOAD        00002400
001057      000      620 FORMAT(I6,5X,2I4,2F20.7)              00002500
001058      000      500 FORMAT(I5,F20.9)                       00002600
001059      000      510 FORMAT(3I5)                           00002700
001060      000      600 FORMAT(///'      CALCULATION OF EQUIVALENT NODAL LOAD'//  00002800
001061      000      *      '      NUMBER OF LOADED ELEMENTS = ',I4/  00002900
001062      000      *      ' ELEM NO----LOADED NODE----EQUIV LOAD----GIVEN U.LOAD'//  00003000
001063      000      610 FORMAT(2I10,2F15.7)                   00003100
001064      000      RETURN                                      00003200
001065      000      END                                          00003300
001066      000      WELT,SIH NASA*TPFS,ZERO,,,114100121110
001067      000      SUBROUTINE ZERO(A,N,M)                      00000100
001068      000      DIMENSION A(1)                             00000200
001069      000      K = N * M                                    00000300
001070      000      DO 100 I = 1,K                             00000400
001071      000      100 A(I) = 0.                                00000500
001072      000      RETURN                                      00000600
001073      000      END                                          00000700
001074      000      WMAP,IX
001075      000      WXT
001076      000      CONE PENETROMETER
001077      000      90 71      3 100      5 2
001078      000      25.
001079      000
001080      000      7000000.      0.3      10.      0.45
001081      000      35.      0.006      0.7      0.76      0.0157      33.      0.07      73.

```

| | | | | | | |
|--------|-----|---------|---------|---|---|----|
| 001082 | 000 | .0000 | .0000 | 0 | 1 | 1 |
| 001083 | 000 | .0000 | .7050 | 0 | 1 | 2 |
| 001084 | 000 | .0000 | 1.4100 | 0 | 1 | 3 |
| 001085 | 000 | .0000 | 1.4100 | 0 | 0 | 4 |
| 001086 | 000 | .0000 | 1.9000 | 0 | 0 | 5 |
| 001087 | 000 | .0000 | 2.5000 | 0 | 0 | 6 |
| 001088 | 000 | .0000 | 3.3000 | 0 | 0 | 7 |
| 001089 | 000 | .0000 | 4.5000 | 0 | 0 | 8 |
| 001090 | 000 | .0000 | 6.3000 | 0 | 0 | 9 |
| 001091 | 000 | .0000 | 8.5000 | 0 | 0 | 10 |
| 001092 | 000 | .0000 | 10.0000 | 0 | 1 | 11 |
| 001093 | 000 | 1.8000 | .0000 | 0 | 1 | 12 |
| 001094 | 000 | 1.8000 | .5000 | 0 | 1 | 13 |
| 001095 | 000 | 1.8000 | .9400 | 0 | 1 | 14 |
| 001096 | 000 | 1.8000 | .9400 | 0 | 0 | 15 |
| 001097 | 000 | 1.8000 | 1.3400 | 0 | 0 | 16 |
| 001098 | 000 | 1.8000 | 1.9400 | 0 | 0 | 17 |
| 001099 | 000 | 1.8000 | 2.7400 | 0 | 0 | 18 |
| 001100 | 000 | 1.8000 | 3.9400 | 0 | 0 | 19 |
| 001101 | 000 | 1.8000 | 5.7400 | 0 | 0 | 20 |
| 001102 | 000 | 1.8000 | 7.9400 | 0 | 0 | 21 |
| 001103 | 000 | 1.8000 | 10.0000 | 0 | 1 | 22 |
| 001104 | 000 | 3.6000 | .0000 | 0 | 1 | 23 |
| 001105 | 000 | 3.4000 | .2500 | 0 | 1 | 24 |
| 001106 | 000 | 3.6000 | .4700 | 0 | 1 | 25 |
| 001107 | 000 | 3.6000 | .4700 | 0 | 0 | 26 |
| 001108 | 000 | 3.6000 | .8700 | 0 | 0 | 27 |
| 001109 | 000 | 3.6000 | 1.4700 | 0 | 0 | 28 |
| 001110 | 000 | 3.6000 | 2.2700 | 0 | 0 | 29 |
| 001111 | 000 | 3.6000 | 3.4700 | 0 | 0 | 30 |
| 001112 | 000 | 3.6000 | 5.2700 | 0 | 0 | 31 |
| 001113 | 000 | 3.6000 | 7.4700 | 0 | 0 | 32 |
| 001114 | 000 | 3.6000 | 10.0000 | 0 | 1 | 33 |
| 001115 | 000 | 5.3000 | .0000 | 0 | 1 | 34 |
| 001116 | 000 | 5.3000 | .0000 | 0 | 0 | 35 |
| 001117 | 000 | 5.3000 | .4000 | 0 | 0 | 36 |
| 001118 | 000 | 5.3000 | 1.0000 | 0 | 0 | 37 |
| 001119 | 000 | 5.3000 | 1.8000 | 0 | 0 | 38 |
| 001120 | 000 | 5.3000 | 3.0000 | 0 | 0 | 39 |
| 001121 | 000 | 5.3000 | 4.8000 | 0 | 0 | 40 |
| 001122 | 000 | 5.3000 | 7.0000 | 0 | 0 | 41 |
| 001123 | 000 | 5.3000 | 10.0000 | 0 | 1 | 42 |
| 001124 | 000 | 7.4000 | .0000 | 0 | 1 | 43 |
| 001125 | 000 | 7.4000 | .4000 | 0 | 0 | 44 |
| 001126 | 000 | 7.4000 | 1.0000 | 0 | 0 | 45 |
| 001127 | 000 | 7.4000 | 1.8000 | 0 | 0 | 46 |
| 001128 | 000 | 7.4000 | 3.0000 | 0 | 0 | 47 |
| 001129 | 000 | 7.4000 | 4.8000 | 0 | 0 | 48 |
| 001130 | 000 | 7.4000 | 7.0000 | 0 | 0 | 49 |
| 001131 | 000 | 7.4000 | 10.0000 | 0 | 1 | 50 |
| 001132 | 000 | 10.0000 | .0000 | 0 | 1 | 51 |
| 001133 | 000 | 10.0000 | .4000 | 0 | 0 | 52 |
| 001134 | 000 | 10.0000 | 1.0000 | 0 | 0 | 53 |
| 001135 | 000 | 10.0000 | 1.8000 | 0 | 0 | 54 |
| 001136 | 000 | 10.0000 | 3.0000 | 0 | 0 | 55 |
| 001137 | 000 | 10.0000 | 4.8000 | 0 | 0 | 56 |
| 001138 | 000 | 10.0000 | 7.0000 | 0 | 0 | 57 |

| | | | | | | |
|--------|-----|---------|---------|---|---|----|
| 001139 | 000 | 10.0000 | 10.0000 | 0 | 1 | 58 |
| 001140 | 000 | 14.0000 | .0000 | 0 | 1 | 59 |
| 001141 | 000 | 14.0000 | .4000 | 0 | 0 | 60 |
| 001142 | 000 | 14.0000 | 1.0000 | 0 | 0 | 61 |
| 001143 | 000 | 14.0000 | 1.8000 | 0 | 0 | 62 |
| 001144 | 000 | 14.0000 | 3.0000 | 0 | 0 | 63 |
| 001145 | 000 | 14.0000 | 4.8000 | 0 | 0 | 64 |
| 001146 | 000 | 14.0000 | 7.0000 | 0 | 0 | 65 |
| 001147 | 000 | 14.0000 | 10.0000 | 0 | 1 | 66 |
| 001148 | 000 | 19.0000 | .0000 | 0 | 1 | 67 |
| 001149 | 000 | 19.0000 | .4000 | 0 | 0 | 68 |
| 001150 | 000 | 19.0000 | 1.0000 | 0 | 0 | 69 |
| 001151 | 000 | 19.0000 | 1.8000 | 0 | 0 | 70 |
| 001152 | 000 | 19.0000 | 3.0000 | 0 | 0 | 71 |
| 001153 | 000 | 19.0000 | 4.8000 | 0 | 0 | 72 |
| 001154 | 000 | 19.0000 | 7.0000 | 0 | 0 | 73 |
| 001155 | 000 | 19.0000 | 10.0000 | 0 | 1 | 74 |
| 001156 | 000 | 25.0000 | .0000 | 0 | 1 | 75 |
| 001157 | 000 | 25.0000 | .4000 | 0 | 0 | 76 |
| 001158 | 000 | 25.0000 | 1.0000 | 0 | 0 | 77 |
| 001159 | 000 | 25.0000 | 1.8000 | 0 | 0 | 78 |
| 001160 | 000 | 25.0000 | 3.0000 | 0 | 0 | 79 |
| 001161 | 000 | 25.0000 | 4.8000 | 0 | 0 | 80 |
| 001162 | 000 | 25.0000 | 7.0000 | 0 | 0 | 81 |
| 001163 | 000 | 25.0000 | 10.0000 | 0 | 1 | 82 |
| 001164 | 000 | 33.0000 | .0000 | 1 | 1 | 83 |
| 001165 | 000 | 33.0000 | .4000 | 1 | 1 | 84 |
| 001166 | 000 | 33.0000 | 1.0000 | 1 | 1 | 85 |
| 001167 | 000 | 33.0000 | 1.8000 | 1 | 1 | 86 |
| 001168 | 000 | 33.0000 | 3.0000 | 1 | 1 | 87 |
| 001169 | 000 | 33.0000 | 4.8000 | 1 | 1 | 88 |
| 001170 | 000 | 33.0000 | 7.0000 | 1 | 1 | 89 |
| 001171 | 000 | 33.0000 | 10.0000 | 1 | 1 | 90 |
| 001172 | 000 | 12 13 | 2 1 | | | |
| 001173 | 000 | 13 14 | 3 2 | | | |
| 001174 | 000 | 23 24 | 13 12 | | | |
| 001175 | 000 | 24 25 | 14 13 | | | |
| 001176 | 000 | 34 25 | 24 23 | | | |
| 001177 | 000 | 14 15 | 4 3 | | | |
| 001178 | 000 | 25 26 | 15 14 | | | |
| 001179 | 000 | 34 35 | 26 25 | | | |
| 001180 | 000 | 15 16 | 5 4 | | | |
| 001181 | 000 | 26 27 | 16 14 | | | |
| 001182 | 000 | 35 36 | 27 26 | | | |
| 001183 | 000 | 16 17 | 6 5 | | | |
| 001184 | 000 | 27 28 | 17 16 | | | |
| 001185 | 000 | 36 37 | 28 27 | | | |
| 001186 | 000 | 17 18 | 7 6 | | | |
| 001187 | 000 | 28 29 | 28 17 | | | |
| 001188 | 000 | 37 38 | 29 28 | | | |
| 001189 | 000 | 18 19 | 8 7 | | | |
| 001190 | 000 | 29 30 | 19 18 | | | |
| 001191 | 000 | 38 39 | 30 29 | | | |
| 001192 | 000 | 19 20 | 9 8 | | | |
| 001193 | 000 | 30 31 | 20 19 | | | |
| 001194 | 000 | 39 40 | 31 30 | | | |
| 001195 | 000 | 20 21 | 10 9 | | | |

| | | | | | |
|--------|-----|----|-----|----|----|
| 001196 | 000 | 31 | 32 | 21 | 20 |
| 001197 | 000 | 40 | 41 | 32 | 31 |
| 001198 | 000 | 21 | 22 | 11 | 10 |
| 001199 | 000 | 32 | 33 | 22 | 21 |
| 001200 | 000 | 41 | 42 | 33 | 32 |
| 001201 | 000 | 43 | 44 | 36 | 35 |
| 001202 | 000 | 44 | 45 | 37 | 36 |
| 001203 | 000 | 45 | 46 | 38 | 37 |
| 001204 | 000 | 46 | 47 | 39 | 38 |
| 001205 | 000 | 47 | 48 | 40 | 39 |
| 001206 | 000 | 48 | 49 | 41 | 40 |
| 001207 | 000 | 49 | 50 | 42 | 41 |
| 001208 | 000 | 51 | 52 | 44 | 43 |
| 001209 | 000 | 52 | 53 | 45 | 44 |
| 001210 | 000 | 53 | 54 | 46 | 45 |
| 001211 | 000 | 54 | 55 | 47 | 46 |
| 001212 | 000 | 55 | 56 | 48 | 47 |
| 001213 | 000 | 56 | 57 | 49 | 48 |
| 001214 | 000 | 57 | 58 | 50 | 49 |
| 001215 | 000 | 59 | 60 | 52 | 51 |
| 001216 | 000 | 60 | 61 | 53 | 52 |
| 001217 | 000 | 61 | 62 | 54 | 53 |
| 001218 | 000 | 62 | 63 | 55 | 54 |
| 001219 | 000 | 63 | 64 | 56 | 55 |
| 001220 | 000 | 64 | 65 | 57 | 56 |
| 001221 | 000 | 65 | 66 | 58 | 57 |
| 001222 | 000 | 67 | 68 | 60 | 59 |
| 001223 | 000 | 68 | 69 | 61 | 60 |
| 001224 | 000 | 69 | 70 | 62 | 61 |
| 001225 | 000 | 70 | 71 | 63 | 62 |
| 001226 | 000 | 71 | 72 | 64 | 63 |
| 001227 | 000 | 72 | 73 | 65 | 64 |
| 001228 | 000 | 73 | 74 | 66 | 65 |
| 001229 | 000 | 75 | 76 | 68 | 67 |
| 001230 | 000 | 76 | 77 | 69 | 68 |
| 001231 | 000 | 77 | 78 | 70 | 69 |
| 001232 | 000 | 78 | 79 | 71 | 70 |
| 001233 | 000 | 79 | 80 | 72 | 71 |
| 001234 | 000 | 80 | 81 | 73 | 72 |
| 001235 | 000 | 81 | 82 | 74 | 73 |
| 001236 | 000 | 83 | 84 | 76 | 75 |
| 001237 | 000 | 84 | 85 | 77 | 76 |
| 001238 | 000 | 85 | 86 | 78 | 77 |
| 001239 | 000 | 86 | 87 | 79 | 78 |
| 001240 | 000 | 87 | 88 | 80 | 79 |
| 001241 | 000 | 88 | 89 | 81 | 80 |
| 001242 | 000 | 89 | 90 | 82 | 81 |
| 001243 | 000 | 2 | 9 | 4 | 10 |
| 001244 | 000 | 2 | 50. | | |
| 001245 | 000 | 1 | 1 | 2 | |
| 001246 | 000 | 2 | 2 | 3 | |

30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71

APPENDIX 2

DATA INPUT FORMAT

Card 1: FORMAT (20A4)

TITLE - Title of the problem

Card 2: FORMAT (10I5)

(1) INODE - No. of nodes

(2) NELEM - No. of elements

(3) NAPC - No. of applied point load

(4) NBC - No. of interface element

(5) NINCR - No. of load increment

(6) NCYCL - a dummy

(7) ICASE = 0 for plastic analysis

(8) NRIGD - No. of rigid element

(9) NULOAD - No. of uniformly loaded element

Card 3: FORMAT (8F10.4)

(1) YSTRS - a dummy

(2) DELL - "

(3) ZETA - "

(4) PMAX - Maximum load one wants to apply

(5) DLMAX - a dummy

Card 4: FORMAT (4F20.5)

(1) DZI - Shear modulus for interface element

(2) DRI - Rotational modulus for interface

Card 5: FORMAT (4F20.5)

- (1) EC - Modulus of elasticity for rigid element
- (2) XNUC - Poisson's ratio for rigid element
- (3) ES - Modulus of elasticity for soil
- (4) XNUS - Poisson's ratio for soil

Card 6: FORMAT (8F10.4)

- (1) PI - Angle of friction in degree
- (2) SMALLK - Swelling index
- (3) XI - Adhesion (for interface element)
- (4) VOIDI - Initial void ratio
- (5) PO - Initial density
- (6) DEPTH - Maximum depth of soil
- (7) ALAMDA - Compression index
- (8) EPSLON - Angle of friction for interface element

Card 7: FORMAT (2F10.4, 2I5)

- (1) Z(I) - Z - Coordinate value (downward positive)
- (2) R(I) - R - Coordinate value
- (3) IZ = 0 if free to Z-direction
= 1 if note
- (4) IR = 0 if free to R-direction
= 1 if note

Repeat INODE times in the order of node number

Card 8: FORMAT (4I5)

4 node numbers of an element in counter-clockwise. Repeat NELEM times in the order of element number. Ordering of element should be:

- (1) Rigid element (2) interface element
- (3) soil element

* Card(s) 9: FORMAT (10I5)

- (1) ID(I,1) - Element number left to I^{th} interface element
- (2) ID(L,2) - Element number right to I^{th} interface element

Repeat NBC/5 times with 5 sets of data on one card

* Not required if NBC=0

Card(s) 10: FORMAT (I5, 2F15.6)

- (1) NODE - Node number with point load
- (2) PZ - Z-component
- (3) PR - R-component

Repeat NAPC times

Not required if NAPC=0

** Card 11: FORMAT (I5, F20.9)

- (1) NLDEL - No. of uniformly loaded elements
- (2) ULOAD - Load intensity (compression is positive)

** Card(s) 12: FORMAT (3I5)

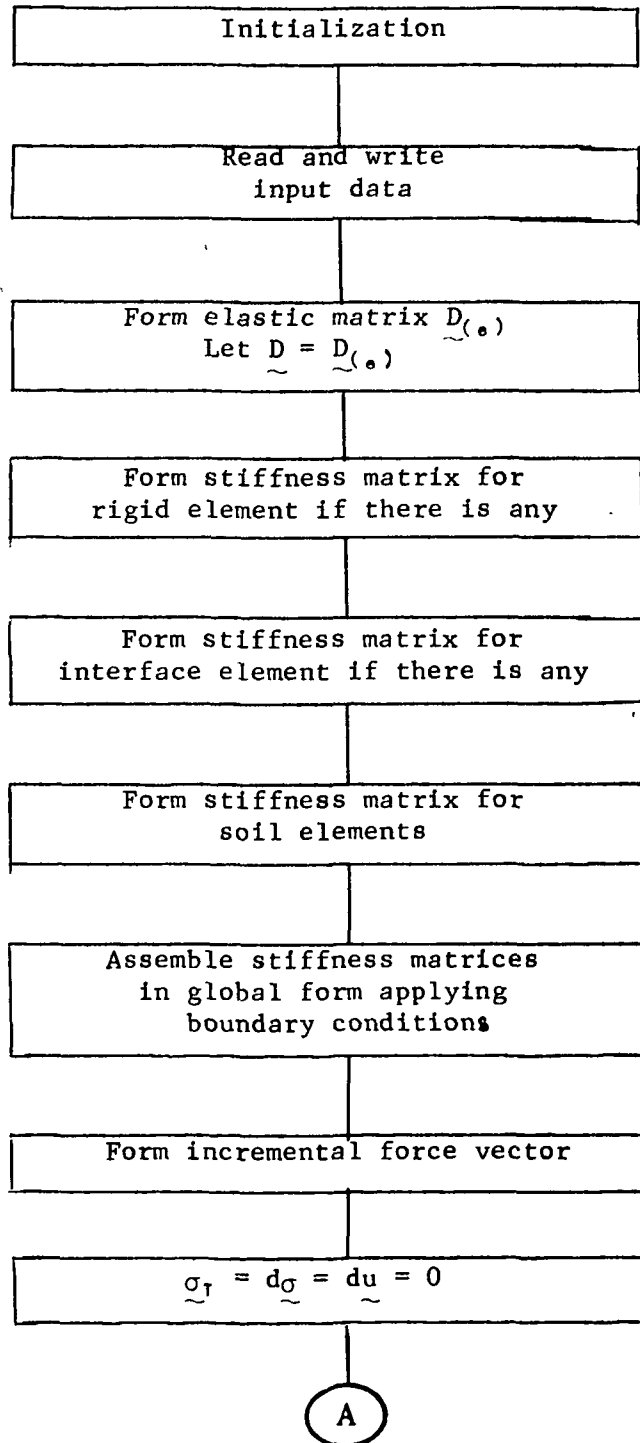
- (1) NOL - Loaded element number
- (2) NLFT - Node No. at left
- (3) NRHT - Node No. at right

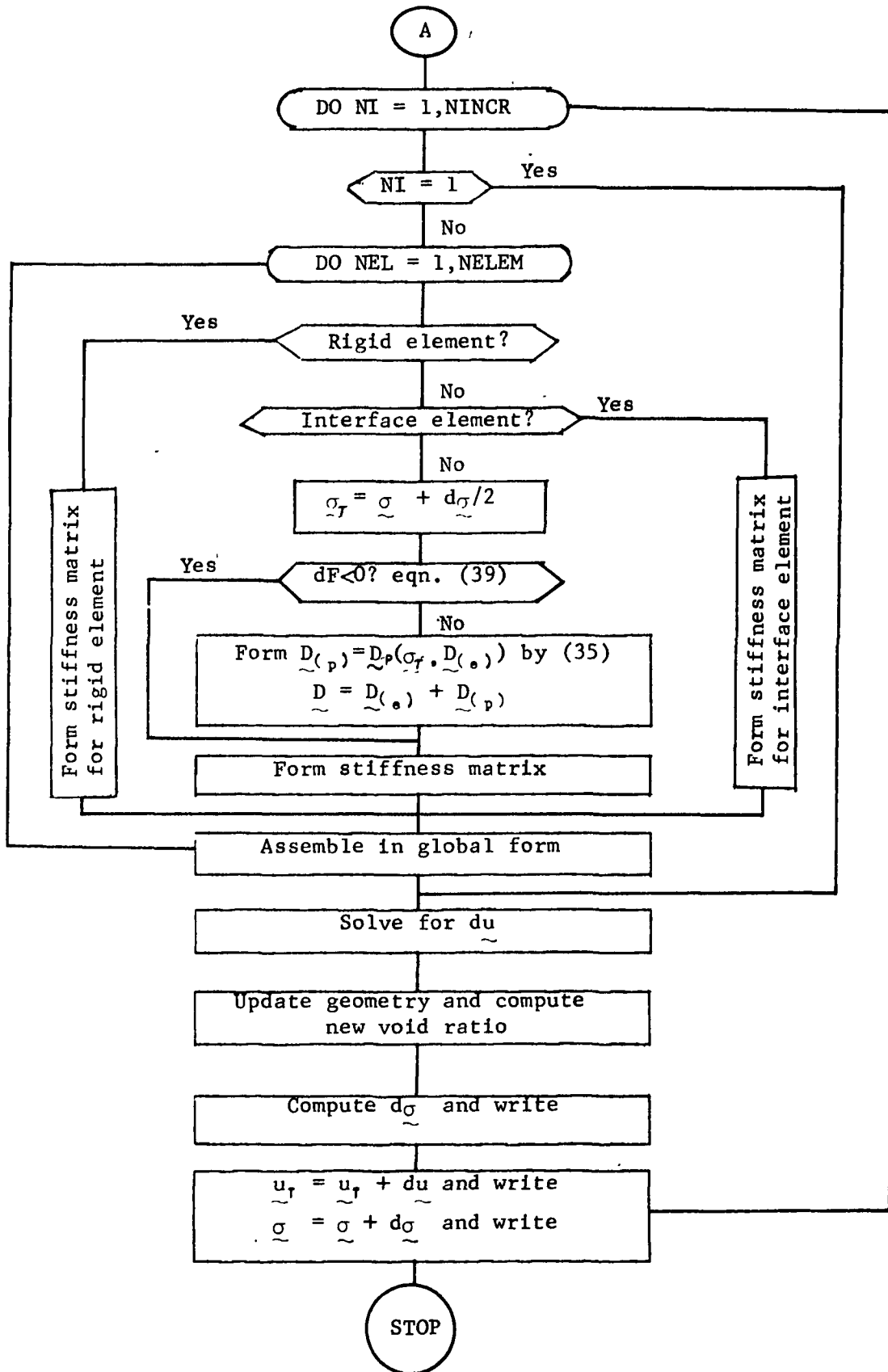
Repeat NLDEL times

** Not required if NULOAD=0

APPENDIX 3

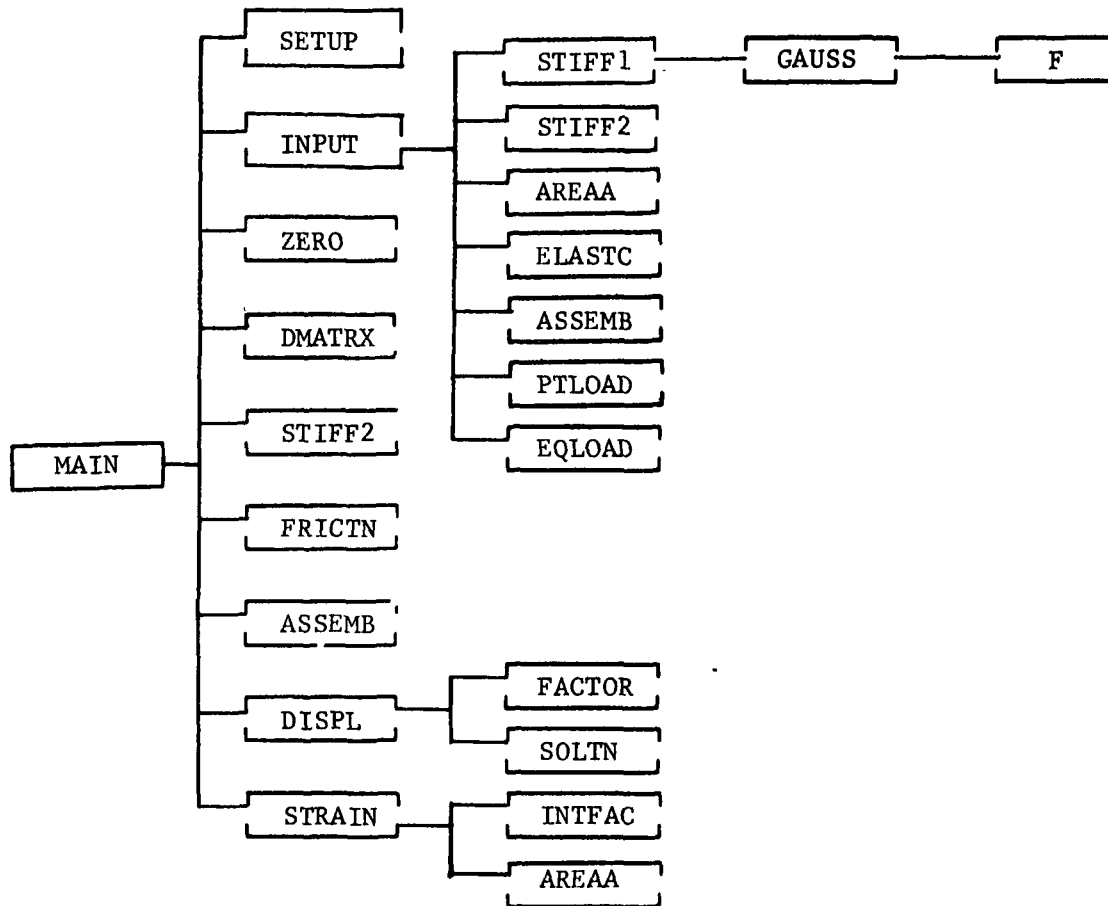
FLOW CHART





APPENDIX 4

SUBROUTINE ORGANIZATION CHART



APPENDIX 5

DESCRIPTIONS OF SUBROUTINES

| Subroutine Name | DESCRIPTIONS |
|--------------------|--|
| SETUP | Assigns necessary constants for integrations |
| INPUT | Reads and writes input data, and gets ready for the first linear elastic solution. |
| ZERO | Clear a given matrix. |
| DMATRX | Computes $D_{(p)}$ and forms $D = D_{(e)} + D_{(p)}$ if $dF \geq 0$ for a given element. |
| STIFF 1 | Forms submatrices of stiffness matrices |
| GAUSS | Integrates by Gaussian quadrature |
| F | Gives functions to be integrated |
| FRICTN | Updates interface moduli and forms interface element stiffness. |
| AREAA | Computes cross sectional area of an element. |
| ELASTC | Forms elastic matrix $D_{(e)}$ |
| ASSEMB | Assembles stiffness matrices into global stiffness matrix applying boundary conditions. |
| PTLOAD | Reads and writes applied point load if there is any. |
| EQLOAD | Reads and writes applied uniform load and computes equivalent nodal forces. |
| DISPL | Calls FACTOR and SOLTN, and writes du for first 10 nodes. |
| FACTOR | Factors the given simultaneous eqns. in one dimensional array. |
| SOLTN | Backward substitution is performed to give a set of solutions. |
| STRAIN | Computes incremental strains and stresses. Also |

computes new void ratio.

INTFAC Computes incremental stresses for interface element
if there is any.

STIFF2 Forms stiffness matrix using \underline{D} .

PART II

DYNAMICS OF WHEEL-SOIL INTERACTION

II -1. INTRODUCTION

Deformations and stresses of soil media under a moving wheel are complex phenomena. A rate-dependent inelastic behavior associated with inertia effects must be considered. Somewhat simplified analyses have been reported by various authors. Earlier contributions to the wheel-soil interaction by Bekker were followed by Micklethwaite [2], Evans [3], Uffelman [4], and Bekker [5]. Rigorous experimental and theoretical studies on this subject have also been reported by Onaffeko and Reece [6], Wong and Reece [7,8]. Yong and Webb [9] and Schuring [10] studied energy dissipation in soil-wheel interaction from the viewpoint of viscoplasticity. Windisch and Yong [11] further examined the strain-rate phenomena and presented a method of computing soil displacements and strain rates from the experiment-based "marker position". In contrast to these studies, Perumpral, Liljedahl and Perloff [12] used the finite element method to calculate stresses and deformations due to a rigid wheel interaction. They used variable modulus of elasticity determined from the stress-strain curve of the triaxial tests but ignored the effects of inertia and rate-dependency.

Elsamny and Ghobarah [13] studied the stress field in the soil mass under the loading of a rigid cylindrical wheel on the verge of

spinning. However, the fact that the kinematic characteristics of the wheel and the velocity boundary conditions on the wheel-soil interface is ignored has been criticized by Wong [14]. More recently, Kloc [15] presented analytical formulations on mechanical interaction of a driven roller on soil slopes. In this study, a gravitating cohesive-frictional soil was considered with Kötter's quasi-static equilibrium equations applied to a plastic stress configuration (Mohr-Coulomb criteria) satisfying Shield's velocity conditions along the characteristic lines. Energy dissipation was not considered in this study.

In the present study we propose a rational approach in which the rate-dependent inelastic properties together with effects of inertia are adequately taken into account. Equilibrium conditions for wheel-soil interaction reported by Onaffeko and Reece [6] and Wong and Reece [7] are used to obtain radial and tangential stresses at the interface. Although the nonisothermal conditions may be considered without special difficulties in the framework of continuum mechanics and irreversible thermodynamic process, the present study is limited to an isothermal condition. The Mohr-Coulomb failure criterion appears to dominate most of the wheel-soil interaction studies. However, in view of the fact that the soil behaves as a strain-hardening material, in general, rather than perfectly plastic or rigid plastic material, we will overcome such deficiency by using the concept of critical state soil mechanics.

In what follows we make use of the internal state variable approaches of Coleman and Gurtin [16] and Perzyna and Wojno [17]

However, a basic difference from their approach is introduced in the present study such that the free energy functional containing inelastic behavior is not considered smooth for its entire domain of histories. Rather, we assume a form of discretized free energy as a function of elastic strains, plastic strains and internal or hidden variables of incremental quantity considered to be valid only for a small time interval or a fraction of loading increments. Here the hidden variables may represent a viscous or physicochemical behavior, properties other than what is commonly known as "elastic" and "plastic". Once the form of incremental free energy containing all nonlinear functions is prescribed for a small time interval, then the superposition of these nonlinear terms is permissible. Namely, the plastic material kernel may be calculated from the independent viscoelastic responses within this small time interval. Thus the histories can be carried over from one time increment to another until desired histories are completed. This will be accomplished by a suitable difference operator.

To represent inelastic behavior of soil we use the concept of critical state [18] and yield surface of Roscoe and Burland [19]. A derivation of the plastic tangent stiffness matrix based on this theory in the context of incremental theory of plasticity and its finite element applications were presented in Part I of this report. It should be noted that the particular internal state variable approach used here in conjunction with incremental free energy expression leads to a valid coupling of the completely independent plasticity theory and the rate dependent hidden variables.

Numerical examples are presented to demonstrate effectiveness of the present method. The well-known finite element method [20,21] is

utilized in the computation.

II - 2. BALANCE OF ENERGY AND LINEAR MOMENTUM

We record here the principle of conservation of energy which states that the time rate of change of the kinetic energy k plus the internal energy U is equal to R , the mechanical power on the system.

$$\dot{k} + \dot{U} = R \quad (1)$$

Here the superposed dot indicates a time rate, and

$$k = \frac{1}{2} \int_V \rho v_i v_i dV \quad (2)$$

$$U = \int_V \rho \epsilon dV \quad (3)$$

$$R = \int_V \rho F^j v_j dV + \int_A s^{ij} v_j n_i dA \quad (4a)$$

in which ρ is the density, v_i is the velocity component; ϵ is the internal energy density; F^i is the body force; s^{ij} is the surface traction; and n_i is the unit normal to the surface. Using the Green - Gauss theorem, (4a) becomes

$$R = \int_V (\rho F^i v_i + \sigma^{ij} v_{j,i} + \sigma_{,i}^{ij} v_j) dV \quad (4b)$$

Now, inserting (2) and (4b) into (1) yields

$$\int_V [(\sigma_{,i}^{ij} + \rho F^j - \rho a^j) v_j - \rho \dot{\epsilon} + \sigma^{ij} v_{j,i}] dV = 0 \quad (5)$$

For the principle of balance of linear momentum to hold and for arbitrary volumes we must have

$$\sigma_{,i}^{ij} + \rho F^j - \rho a^j = 0 \quad (6)$$

and

$$\rho \dot{\epsilon} = \sigma^{ij} v_{j,i} = \sigma^{ij} \dot{\gamma}_{ij} \quad (7)$$

Here σ^{ij} and γ_{ij} are the stress tensor and strain tensor; the comma denotes ordinary differentiation; and a^j is the acceleration. It should be noted that equations (2) through (7) refer to rectangular cartesian coordinates. We regard (7) as the balance of energy.

II - 3. INCREMENTAL FREE ENERGY FUNCTIONS

In view of the earlier discussion our objective is to propose a form of free energy functions in incremental quantity such that the non-smooth or inelastic strains may be included for a small time interval Δt . For isothermal conditions, the incremental free energy $\phi(\Delta t)$ and stresses $\sigma^{ij}(\Delta t)$ are assumed to be functions of incremental strains $\gamma_{ij}(\Delta t) = \gamma_{ij}^{(e)}(\Delta t) + \gamma_{ij}^{(p)}(\Delta t)$ and incremental internal state variables (or hidden variables) $\alpha_{ij}^{(r)}(\Delta t) = \alpha_{ij}^{(r)(e)}(\Delta t) + \alpha_{ij}^{(r)(p)}(\Delta t)$ where (e) and (p) represent elastic and plastic components, respectively. This statement may be given by

$$\phi(\Delta t) = \hat{\phi}[\gamma_{ij}^{(e)}(\Delta t), \gamma_{ij}^{(p)}(\Delta t), \alpha_{ij}^{(r)(e)}(\Delta t), \alpha_{ij}^{(r)(p)}(\Delta t)] \quad (8)$$

$$\sigma^{ij}(\Delta t) = \hat{\sigma}[\gamma_{ij}^{(e)}(\Delta t), \gamma_{ij}^{(p)}(\Delta t), \alpha_{ij}^{(r)(e)}(\Delta t), \alpha_{ij}^{(r)(p)}(\Delta t)] \quad (9)$$

For isothermal conditions, the free energy is the same as the internal energy so that

$$\rho \dot{\phi} = \rho \dot{\epsilon} = \sigma^{ij} \dot{\gamma}_{ij}$$

or for the small time interval Δt ,

$$\sigma^{ij}(\Delta t) \dot{\phi}(\Delta t) = \sigma^{ij}(\Delta t) (\dot{\gamma}_{ij}^{(e)}(\Delta t) + \dot{\gamma}_{ij}^{(p)}(\Delta t)) \quad (10)$$

At this point we introduce here the incremental form of free energy in a truncated Taylor series expansion,

$$\begin{aligned} \rho \phi(\Delta t) = & \frac{1}{2} E^{ijkl} Y_{ij}^{(e)} Y_{kl}^{(e)} + \frac{1}{2} E^{*ijkl} Y_{ij}^{(p)} Y_{kl}^{(p)} \\ & + \frac{1}{2} \sum_{r=1}^n E_{(r)}^{ijkl} (\alpha_{ij}^{(e)} + \alpha_{ij}^{(p)}) (\alpha_{kl}^{(e)} + \alpha_{kl}^{(p)}) \\ & + \sum_{r=1}^n E_{(r)}^{ijkl} (\alpha_{kl}^{(e)} + \alpha_{kl}^{(p)}) (Y_{ij}^{(e)} + Y_{ij}^{(p)}) \end{aligned} \quad (11)$$

where E^{ijkl} and E^{*ijkl} represent tensors of elastic and plastic moduli, respectively; $E_{(r)}^{ijkl}$ are stiffness constants associated with the internal variables. Note that (11) has the form of truncated Taylor series expansion only to include quadratic terms. However, the product term of $Y_{ij}^{(e)}$ and $Y_{ij}^{(p)}$ is missing. This is because the coupling of elastic and plastic strains can be obtained using any one of the failure theories and an explicit material kernel relating the product of $Y_{ij}^{(e)}$ and $Y_{ij}^{(p)}$ is nonexistent.

Lastly, $\alpha_{ij}^{(r)}$ defined here as the internal variables represent time dependent physicochemical properties or simply a viscous behavior which may be expressed as

$$\alpha_{ij}^{(r)} = \int_0^t \exp\left[-\frac{(t-\tau)}{T_{(r)}}\right] \dot{Y}_{ij}(\tau) d\tau \quad (12)$$

where τ is the time variable and $T_{(r)}$ is the relaxation time. In order to facilitate an explicit integration we assume a linear variation of \dot{Y}_{ij} within the time interval Δt given by

$$\dot{Y}_{ij}(\theta) = \dot{Y}_{ij}(\theta-\Delta t) + \frac{\tau-(t-\Delta t)}{\Delta t} (\dot{Y}_{ij}(\theta=1) - \dot{Y}_{ij}(\theta)) \quad (13)$$

where s is the current time step. Substituting (13) in (12) and performing integration we obtain

$$\alpha_{ij}^{(r)}(s) = A^{(r)} \alpha_{ij}^{(r)}(s-1) + B^{(r)} \dot{Y}_{ij}^{(r)}(s-1) + C^{(r)} \dot{Y}_{ij}^{(r)}(s) \quad (14)$$

in which

$$A^{(r)} = \exp\left(\frac{-\Delta t}{T_{(r)}}\right), \quad B^{(r)} = T_{(r)}(D^{(r)} - A^{(r)})$$

$$C^{(r)} = T_{(r)}(1 - D^{(r)}), \quad D^{(r)} = \frac{T_{(r)}}{\Delta t}(1 - A^{(r)})$$

The derivation of these parameters is given in Appendix 1.

Rewriting (10) for the current time step (s) as

$$\rho \left\{ \frac{\partial \Phi(s)}{\partial Y_{ij}^{(e)}(s)} \dot{Y}_{ij}^{(e)}(s) + \frac{\partial \Phi(s)}{\partial Y_{ij}^{(p)}(s)} \dot{Y}_{ij}^{(p)}(s) + \frac{\partial \Phi(s)}{\partial \alpha_{ij}^{(r)}(s)} \alpha_{ij}^{(r)}(s) + \frac{\partial \Phi(s)}{\partial \alpha_{ij}^{(p)}(s)} \alpha_{ij}^{(p)}(s) \right\} - \sigma^{ij}(s) (\dot{Y}_{ij}^{(e)}(s) + \dot{Y}_{ij}^{(p)}(s)) = 0 \quad (15)$$

and substituting (14) and (11) into (15) yields

$$\begin{aligned} & \{ E^{ijkl} Y_{kl}^{(e)}(s) + \sum_{r=1}^n g_{(r)}^{ijkl} (A^{(r)} \alpha_{kl}^{(r)}(s-1) + B^{(r)} \dot{Y}_{kl}^{(r)}(s-1) + C^{(r)} \dot{Y}_{kl}^{(r)}(s) \\ & - \sigma^{ij}(s) \} \dot{Y}_{ij}^{(e)}(s) + \sum_{r=1}^n g_{(r)}^{ijkl} (\alpha_{ij}^{(r)}(s) \alpha_{kl}^{(p)}(s) + \alpha_{kl}^{(p)}(s) \alpha_{ij}^{(e)}(s)) \\ & + \alpha_{kl}^{(e)}(s) \dot{Y}_{ij}^{(p)}(s) + Y_{ij} \alpha_{kl}^{(p)}(s) - \sigma^{ij}(s) \dot{Y}_{ij}^{(p)}(s) = 0 \end{aligned}$$

Since all variations other than $\dot{Y}_{ij}^{(e)}$ are not arbitrary we must have

the relationship

$$\sigma^{ij}(s) = E^{ijkl} Y_{kl}^{(e)}(s) + \sum_{r=1}^n g_{(r)}^{ijkl} (A^{(r)} \alpha_{kl}^{(r)}(s-1) + B^{(r)} \dot{Y}_{kl}^{(r)}(s-1) + C^{(r)} \dot{Y}_{kl}^{(r)}(s)) \quad (16)$$

$$\begin{aligned}
 \frac{1}{E} \epsilon_{ijkl} \dot{\gamma}_{ij}^{(p)}(s) - \sigma_{ij}^{(s)} \dot{\gamma}_{ij}^{(p)} + \sum_{r=1}^n \epsilon_{ijkl}^{(r)} \left[\alpha_{ij}^{(r)}(s) \dot{\gamma}_{ij}^{(p)}(s) \right. \\
 \left. + \alpha_{kl}^{(r)}(s) \dot{\gamma}_{ij}^{(p)}(s) + \gamma_{ij}^{(r)} \dot{\alpha}_{kl}^{(r)}(s) \right] = 0
 \end{aligned} \tag{17}$$

Here (16) represents the relationship

$$\sigma_{ij} = \rho \frac{\partial \phi}{\partial \gamma_{ij}}$$

which states that the stresses are derivable from the free energy functions. It should be noted that, in our specific problem, this stress is due to an elastic strain and a law governing the plastic strain and stress is needed to obtain the stress due to a total strain. The relationship (17) may be considered as the dissipation which plays a significant role in heat conduction problems. However, for the isothermal conditions as considered in the present study, the entire terms of (17) need not be used in the analysis. Only the first term will be recovered as we apply a yield criterion in (16).

II - 4. INELASTIC RESPONSE

Extensive research has been carried out at Cambridge University by Roscoe and his colleagues [19] on the subject of the critical state soil mechanics. The yield criteria adopted here were originally proposed by Roscoe and Burland [19]. A plastic tangent matrix in the context of the incremental theory of plasticity was derived by the authors [22,23,24]. A new method of checking conditions of yielding is elaborated in Part I of this report. For the purpose of reference we repeat the expression for the incremental stress associated with rate-independent

elastoplastic behavior,

$$d\sigma^{ij} = (E^{ijkl} + \overset{*}{E}^{ijkl})d\gamma_{kl} \tag{18}$$

A close examination of (16) reveals that $\sigma^{ij(s)}$ is the total stress due to the elastic component of strain and internal variable for the current time step. On the other hand, (18) represents an incremental stress for a fraction of loading increments with inelastic strain coupled. It is then immediately clear that if the viscoelastic stress as given by (16) is used to calculate $\overset{*}{E}^{ijkl}$ within the time interval and if we proceed with (18) with iterative cycling for further updating $\overset{*}{E}^{ijkl}$ without participation of the viscous part of (16), then at the end of the time interval the total strain reached simply reflects the coupling of viscoelastic and plastic properties. Thus from (16) and (20), we obtain,

$$\begin{aligned} d\sigma^{ij(s)} = & E^{ijkl} d\gamma_{kl}^{(s)} + \sum_{r=1}^n \overset{r}{E}^{ijkl} \left(\overset{r}{A} d\alpha_{kl}^{(s-1)} \right. \\ & \left. + \overset{r}{B} d\dot{\gamma}_{kl}^{(s-1)} + \overset{r}{C} d\dot{\gamma}_{kl}^{(s)} \right) + \overset{*}{E}^{ijkl} d\dot{\gamma}_{kl}^{(s)} \end{aligned} \tag{19}$$

Note that viscoelastic strain is now associated with the total strain as coupling is established.

II - 5. FINITE ELEMENT EQUATIONS OF MOTION

The finite element method is widespread in engineering applications [11,12]. No elaboration on this method is attempted here.

In view of (7), (2) and (3) we rewrite (1) as

$$\int_V \rho \ddot{u}_i \dot{u}_i dV + \int_V \sigma^{ij} \dot{\gamma}_{ij} dV - \int_V \hat{F}^m \dot{u}_m dV = 0 \quad (20)$$

Here the body force $\rho F^m = \hat{F}^m$ alone is considered merely for simplicity. The surface traction can easily be added later if needed.

In the present study we use the plane strain isoparametric element with 4 corner nodes. This gives the linear variation of displacements in the form

$$u_i = \Psi_N u_i^N \quad (21)$$

where Ψ_N is the interpolation function and u_i^N is the nodal values of displacements u_i ($i = 1, 2$) and $N = 1, 2, 3, 4$.

The strain tensor is given by

$$\gamma_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i}) \quad (22)$$

Inserting (25) in (26) yields

$$\gamma_{ij} = A_{Nij}^k u_k^N \quad (23)$$

where

$$A_{Nij}^k = \frac{1}{2}(\Psi_{N,i} \delta_j^k + \Psi_{N,j} \delta_i^k) \quad (24)$$

In view of (21), (23) and (20) we have

$$\left\{ \int_V \rho \Psi_M \Psi_N dV \ddot{u}_k^M + \int_V \sigma^{ij} A_{Nij}^k dV - \int_V \hat{F}^k \Psi_N dV \right\} \dot{u}_k^N = 0 \quad (25)$$

For all arbitrary values of \dot{u}_k^N we require the terms inside the bracket to vanish, which yields

$$M_{MN} \ddot{u}_k^M + \int_V \sigma^{ij} A_{Nij}^k dV = F_{Nk} \quad (26)$$

where M_{MN} and F_{Nk} are the mass matrix and the force vector, respectively,

$$M_{MN} = \int_V \rho \psi_M \psi_N dV \quad (27)$$

$$F_{Nk} = \int_V F^k \psi_N dV \quad (28)$$

To obtain an incremental form of (26), we take a variation or induce a perturbation such that

$$M_{MN} d\dot{u}_k^M + \int_V d\sigma^{ij} A_{Nij}^k dV = dF_{Nk} \quad (29)$$

Introducing the incremental stress (19) into (29) yields

$$M_{MN} d\dot{u}_k^M + C_{MN}^{\ell k} d\dot{u}_\ell^M + \left(K_{MN}^{(e)\ell k} + K_{MN}^{(p)\ell k} \right) du_\ell^M = dF_{Nk} + dF_{Nk}^{(v)} \quad (30)$$

in which $C_{MN}^{\ell k}$, $K_{MN}^{(e)\ell k}$, and $K_{MN}^{(p)\ell k}$ are the viscosity matrix, elastic stiffness matrix and plastic stiffness matrix, respectively,

$$C_{MN}^{\ell k} = \int_V \sum_{r=1}^n \xi_{(r)}^{ijmn} C A_{Mij}^k A_{Nmn}^k dV \quad (31)$$

$$K_{MN}^{(e)\ell k} = \int_V E^{ijmn} A_{Mij}^k A_{Nmn}^k dV \quad (32)$$

$$K_{MN}^{(p)\ell k} = \int_V E^{*ijmn} A_{Mij}^k A_{Nmn}^k dV \quad (33)$$

The pseudo viscous load vector $dF_{Nk}^{(v)}$ is given by

$$dF_{Nk}^{(v)} = \int_V \sum_{r=1}^n \xi_{(r)}^{ijmn} A_{Nij}^k \alpha_{mn}^{(s-1)} A_{Mij}^k dV + \int_V \sum_{r=1}^n \xi_{(r)}^{ijmn} B A_{Nij}^k A_{Mmn}^k dV \{d\dot{u}_\ell^M(s-1)\} \quad (34)$$

The expression (30) is called the finite element equations of motion.

II - 6. SOLUTION PROCEDURE FOR INCREMENTAL EQUATIONS OF MOTION

A solution of (30) can easily be obtained by any scheme of direct numerical integration [13]. In this study, a constant acceleration for a small time increment is assumed, which gives a recurrence formula for displacements, velocities and accelerations in the form,

$$\left\{ M_{MN} + \frac{\Delta t}{2} C_{MN} + \frac{\Delta t^2}{4} \left(\overset{(s)}{K}_{MN}^{\ell k} + K_{MN}^{\ell k} \right) \right\} \ddot{u}_\ell^M(s) = dF_{Nk}^{(s)} - dF_{Nk}^{(v)(s)} - Q_{Nk}^{(s)} \quad (35)$$

where

$$Q_{Nk}^{(s)} = C_{MN} \left\{ \dot{d}u_k^M(s-1) + \frac{\Delta t^2}{2} \ddot{d}u_k^M(s-1) \right\} + \left(\overset{(s)}{K}_{MN}^{\ell k} + \overset{(p)}{K}_{MN}^{\ell k} \right) \left\{ du_\ell^M(s-1) + \frac{\Delta t^2}{4} \ddot{d}u_\ell^M(s-1) \right\} \quad (36)$$

$$\dot{d}u_\ell^M(s) = \dot{d}u_\ell^M(s-1) + \frac{\Delta t}{2} \ddot{d}u_\ell^M(s-1) + \frac{\Delta t}{2} \ddot{d}u_\ell^M(s) \quad (37)$$

$$du_\ell^M(s) = du_\ell^M(s-1) + \frac{\Delta t^2}{4} \ddot{d}u_\ell^M(s-1) + \frac{\Delta t^2}{4} \ddot{d}u_\ell^M(s) + \Delta t \dot{d}u_\ell^M(s-1) \quad (38)$$

Initially all terms associated with $(s-1)$ are zero and $\ddot{d}u_\ell^M(s)$ in (35) can be solved from given initial and boundary conditions. Subsequently, $\dot{d}u_\ell^M(s)$ and $du_\ell^M(s)$ are calculated from (38). These responses or histories are then carried to the next time increment and back to (35). However, for the second increment it is necessary to check yield con-

ditions and a standard incremental loading method of iteration [14] can be applied to each time increment with the total dynamic load on the structure.

II - 7. EQUIVALENT DYNAMIC WHEEL LOADS

Theoretical and experimental studies for the prediction of rigid and flexible wheel performance on soil have been reported by various authors as mentioned in Introduction. Onaffeko and Reece [6] presented practical procedures in determining radial and tangential stresses along the wheel-soil interface. Wong and Reece [7,8] derived expressions for sinkage, drawbar pull and torque input based on the plate penetration test but with considerations of the important aspects of the slip and the actual interaction between the wheels and soil.

In the present study the finite element equivalent nodal dynamic loadings are determined from the expressions for radial and tangential stresses given by Onaffeko and Reece [6] and explicit forms of these stresses as elaborated by Wong and Reece may also be used (See Appendix 2).

In order to compare the dynamic rate-dependent elastoplastic responses with the results of Perumpral, et al [12] who neglected the effects of inertia and rate-dependency, we consider here the identical geometry and material constants. The discretized wheel-soil medium is shown in Figure 1.

$E_s = 2,000 \text{ psi}$, $\nu_s = 0.45$
 $\omega = 36^\circ$, $e = 0.75$, $\nu = 0.05787 \text{ pci}$
 $T_{(r)} = 0.1 \text{ sec. (r = 1,2,3)}$
 $A = 0.05$, $\nu = 0.0001$

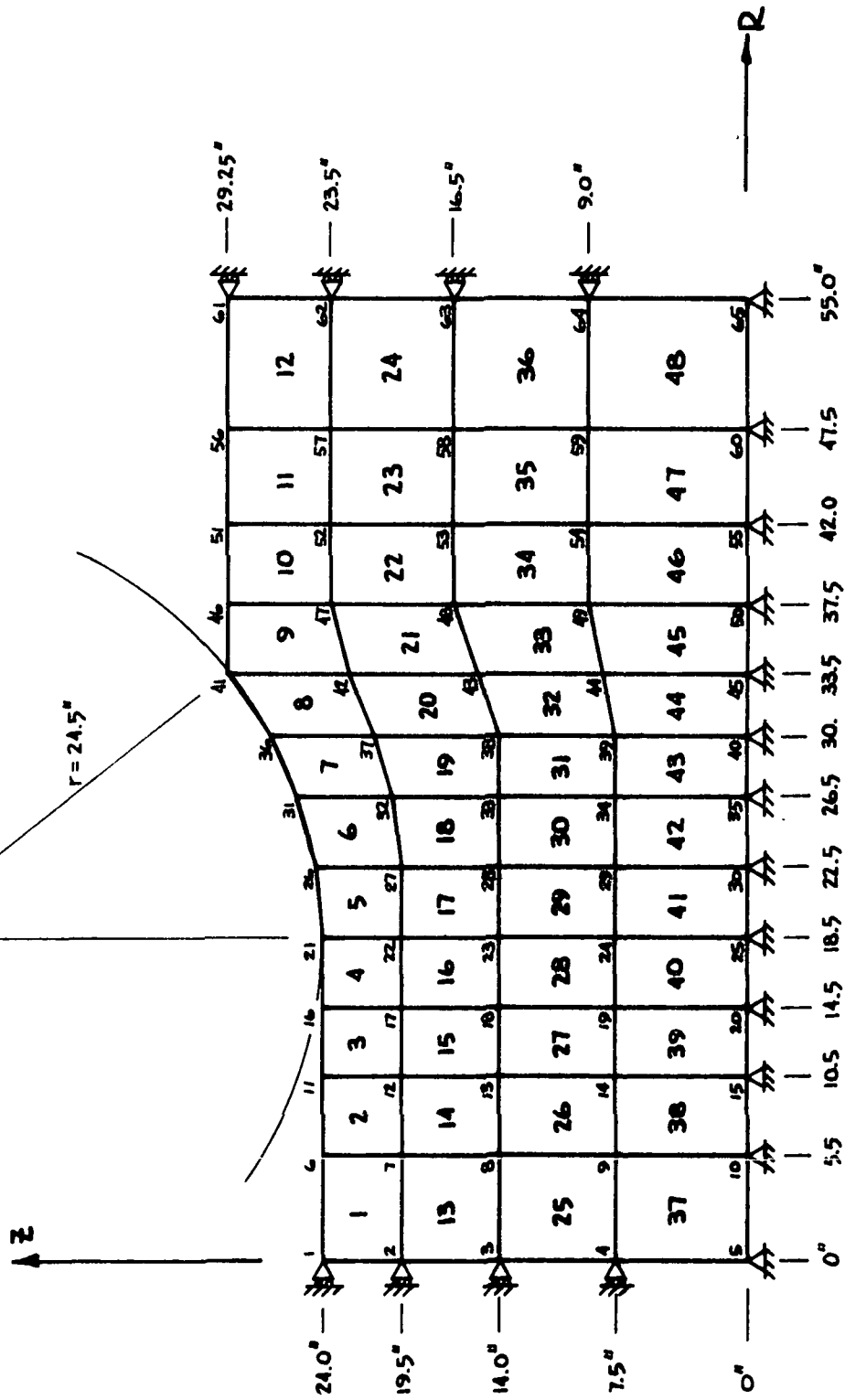


Figure 1: Wheel-Soil Interaction Geometry

Distributions of radial and tangential stresses on the rim of a 49 in. x 6 in. wide wheel on compact sand with 3.1% slip and 41.4% slip are shown in Figure 2 and Figure 3, respectively. Equivalent nodal loadings as calculated from the tributary area method in Figures 2 and 3 are shown in Figure 4. It is seen that the area under the curve corresponding to the wheel-soil contact area for each finite element may be conveniently approximated by the equivalent rectangular block. It should be noted that as the slippage increases the vertical downward loads decrease whereas the horizontal loads increase in the direction opposite to the wheel movement.

II - 8. DEFORMATION AND STRESS FIELDS

The equations of motion in assembled form for all finite elements are solved as described in Section II - 6. In order to compare the results for all possible effects, the computer program (Appendix 3) was written with many optional versions. Various cases studied include static analyses for elastic and elastoplastic responses and dynamic analyses for elastic, viscoelastic and viscoelastoplastic responses.

The material constants used are soil modulus $E_s = 2000$ psi, Poisson's ratio $\nu_s = 0.45$, angle of internal friction $\phi = 36^\circ$, density $\gamma = 0.05787$ pci, relaxation time $T_{(r)} = 0.1$ sec ($r = 1, 2, 3$), compression index $\lambda = 0.05$, and swelling index $\kappa = 0.0001$. These constants are chosen

to correspond to the compact sand which is used in the equivalent load representation as shown in Figures 2, 3, and 4. For dynamic analyses, a time increment $\Delta t = 0.0006$ Sec. for viscoelastoplastic response and $\Delta t = 0.0003$ Sec. for other responses are used.

Figure 5 shows these various responses at node No. 31. For static analyses, the elastoplastic displacement in the vertical direction is slightly larger than the elastic behavior. For dynamic analyses, the viscoelastic and viscoelastoplastic responses are considerably smaller than elastic and elastoplastic behavior. Once again, effects of plasticity result in larger deformations for both viscous and nonviscous cases.

The vector representations of elastoplastic deformations for the static analysis are shown in Figures 6 and 7. Deformations for 41.4% slip are larger than these for 3.1% slip. For the case of dynamic analysis (41.4% slip) the curvilinear transient deformation vectors for viscoelastoplastic response are shown in Figure 8. These vectors represent the time history from $t = 0$ to $t = 0.6$ sec. No doubt that the effects of inertia under dynamic loads caused larger deformations than under static loads but energy dissipation through the viscous behavior retarded the motion considerably in comparison with the nonviscous cases as noted in Figure 5. Deformed shapes for dynamic viscoelastoplastic responses at $t=0.3$ sec. are shown in Figure 9.

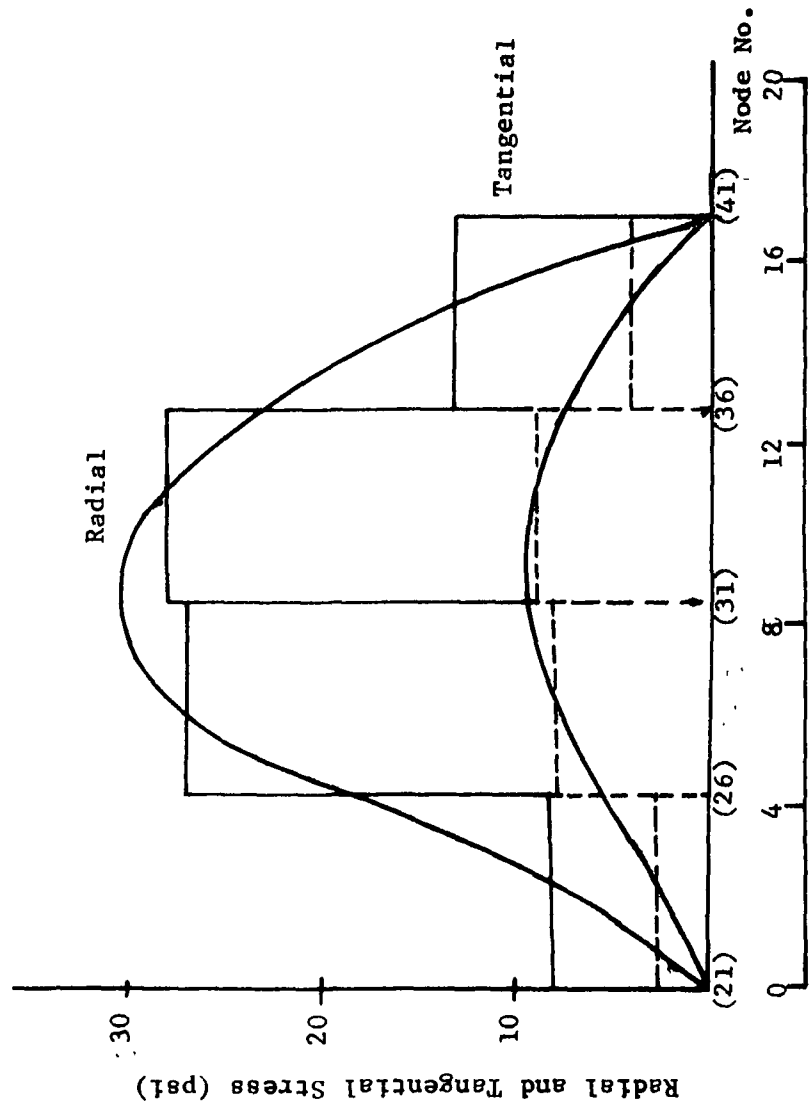


Figure 2: - Radial and Tangential Stress Distribution at the Interface for 3.1% Slip on Compact Sand. Ref. [6]

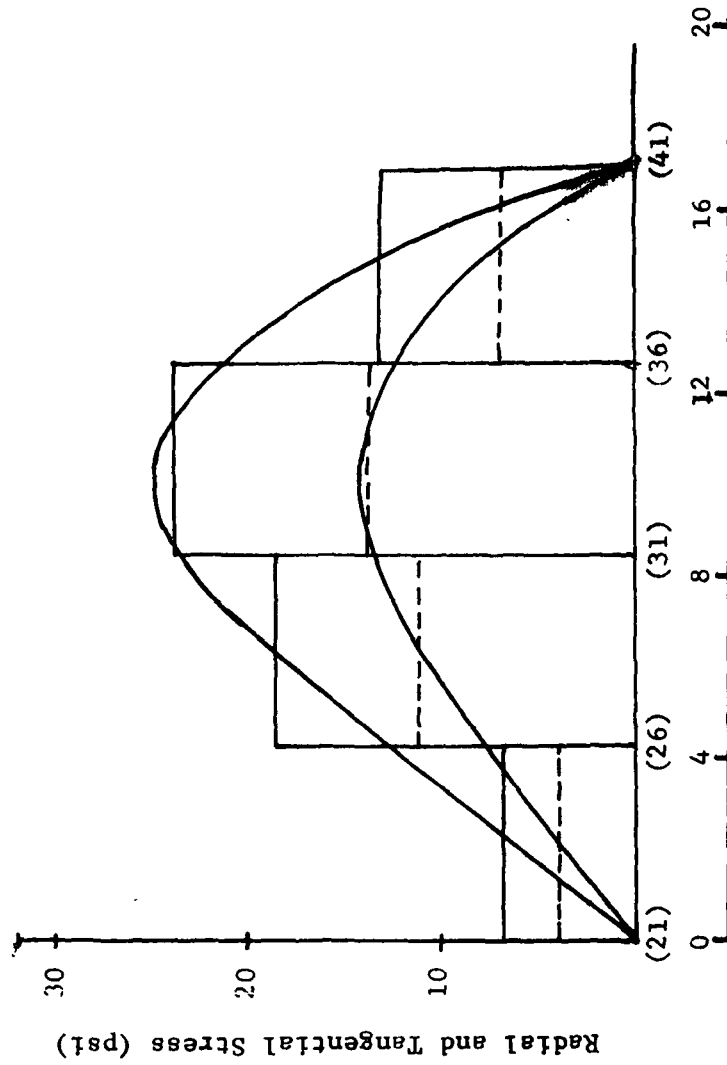
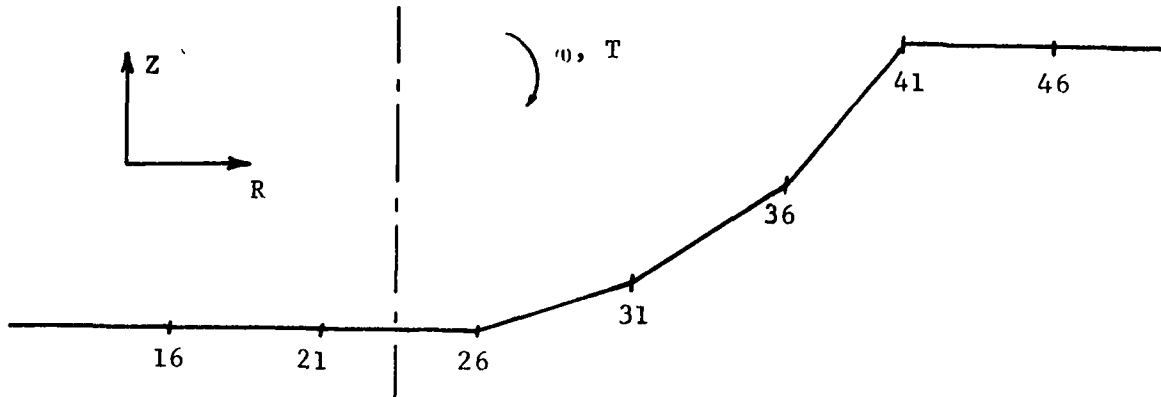


Figure 3: Radial and Tangential Stress Distribution at the Interface for 41.4% Slip on Compact Sand. Ref. [6]



| NODE NO. | 3.1% SLIP | | 41.4% SLIP | |
|----------|-----------|-----------|------------|-----------|
| | F_z (#) | F_R (#) | F_z (#) | F_R (#) |
| 21 | -16.17 | -6.61 | -13.42 | -9.5 |
| 26 | -74.5 | -14.99 | -55.21 | -26.67 |
| 31 | -119.62 | -4.33 | -98.27 | -27.78 |
| 36 | -88.86 | -10.97 | -89.1 | -8.4 |
| 41 | -27.11 | 8.3 | -32.1 | 2.85 |

Figure 4: Equivalent Nodal Forces as determined from Figures 3 and 4.

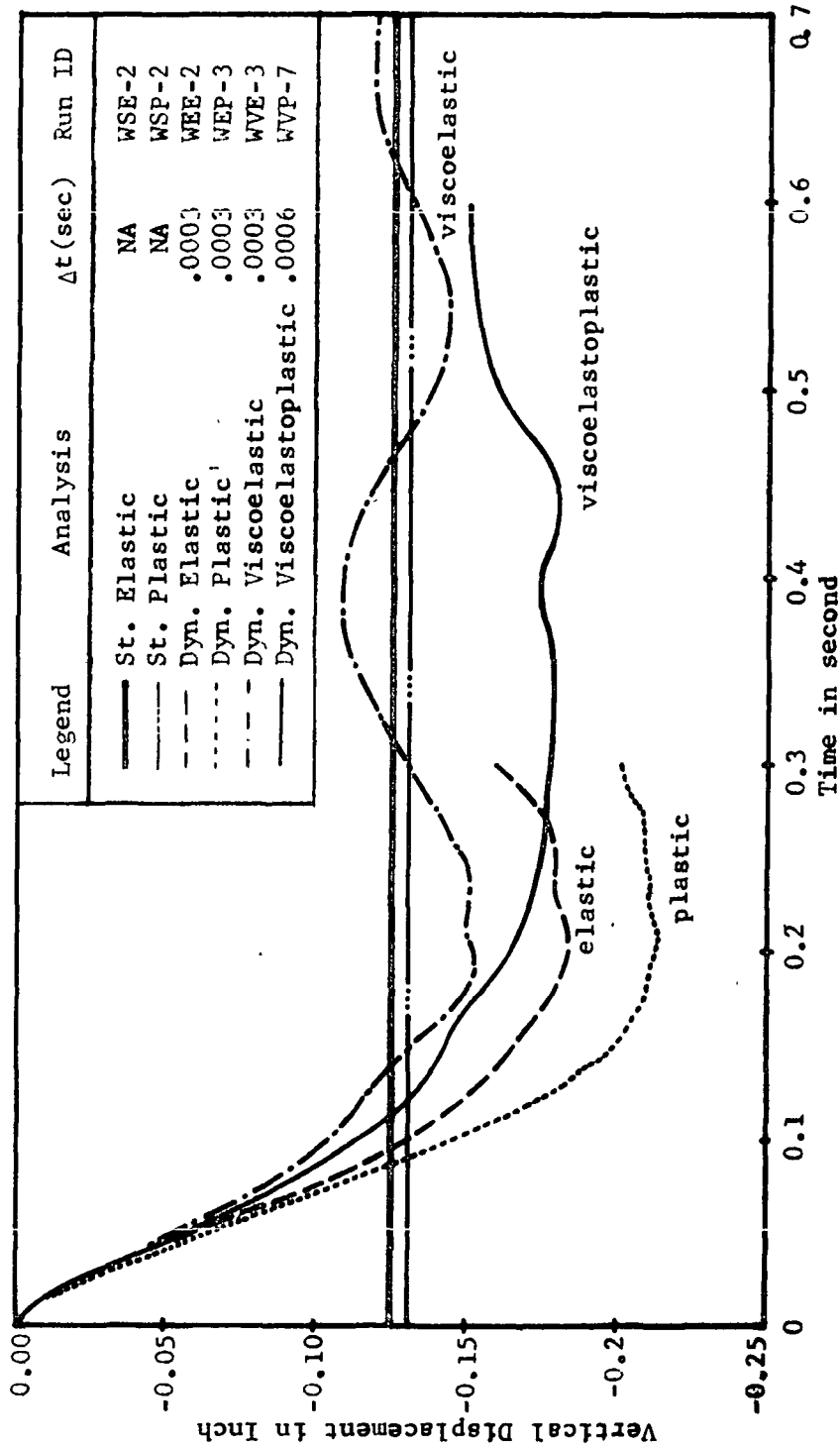


Figure 5: Time-Displacement Curves for 3.1% slip at Node No. 31.

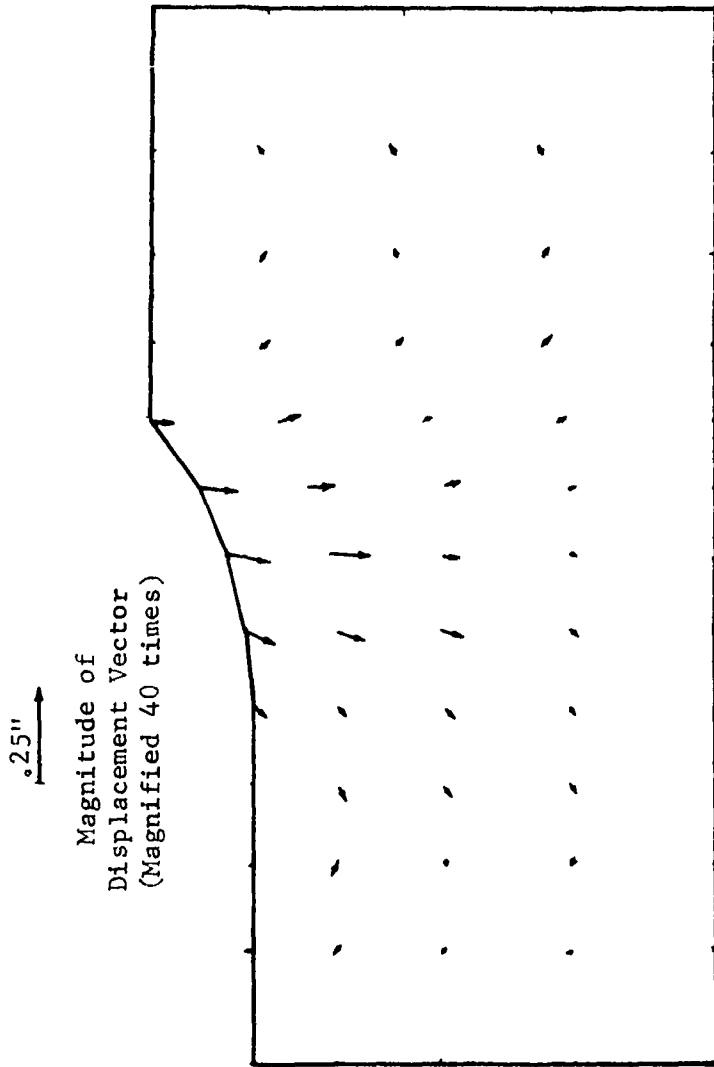


Figure 6: Vector Representation of Displacements (Static Elastoplastic Analysis for 3.1% Slip)

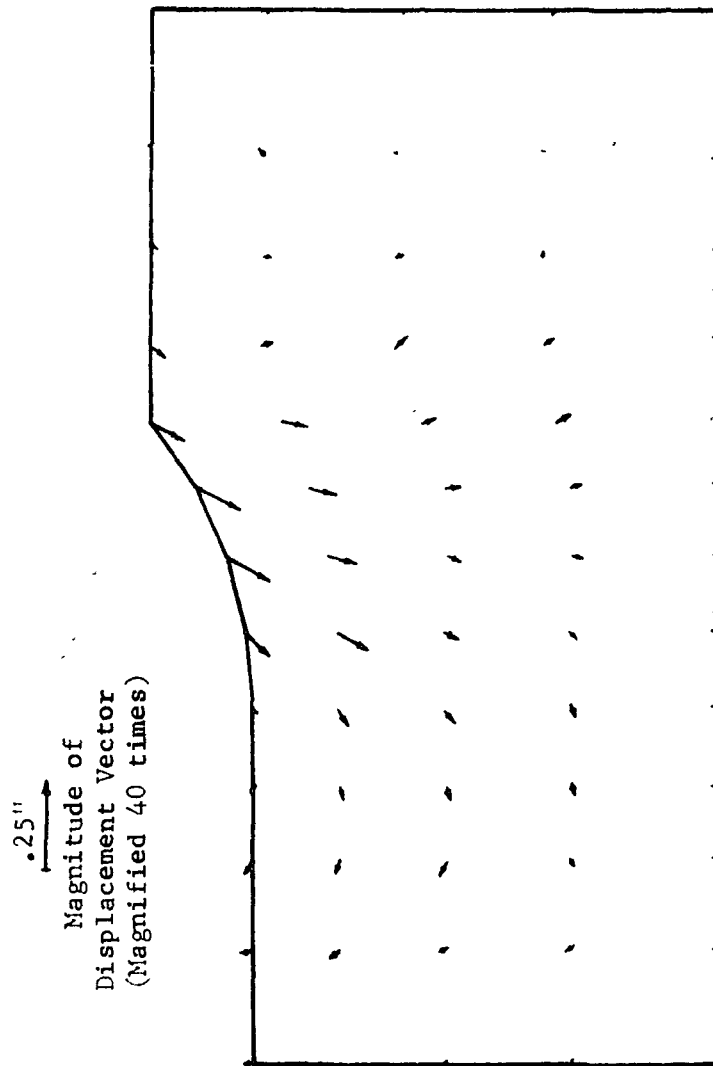


Figure 7: Vector Representation of Displacements (Static Elastoplastic Analysis for 41.4% Slip)

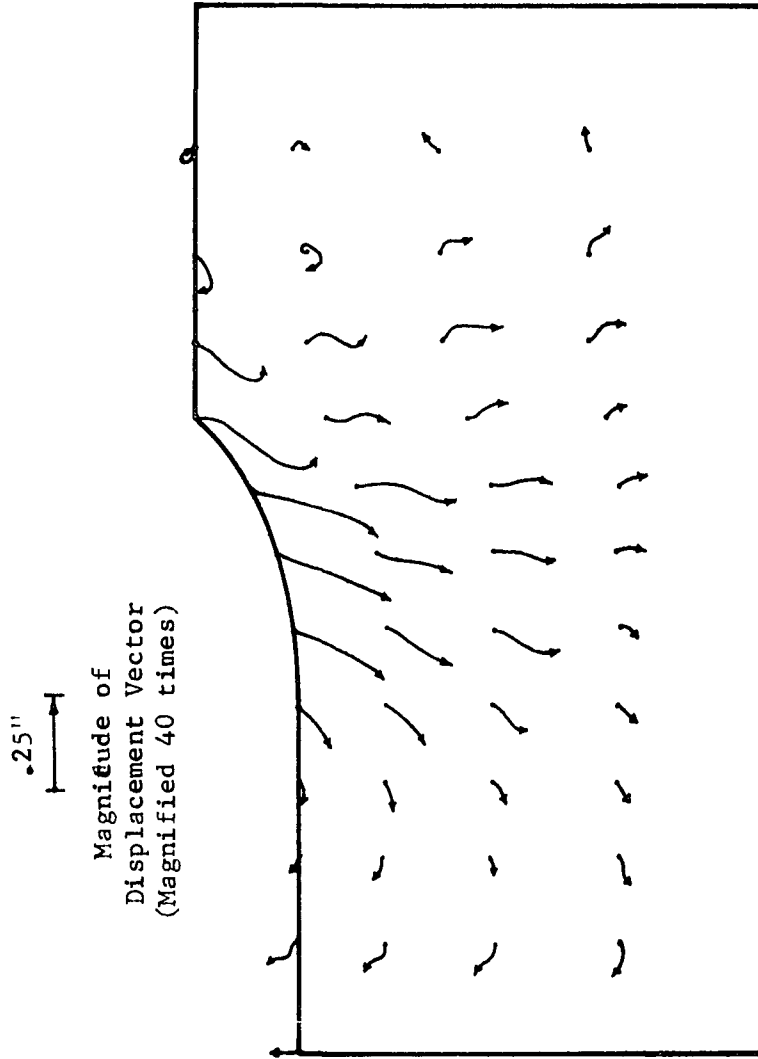


Figure 8: Vector Representation of Deformation (Dynamic Visco-
elastoplastic Analysis for 41.4% Slip)

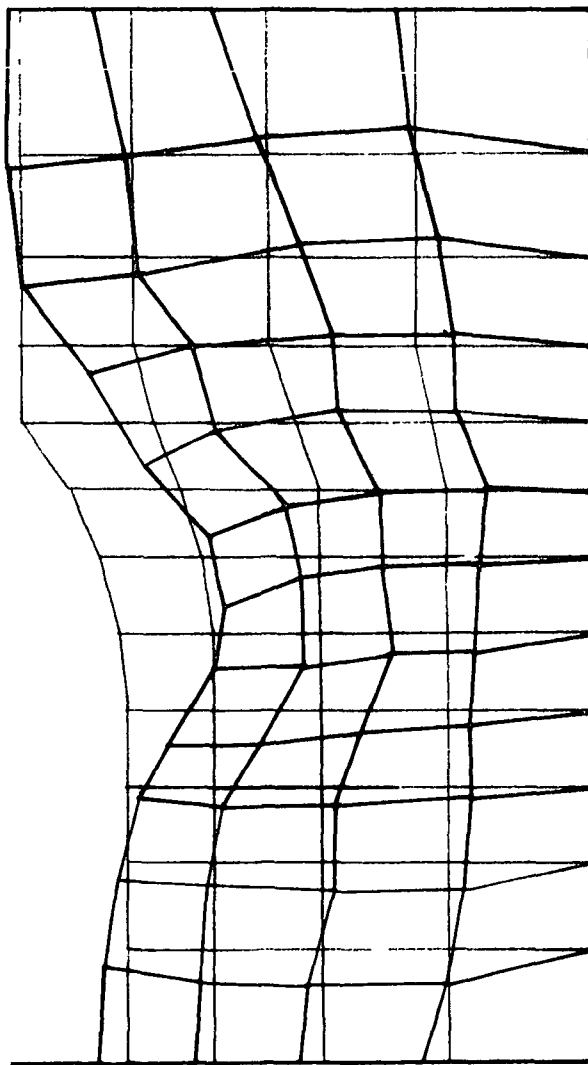


Figure 9: Deformed Configuration at .3 sec,
(Dynamic Viscoplastic Analysis for 41.4% Slip)

From the deformation fields various stress components are calculated and the results shown in the form of isobars in Figures 10 through 21. In the region close to the wheel the major principal stresses due to the elastoplastic deformations are smaller than those of Ref. [12] as shown in Figures 10 and 11 for 3.1% slip and 41.4% slip, respectively. Slightly larger major principal stresses develop at the mid-depth for the 3.1% slip. For the case of maximum shear stresses (Figures 12 and 13) the present analysis gives larger values than Ref. [12] for 3.1% slip, but this trend is reversed for 41.4% slip. In general, the maximum shear stresses for the 3.1% slip are larger than for the 41.4% slip, the same trend as in the case of major principal stresses. Dynamic elastoplastic major principal stresses and maximum shear stresses for 3.1% slip at $t=0.072$ sec., 0.15 sec., 0.228 sec., 0.3 sec., 0.6 sec. are shown in Figures 14 through 17. Variations of stresses with time until maximum stresses are reached are clearly shown. The effects of viscosity or rate-dependent plasticity for 3.1% slip at $t=0.3$ sec. and $t=0.6$ sec. are shown in Figures 18 and 19, respectively. The same information for 41.4% slip is given in Figures 20 and 21. It is seen that as the slip increases the major principal and maximum shear stresses tend to decrease.

II-9. CHARACTERIZATION OF SOIL MECHANICS PARAMETERS

Studies on deformation and stress fields as described in Section II-7 indicate that constitutive relationships for the soil behavior significantly influence the response patterns. The mechanics of wheel-soil

--- Reference 12
— Static Elastoplastic Analysis
(Run ID. MSP - 2)

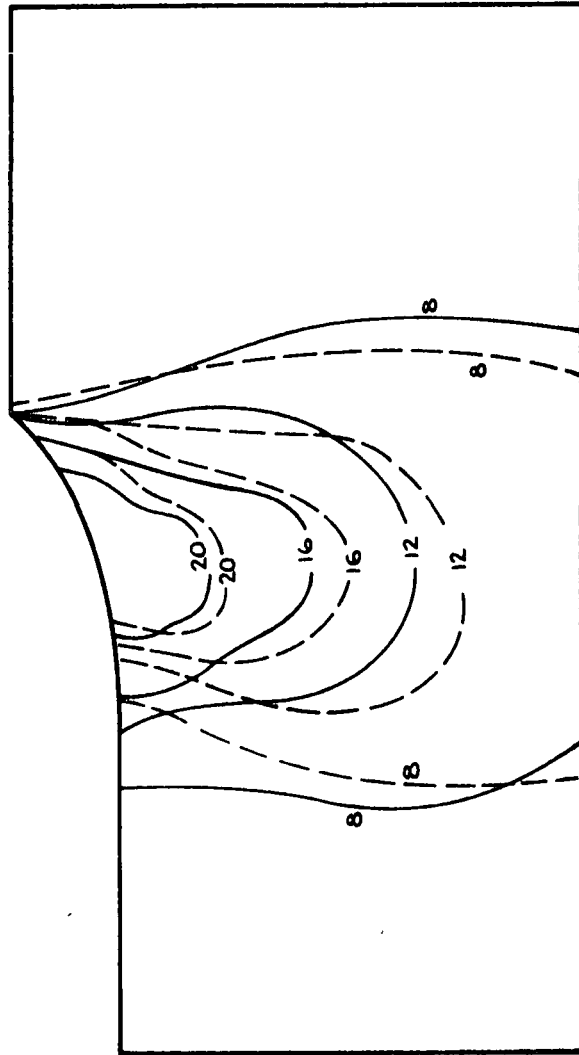


Figure 10: Isobars of Major Principal Stress (in psi) for 3.1% Slip

--- Reference 12

— Static Elastoplastic Analysis
(Run ID: WSP - 3)

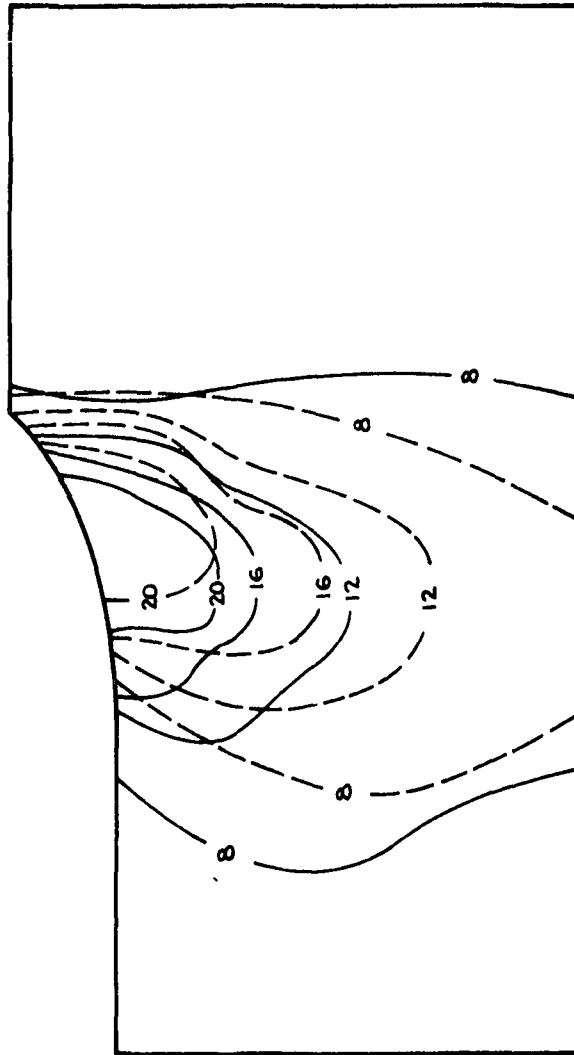


Figure 11: Isobars of Major Principal Stress (in psi)
for 41.4% Slip

--- Reference 12
— Static Plastic Analysis
(Run ID: WSP - 2)

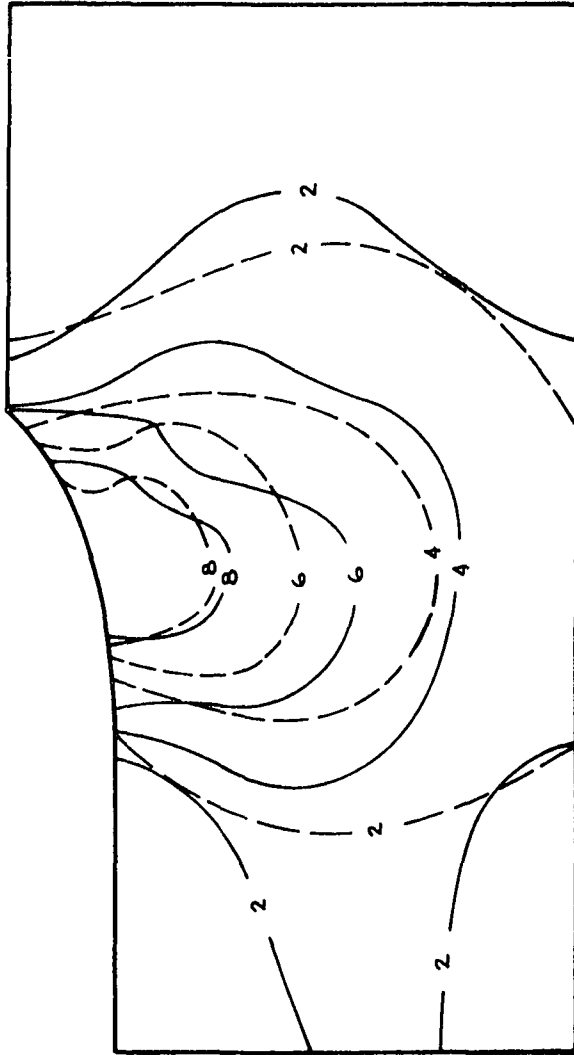


Figure 12: Isobars of Maximum Shear Stress for 3.1% Slip

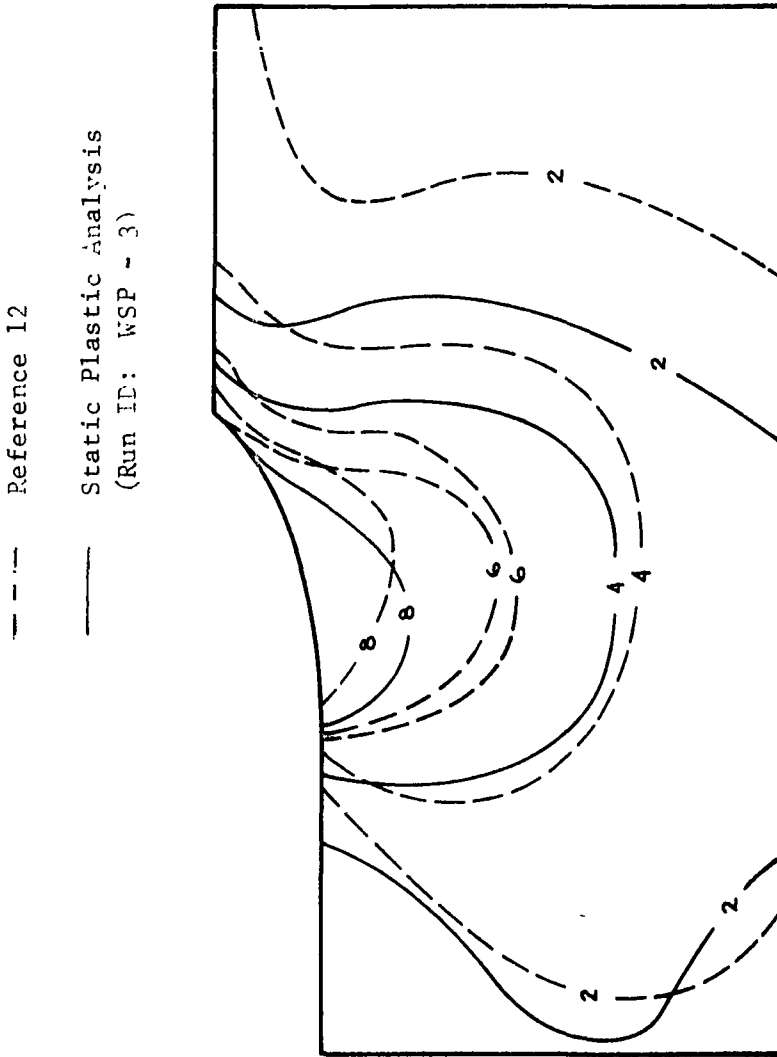


Figure 13: Isobars of Maximum Shear Stress for 41.4% Slip

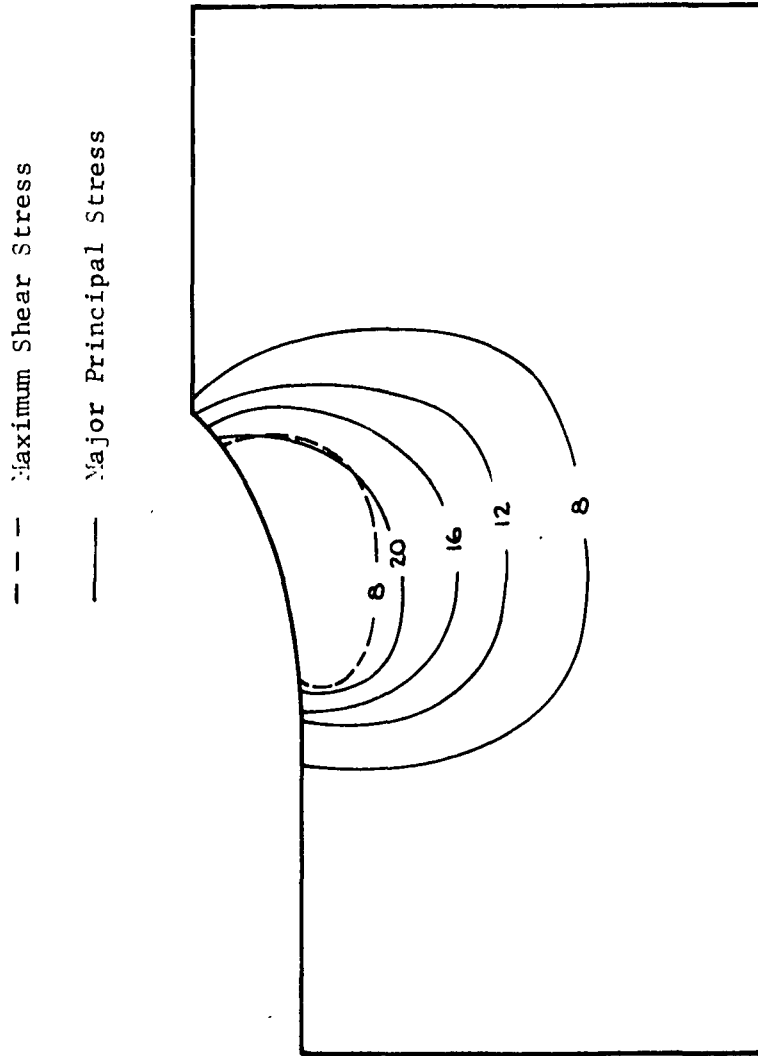


Figure 14: Isobaré of Major Principal Stresses and Maximum Shear Stresses (psi) at $t = 0.072$ sec.

(Dynamic Elasto-plastic Analysis for 3.1% Slip)

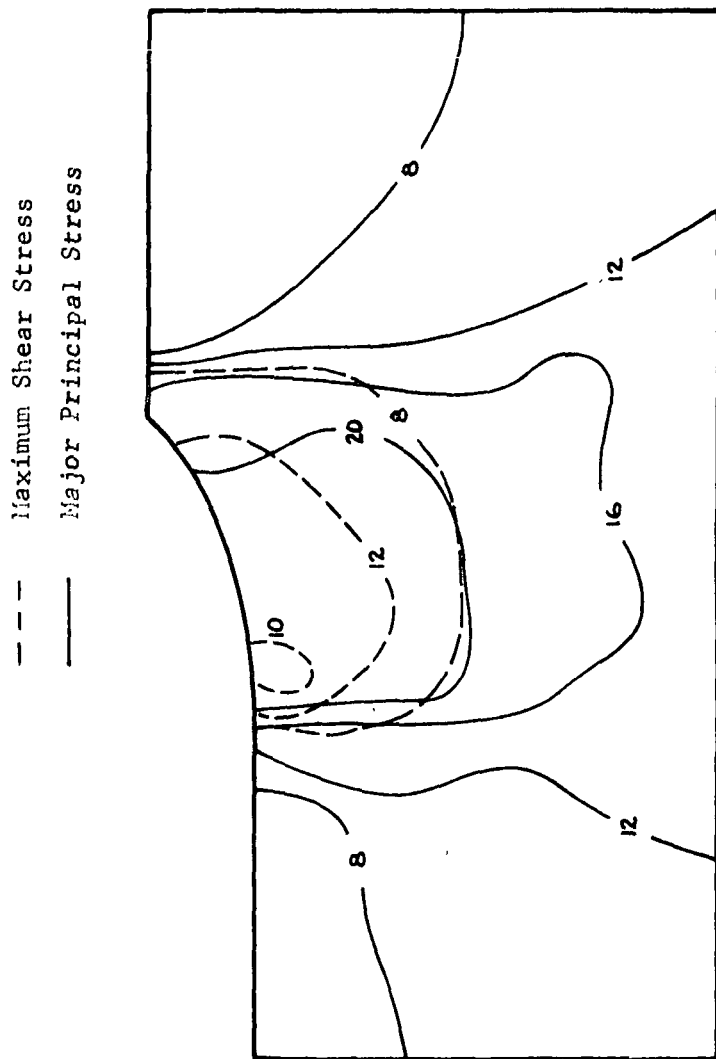


Figure 15: Isobars of Major Principal Stress and Maximum Shear Stress (Psf) at $t=0.15$ sec. (Dynamic Elastoplastic Analysis for 3.1% Slip)

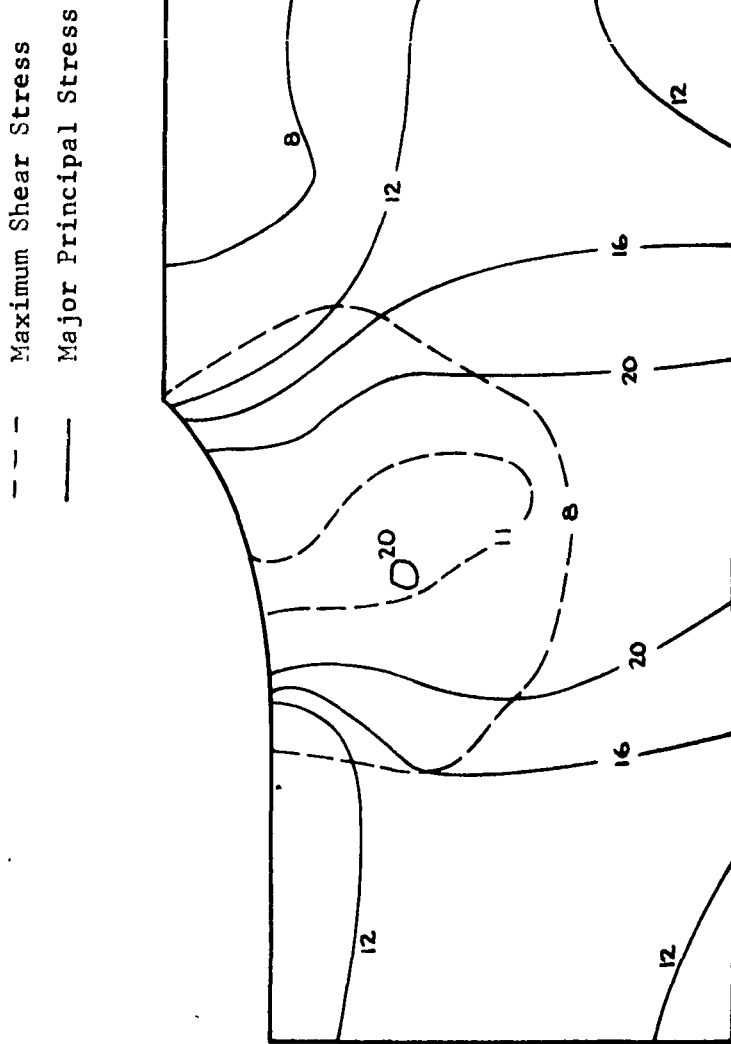


Figure 16: Isobars of Principal Stress and Maximum Shear Stress (Psi) at t=0.228 sec. (Dynamic Elastoplastic Analysis for 3.1% slip)

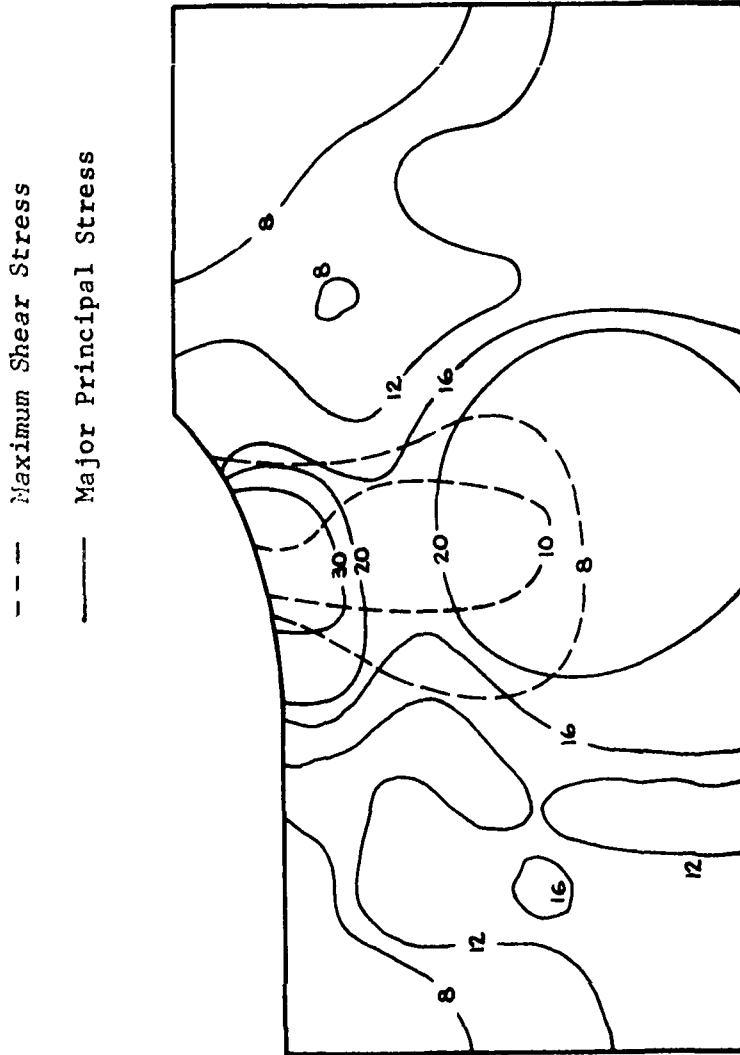


Figure 17: Isobars of Major Principal Stress at Maximum Shear Stress (psi) at $t=0.3$ sec. (Dynamic Elastoplastic Analysis for 3.1% slip)

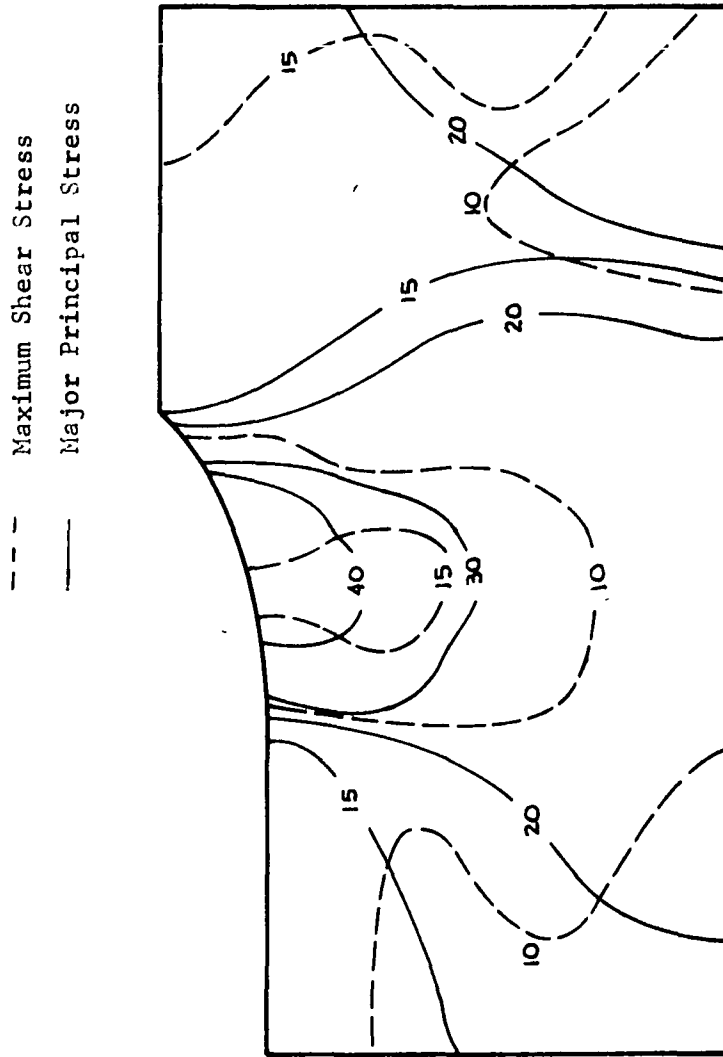


Figure 18: Isobars of Major Principal Stresses and Maximum Shear Stresses at $t=0.3$ sec (Dynamic Viscoelastoplastic Analysis for 3.1% Slip)

--- Maximum Shear Stress
— Major Principal Stress

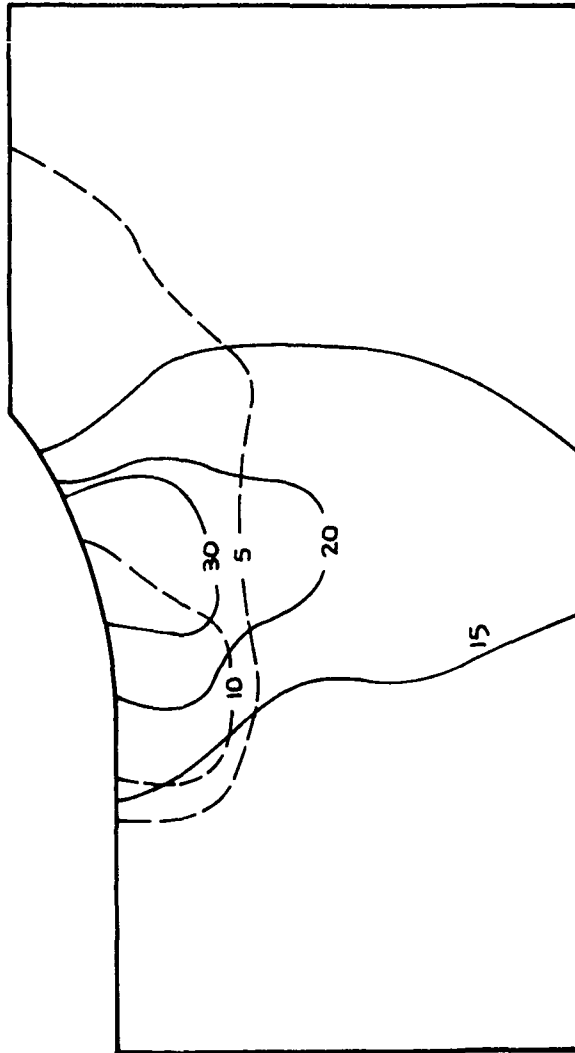


Figure 19: Isobars of Major Principal Stresses and Maximum Shear Stresses (psi) at .6 sec (Dynamic Viscoelastoplastic Analysis for 3.1% Slip)

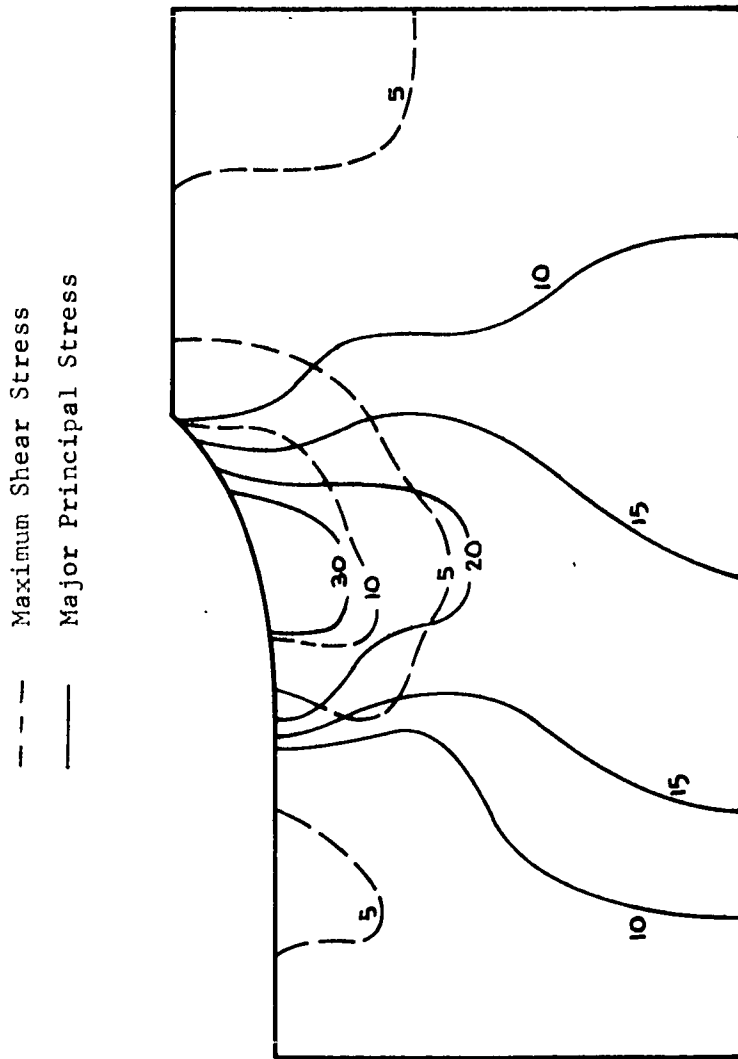


Figure 20: Isobars of Maximum Shear and Major Principal Stresses (psf) at $t=0.3$ sec. (Dynamic Viscoelastoplastic Analysis for 41.4% Slip)

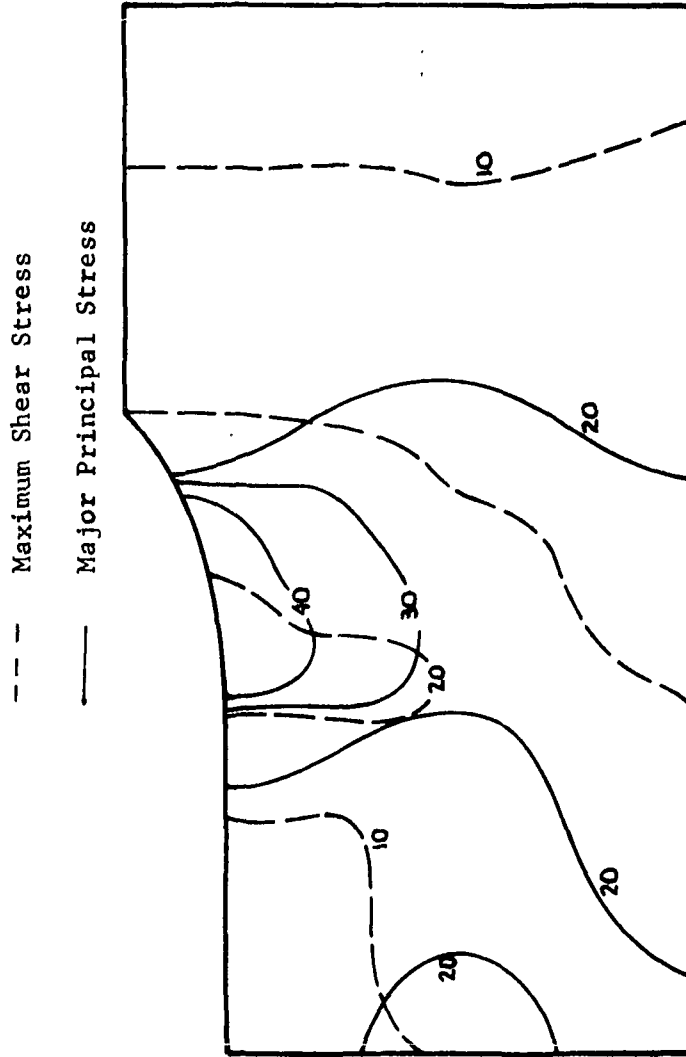


Figure 21: Isobars of Major Principal Stress and Maximum Shear Stress (Psi) at $t=0.65$ sec (Dynamic Viscoelastoplastic Analysis for 41.4% Slip)

interaction cannot be understood properly if incorrect judgement or oversimplification in the theoretical formulation obscures the true deformation and stress fields. For this reason, the present study was devoted to a new approach in which rate-dependent inelastic behavior coupled with effects of inertia was considered.

The analysis presented in the previous sections becomes the stepstone for characterizing the soil mechanics parameters more realistically. Of course, all the results obtained here are based on hypothetical material constants. However, if the analytical formulations are correct, then the wheel-soil interaction data as observed qualitatively and quantitatively may be used to correlate with material constants. Such characterization can be achieved by holding some of the material parameters constant and comparing the load-deformation data between the calculated and observed values.

Because the present study does not include the dust cloud motion behind the lunar rover the observed rooster-tailing cannot be related to the material characterization. However, the sinkage of the rover wheel together with the vehicle performance data can be used for correlation with deformation and stress fields as mentioned in the previous paragraph.

II-10. CONCLUSIONS

The main objective of the present study was to introduce a feasible constitutive relationship for soil deformation and stress fields under a moving wheel. The load transmitted by the moving wheel is dynamic rather

than static. The soil is dissipative media in which inelastic deformation of the soil is governed by the rate-dependent plasticity or viscoelastoplasticity. The yield surface theory of Roscoe and Burland is utilized here for inelastic behavior. The internal variables are then introduced to account for rate-dependent viscous behavior. Effects of soil inertia are included. Combinations of all of these properties result in dynamic analysis of viscoelastoplastic media.

The numerical results obtained here appear very reasonable. Comparisons with the results of other investigators are made and deviations are believed to be due to more rigorous treatment of material behavior considered in the present study. In order to verify the impact of the theoretical formulations given here, however, additional comparison study through experimental data is needed.

REFERENCES

1. Bekker, M. G., Theory of Land Locomotion, University of Michigan Press, Ann Arbor, Michigan, 1956.
 2. Micklethwaite, E. W. E., Soil Mechanics in Relation to Fighting Vehicles, Military Science, Chestsey, 1944.
 3. Evans, I., "The Performance of Rigid Cylindrical Wheels on Clay Soil", Proc. of the First Int. Conf. on Mechanics of Soil-Vehicle System, Turin, 1961.
 4. Uffelmann, F. L., "The Performance of Rigid Cylindrical Wheels on Clay Soil," Proc. of the First Int. Conf. on Mechanics of Soil-Vehicle Systems, Turin, 1961.
 5. Bekker, A. R., Off the Road Locomotion, University of Michigan Press, Ann Arbor, 1960.
 6. Onaffeko, O., and Reece, A. R., "Soil Stresses and Deformations Beneath Rigid Wheels," J. of Terramechanics, Vol. 4, No. 1, 1967, pp. 59-80.
 7. Wong, J. Y. and Reece, A. R., "Prediction of Rigid Wheel Performance Based on the Analysis of Soil-Wheel Stresses, Part I. Performance of Driven Rigid Wheels," J. of Terramechanics, Vol. 4, No. 1, 1967, pp. 81-98.
 8. Wong, J. Y. and Reece, A. R., "Prediction of Rigid Wheel Performance Based on the Analysis of Soil-Wheel Stresses, Part II. Performance of Towed Rigid Wheels", J. of Terramechanics, Vol. 4, No. 2, 1967, pp. 7-25.
 9. Yong, R. N. and Webb, G. L., "Energy Dissipation and Drawbar Pull Prediction in Soil-Wheel Interaction," Proc. of Third Int. Conf. on Terrain-Vehicle Systems, Essen, West Germany, 1969.
 10. Schuring, D., "Mathematical Models of the Rigid Wheel on Soft Soil", Proc. of Third Int. Conf. on Terrain-Vehicle Systems, Essen, West Germany, 1969.
-

11. Windisch, E. J., and Yong, R. N., "The Determination of Soil Strain-Rate Behavior Beneath a Moving Wheel," J. of Terramechanics, Vol. 7, No. 1, 1970, pp. 55-67.
12. Perumpral, J. V., Liljedahl, J. B., and Perloff, W. H., "A Numerical Method for Predicting the Stress Distribution and Soil Deformation Under a Tractor Wheel", J. of Terramechanics, Vol. 8, No. 1, 1971, pp. 9-22.
13. Elsamny, M. K. and Ghobarah, A. A., "Stress Field Under Slipping Rigid Wheel", J. of Soil Mechanics and Foundations Div., ASCE Proc. Paper 8625, January, 1972.
14. Wong, J. Y., Discussions of Paper "Stress Field Under Slipping Rigid Wheel" by Elsamny, M. K. and Ghobarah, A. A., J. of Soil Mechanics and Foundations Div., ASCE, Vol. 98, No. SM 9, September, 1972.
15. Kloc, I., "Mechanical Interaction of a Driven Roller (Wheel) on Soil Slopes", Part I, Technical Memorandum 33-477, Jet Propulsion Lab, Pasadena, Calif., 1971.
16. Coleman, B. D. and Gurtin, M. E., "Thermodynamics with Internal State Variables", Journal of Chemistry and Physics, Vol. 47, 1967, pp. 597-613.
17. Perzyna, P. and Wojno, W., "Thermodynamics of a Rate Sensitive Plastic Material", Archive Mechaniki Stosowanej, Vol. 20, 1968, pp. 501-510.
18. Schofield, A. N. and Wroth, C. P., Critical State Soil Mechanics, McGraw-Hill Book Co., New York, 1968.
19. Roscoe, K. H. and Burland, J. B., "On the Generalized Stress-Strain Behavior of Wet Clay," Symposium on Engineering Plasticity, Cambridge University Press, 1968, pp. 535-609.
20. Zienkiewicz, O. C., Finite Element Method in Engineering Science, McGraw-Hill Book Co., New York, 1972.

21. Oden, J. T., Finite Elements in Nonlinear Continua, McGraw-Hill Book Co., New York, 1972.
 22. Chung, T. J., and Lee, J. K., "Mathematical Characterization of Mechanical Behavior of Porous Frictional Granular Media", Final Technical Report, Contract NAS 8-25102, January, 1971.
 23. Chung, T. J., Costes, N. C., and Lee, J. K., "Boundary Value Problems with Incremental Plasticity in Granular Media", Discussion presented at International Symposium on Foundations of Plasticity, Warsaw, Poland, August 29-September 1, 1971; also submitted to Archives of Mechanics.
 24. Chung, T. J., and Lee, J. K., "Incremental Plasticity Theory Applied to Boundary Value Problems in Soil", Proc. WES Symposium on Finite Element Applications to Geotechnical Engineering, April 1-3, 1972.
-

APPENDIX 1

DERIVATION OF INTERNAL (HIDDEN) VARIABLES

Consider the internal variable $\alpha_{1j}^{(r)}$

$$\alpha_{1j}^{(r)}(t) = \int_0^t \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{1j}(\tau) d\tau \quad (\text{A-1})$$

where $\dot{\gamma}_{1j}(\tau)$ may be considered to vary linearly within the small time interval Δt ,

$$\dot{\gamma}_{1j}(\tau) = \dot{\gamma}_{1j}(t-\Delta t) + \frac{\tau-(t-\Delta t)}{\Delta t} [\dot{\gamma}_{1j}(t) - \dot{\gamma}_{1j}(t-\Delta t)] \quad (\text{A-2})$$

Substituting (A-2) in (A-1),

$$\begin{aligned} \alpha_{1j}^{(r)}(t) &= \int_0^{t-\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{1j}(\tau) d\tau + \int_{t-\Delta t}^t \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{1j}(\tau) d\tau \\ &= \exp\left(\frac{-\Delta t}{T_{(r)}}\right) \alpha_{1j}^{(r)}(t-\Delta t) + \int_{t-\Delta t}^t \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{1j}(\tau) d\tau \\ &= \exp\left(\frac{-\Delta t}{T_{(r)}}\right) \alpha_{1j}^{(r)}(t-\Delta t) + \int_{t-\Delta t}^t \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \left\{ \dot{\gamma}_{1j}(t-\Delta t) \right. \\ &\quad \left. + \frac{\Delta t-t+\tau}{\Delta t} [\dot{\gamma}_{1j}(t) - \dot{\gamma}_{1j}(t-\Delta t)] \right\} d\tau \end{aligned}$$

$$\begin{aligned}
 &= \exp\left(\frac{-\Delta t}{T(r)}\right) \alpha_{ij}^{(r)}(t-\Delta t) + \int_{t-\Delta t}^t \exp\left(\frac{-(t-\tau)}{T(r)}\right) \left\{ \left(1 - \frac{t}{\Delta t} + \frac{\tau}{\Delta t}\right) \dot{\gamma}_{ij}(\tau) \right. \\
 &+ \left. \left(\frac{t}{\Delta t} - \frac{\tau}{\Delta t}\right) \dot{\gamma}_{ij}(t-\Delta t) \right\} d\tau = \exp\left(\frac{-\Delta t}{T(r)}\right) \alpha_{ij}^{(r)}(t-\Delta t) + T(r) \left\{ \left[\exp\left(\frac{-(t-\tau)}{T(r)}\right) \right. \right. \\
 &- \left. \frac{t}{\Delta t} \exp\left(\frac{-(t-\tau)}{T(r)}\right) + \frac{\tau}{\Delta t} \exp\left(\frac{-(t-\tau)}{T(r)}\right) - \frac{T(r)}{\Delta t} \exp\left(\frac{-(t-\tau)}{T(r)}\right) \right]_{t-\Delta t}^t \dot{\gamma}_{ij}(\tau) \right. \\
 &+ \left. \left[\frac{t}{\Delta t} \exp\left(\frac{-(t-\tau)}{T(r)}\right) - \frac{\tau}{\Delta t} \exp\left(\frac{-(t-\tau)}{T(r)}\right) + \frac{T(r)}{\Delta t} \exp\left(\frac{-(t-\tau)}{T(r)}\right) \right]_{t-\Delta t}^t \dot{\gamma}_{ij}(t-\Delta t) \right\} \\
 &= \exp\left(\frac{-\Delta t}{T(r)}\right) \alpha_{ij}^{(r)}(t-\Delta t) + T(r) \left[\left\{ 1 - \exp\left(\frac{-\Delta t}{T(r)}\right) - \frac{t}{\Delta t} \left[1 - \exp\left(\frac{-\Delta t}{T(r)}\right) \right] \right. \right. \\
 &+ \left. \left. \frac{t}{\Delta t} - \frac{t-\Delta t}{\Delta t} \exp\left(\frac{-\Delta t}{T(r)}\right) - \frac{T(r)}{\Delta t} + \frac{T(r)}{\Delta t} \exp\left(\frac{-\Delta t}{T(r)}\right) \right\} \dot{\gamma}_{ij}(t) \right. \\
 &+ \left. \left\{ \frac{t}{\Delta t} \left[1 - \exp\left(\frac{-\Delta t}{T(r)}\right) \right] - \frac{t}{\Delta t} + \frac{t-\Delta t}{\Delta t} \exp\left(\frac{-\Delta t}{T(r)}\right) + \frac{T(r)}{\Delta t} \right. \right. \\
 &- \left. \left. \frac{T(r)}{\Delta t} \exp\left(\frac{-\Delta t}{T(r)}\right) \right\} \dot{\gamma}_{ij}(t-\Delta t) \right] = \exp\left(\frac{-\Delta t}{T(r)}\right) \alpha_{ij}^{(r)}(t-\Delta t) \\
 &+ T(r) \left[\left\{ - \exp\left(\frac{-\Delta t}{T(r)}\right) + \frac{T(r)}{\Delta t} \left[1 - \exp\left(\frac{-\Delta t}{T(r)}\right) \right] \right\} \dot{\gamma}_{ij}(t-\Delta t) \right. \\
 &+ \left. \left\{ 1 - \frac{T(r)}{\Delta t} \left[1 - \exp\left(\frac{-\Delta t}{T(r)}\right) \right] \right\} \dot{\gamma}_{ij}(t) \right] = \overset{(r)}{A} \alpha_{ij}^{(r)}(t-\Delta t) \\
 &+ \overset{(r)}{B} \dot{\gamma}_{ij}(t-\Delta t) + \overset{(r)}{C} \dot{\gamma}_{ij}(t)
 \end{aligned}$$

or

$$\alpha_{ij}^{(r)}(s) = A^{(r)} \alpha_{ij}^{(r)}(s-1) + B^{(r)} \dot{y}_{ij}^{(r)}(s-1) + C^{(r)} \dot{y}_{ij}^{(r)}(s)$$

where

$$A^{(r)} = \exp \frac{-\Delta t}{T_{(r)}}$$

$$B^{(r)} = T_{(r)} \left[\frac{(r)}{\phi} - A^{(r)} \right]$$

$$C^{(r)} = T_{(r)} \left[1 - \frac{(r)}{\phi} \right]$$

$$\frac{(r)}{\phi} = \frac{T_{(r)}}{\Delta t} (1 - A^{(r)})$$

APPENDIX 2

CONTACT STRESSES AT WHEEL-SOIL INTERFACE

The vertical and horizontal forces and torque of a wheel rotating on horizontal ground with constant velocity are given by

$$W = rb \left\{ \int_{\theta_2}^{\theta_1} \sigma(\theta) \cos \theta d\theta + \int_{\theta_2}^{\theta_1} \tau(\theta) \sin \theta d\theta \right\}$$

$$D = rb \left\{ \int_{\theta_2}^{\theta_1} \tau(\theta) \cos \theta d\theta - \int_{\theta_2}^{\theta_1} \sigma(\theta) \sin \theta d\theta \right\}$$

$$T = r^2 b \int_{\theta_2}^{\theta_1} \tau(\theta) d\theta$$

in which $\sigma(\theta)$ and $\tau(\theta)$ are the average radial and tangential stress across the wheel width of b (Fig. 2-1).

The location of the point of the maximum radial stress may be expressed as

$$\theta_M = (C_1 + C_{21})\theta_1$$

where s is the slip (%) defined by

$$s = \left(1 - \frac{V}{\omega r}\right) 100$$

and C_1 and C_2 are the constants [14-17] given in Table 1.

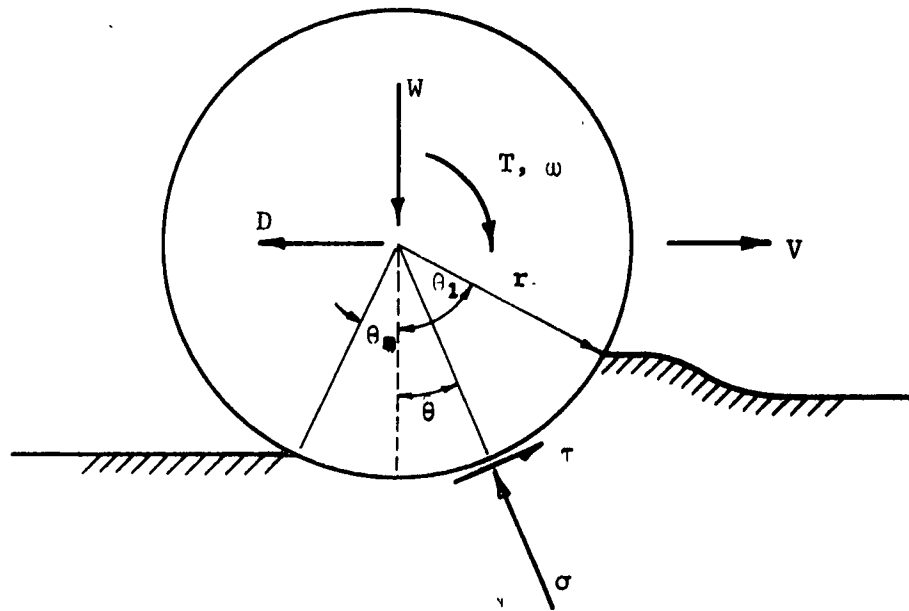


Figure 2-1: Equilibrium of a Driven Rigid Wheel on Soil

TABLE 1

| Soil | Angle of Internal Friction (deg.) | Soil Cohesion | Density (lb/in ³) | C ₁ | C ₂ | K ₁ | K ₂ | n | Shear Deformation Modulus (in) |
|--------------|-----------------------------------|---------------|-------------------------------|----------------|----------------|----------------|----------------|------|--------------------------------|
| Compact sand | 33.3 | 0.10 | 0.0575 | 0.413 | 0.32 | 20 | 2.5 | 0.47 | 1.5 |
| Loose sand | 31.1 | 0.12 | 0.048 | 0.18 | 0.32 | 0 | 2 | 1.15 | 1.5 |
| Sand | 36.0 | 0.10 | 0.0617 | 0.285 | 0.32 | - | - | - | - |
| Dry sand | 24.0 | - | - | 0.38 | 0.41 | - | - | - | - |

In the region between θ_1 and θ_M or the front region, the radial stress is given by [18]

$$\sigma_1(\theta) = (K_1 + K_2 b) \left(\frac{r}{b}\right)^n (\cos \theta - \cos \theta_1)^n$$

where the constants K_1 , K_2 , and n are shown in Table 1. The radial stress acting in the rear region is of the form

$$\sigma_2(\theta) = (K_1 + K_2 b) \left(\frac{r}{b}\right)^n \left[\cos \left\{ \theta_1 - \theta \left(\frac{C_1 + C_{21}}{C_1 + C_{21}} \right) \right\} - \cos \theta_1 \right]^n$$

The shear stress around the rim is given by [14,15],

$$\tau(\theta) = (C + \sigma(\theta) \tan \varphi) (1 - e^\beta)$$

where C is the cohesion, and

$$\beta = \frac{-r}{K} \{ (\theta_1 - \theta) - (1-i)(\sin \theta_1 - \sin \theta) \}$$

In the above expressions θ_1 is still not known but can be determined from the expression of the vertical force W,

$$W = rb \left\{ \int_{\theta_M}^{\theta_1} \sigma_1(\theta) \cos \theta d\theta + \int_0^{\theta_M} \sigma_2(\theta) \cos \theta d\theta \right. \\ \left. + \int_{\theta_M}^{\theta_1} \tau_1(\theta) \sin \theta d\theta + \int_0^{\theta_M} \tau_2(\theta) \sin \theta d\theta \right.$$

where,

$$\tau_1(\theta) = \left(C + \sigma_1(\theta) \tan \phi \right) \left(1 - e^{\beta} \right)$$

$$\tau_2(\theta) = \left(C + \sigma_2(\theta) \tan \phi \right) \left(1 - e^{\beta} \right)$$

If the magnitude of W is given then the above integration may be carried out by the Simpson's rule and θ_1 is solved in terms of known values.

With the value of θ_1 known, we can then calculate the radial and tangential stresses.

Finally, the wheel sinkage z_0 is determined from

$$z_0 = (1 - \cos \theta_1) r$$

APPENDIX 3

COMPUTER PROGRAM LISTING

(Dynamic Wheel-Soil Interaction, Plane Strain)


```

000          CALL STRAIN(1)                                00010900
000          STOP ELASTC                                   00010900
000          C                                             00011000
000          C                                             00011100
000          930 CONTINUE                                  00011200
000          C                                             00011300
000          CALL STEP                                     00011400
000          C                                             00011500
000          C                                             00011600
000          1000 STOP                                     00011700
000          C                                             00011800
000          500 FORMAT(10I5)                              00011900
000          510 FORMAT(20A4)                              00012000
000          520 FORMAT(5X,2F10.4,2I5)                    00012100
000          530 FORMAT(6F10.0)                           00012200
000          540 FORMAT(15,2F10.4)                        00012300
000          550 FORMAT(5X,4I5)                            00012400
000          600 FORMAT(1H1,20X,20A4//30X,'COORDINATE VALUES'//T11,'NODE',I30,'7-COORD'//
000          *CORD',I50,'K=COORD',I65,'0,IF FREE TO Z',3X,'0,IF FREE TO R'//) 00012500
000          610 FORMAT(/5X,'E', AND, DENS, AT, XK, DELT,' ,6E15.7//) 00012700
000          620 FORMAT(1I0,I5,I25,F10.4,I45,F10.4,I65,2(1R,5X)) 00012800
000          630 FORMAT(/5X,'INODE', NLELM, NAPL, NRC,' ,4I5) 00012900
000          640 FORMAT(5X,'PHI', VOIDI, CAPA, RAMDA,' ,4F15.6) 00013000
000          641 FORMAT(/5X,'IMAX =',I5,' IHD =',I5,' LT =',I5,' LAST =',I5) 00013100
000          650 FORMAT(/5X,'CONNECTIVITY'//)              00013200
000          660 FORMAT(5I8)                               00013300
000          677 FORMAT(///10X,'APPLIED (, LOAD'//5X,'NODE',I0X,'FORCE TO Z',5X,'
000          * FORCE TO R'//)                               00013400
000          688 FORMAT(5X,I5,2(5X,F12.4))                 00013500
000          689 FORMAT(1H1,10X,' TOTAL DISPLACEMENT      NO. OF INCREMENTAL STEPS
000          * =',2I5//5X,'NODE',5X,'Z - DISPL',20X,'R - DISPL'//) 00013700
000          690 FORMAT(5X,I5,2E15.7)                     00013800
000          691 FORMAT(///10X,' TOTAL STRESSES'//5X,'ELEM',5X,'SIGMA - Z',10X,
000          *SIGMA - R',10X,'TANGENTIAL',10X,'TAU - ZR'//) 00014000
000          692 FORMAT(5X,I5,3X,4F17.6)                  00014200
000          693 FORMAT(3E15.6)                            00014300
000          END                                             00014400
000          INFL,SIH NASA*IPPS,INIT,,,16J532132410
000          SUBROUTINE INIT                                00000100
000          PARAMETER NF1=150,NF2=150,MX=5000,NF=NF1*2    00000200
000          COMMON /BLK4/ D(3,3),ATD(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8), 00000300
000          * VE(8,8),A1(NELS)                             00000400
000          COMMON /BLK5/ GP(NELS,3),STPAIR(NELS,3),ALPHA,BETA,GAMMA,DELT,XK, 00000500
000          * AT,XMU,E,SM,SMS,VOIDI,CAPA,RAMD,BET         00000600
000          DIMENSION GD(3,3),DV(3,3),ATA(3,3)           00000700
000          CONST = E*XMU / ((1.+XMU)*(1.-XMU*2.))        00000800
000          SHEAR = E / (2.*(1.+XMU))                     00000900
000          D(1,1) = CONST + SHEAR*2.                     00001000
000          D(2,2) = D(1,1)                               00001100
000          D(1,2) = CONST                                 00001200
000          D(2,1) = CONST                                 00001300
000          D(3,3) = SHEAR                                 00001400
000          TI = AT/XK                                     00001500
000          LSUM = 3                                       00001600
000          CALL ZERO(ATA,3,3)                             00001700
000          CALL ZERO(DV,3,3)                             00001800
000          LU 5U 1 = 1,3                                  00001900

```

```

000      50  DV(I,J) = XK
000      ALPHA=0.0
000      BETA =0.0
000      GAMMA=0.0
000      IF (ALC(11).LT.0.000001) GO TO 20
000      DT= -DEL1/IJ
000      ALPHA= EXP(DT)
000      BETA= T1*(-ALPHA-(1.-ALPHA)/DT)
000      GAMMA= T1*(1.+(1.-ALPHA)/DT)
000      20  CONTINUE
000      DO 6 I=1,3
000      DO 6 J=1,3
000      AD(I,J)= 0.
000      HD(I,J)=0.
000      GD(I,J)=0.
000      DO 7K=1,LSUM
000      AD(I,J)= AD(I,J) +DV (I,J)*ALPHA
000      HD(I,J)= HD(I,J)+DV (I,J)*BETA
000      7  GD(I,J)=GD(I,J)+DV (I,J)*GAMMA
000      DO 8 L=1,3
000      8  A7D(L,L)=A7A(L,L)+GD(L,L)
000      6  CONTINUE
000      RETURN
000      END
000  @FL1,SIF NASA*IPFS.SIF#1,,165535132410
000  SUBROUTINE STIFF1(DEFNS)
000  PARAMETER NF=150,NELS=150,MX= 500,NF=NFT*2
000  COMMON /BLK0/ TITL(20),INODE,NFLEM,NAPC,NRC,MAC1,ISW(5)
000  COMMON /BLK1/ W(4),P(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000  * FN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEU(4,4),
000  * AU,B(0),CU,IC,JC,KC,LC,NEL
000  COMMON /BLK2/ ID(INF,2),IJKL(NELS,4),R(NF),Z(NFT),KK(NELS,8)
000  COMMON /BLK3/ XPL(MX),P(NF),IMAX,IHR,IHR1,LT,LAST,NFREE
000  COMMON /BLK4/ D(3,3),AD(3,3),AP(3,3),HD(3,3),STIFF(8,8),CM(8,8),
000  * VE(8,8),AI(NELS)
000  COMMON /BLK6/ BT(NELS,8,3),ARM(NELS,4),AZM(NELS,4),AOJ(NELS)
000  COMMON /BLK8/ UP(NELS,3,3),PRINS(NELS,3)
000  COMMON /BLK9/M,N
000  COMMON /DYN/ CBAR(MX),XM(MX),CC(MX)
000  C
000  C
000  DO 900 NEL = 1,NELLM
000  C
000      IC = IJKL(NEL,1)
000      JC = IJKL(NEL,2)
000      KC = IJKL(NEL,3)
000      LC = IJKL(NEL,4)
000  C
000      CALL AREA(A,NEL,AREA)
000      AI(NEL) = AREA
000  C
000      AZ(1) = Z(LC) - Z(JC)
000      AZ(2) = Z(IC) - Z(KC)
000      AZ(3) = -AZ(1)
000      AZ(4) = -AZ(2)
000      FZ(1) = Z(KC) - Z(LC)
000      BZ(2) = -BZ(1)

```

| | | |
|-----|----------------------------------|----------|
| 000 | PZ(3) = Z(JC) - Z(IC) | 00003300 |
| 000 | PZ(4) = -PZ(3) | 00003400 |
| 000 | CZ(1) = Z(JC) - Z(KC) | 00003500 |
| 000 | CZ(2) = Z(LC) - Z(IC) | 00003600 |
| 000 | CZ(3) = -CZ(2) | 00003700 |
| 000 | CZ(4) = -CZ(1) | 00003800 |
| 000 | AK(1) = K(JC) - K(LC) | 00003900 |
| 000 | AK(2) = K(KC) - K(IC) | 00004000 |
| 000 | AK(3) = -AK(1) | 00004100 |
| 000 | AK(4) = -AK(2) | 00004200 |
| 000 | FK(1) = K(LC) - K(KC) | 00004300 |
| 000 | FK(2) = -FK(1) | 00004400 |
| 000 | FK(3) = K(IC) - K(JC) | 00004500 |
| 000 | FK(4) = -FK(3) | 00004600 |
| 000 | CK(1) = K(KC) - K(JC) | 00004700 |
| 000 | CK(2) = K(IC) - K(LC) | 00004800 |
| 000 | CK(3) = -CK(2) | 00004900 |
| 000 | CK(4) = -CK(1) | 00005000 |
| 000 | PO 100 I = 1,4 | 00005100 |
| 000 | AZ(I) = -AZ(1) | 00005200 |
| 000 | PZ(I) = -PZ(1) | 00005300 |
| 000 | CZ(I) = -CZ(1) | 00005400 |
| 000 | FK(I) = -FK(1) | 00005500 |
| 000 | CK(I) = -CK(1) | 00005600 |
| 000 | AK(I) = -AK(1) | 00005700 |
| 000 | AFM(NEL,1) = AK(1) | 00005800 |
| 000 | AZM(NEL,1) = AZ(1) | 00005900 |
| 000 | 100 CONTINUE | 00006000 |
| 000 | AO = AK(3)*AZ(2) - AK(4)*AZ(1) | 00006100 |
| 000 | BO = FK(2)*PZ(4) - FK(3)*PZ(1) | 00006200 |
| 000 | CO = (K(3)*CZ(1) - CK(4)*CZ(2) | 00006300 |
| 000 | AOJ(NEL) = AO | 00006400 |
| 000 | L | 00006500 |
| 000 | EO 200 M = 1,4 | 00006600 |
| 000 | FO 100 N = 1,4 | 00006700 |
| 000 | CALL GAUSS(1,AA) | 00006800 |
| 000 | TYPEA(M,N) = AA | 00006900 |
| 000 | CALL GAUSS(2,AA) | 00007000 |
| 000 | TYPEB(M,N) = AA | 00007100 |
| 000 | CALL GAUSS(3,AA) | 00007200 |
| 000 | TYPEC(M,N) = AA | 00007300 |
| 000 | CALL GAUSS(4,AA) | 00007400 |
| 000 | TYPEU(M,N) = AA | 00007500 |
| 000 | 190 CONTINUE | 00007600 |
| 000 | CALL GAUSS(5,AA) | 00007700 |
| 000 | ET(NEL,M,1) = AA | 00007800 |
| 000 | ET(NEL,M+4,3) = AA / 2. | 00007900 |
| 000 | CALL GAUSS(6,AA) | 00008000 |
| 000 | ET(NEL,M+4,2) = AA | 00008100 |
| 000 | ET(NEL,M,3) = AA / 2. | 00008200 |
| 000 | 200 CONTINUE | 00008300 |
| 000 | L | 00008400 |
| 000 | WRITE(2) TYPEA,TYPEB,TYPEC,TYPEU | 00008500 |
| 000 | L | 00008600 |
| 000 | L1 = L(1,1) | 00008700 |
| 000 | EO 300 I = 1,4 | 00008800 |
| 000 | EO 300 J = 1,4 | 00008900 |

```

000 C
000 STIFF(I,J) = DP(NEL,1,1)*TYPEA(I,J)+DP(NEL,3,3)*TYPEC(I,J)
000 STIFF(I,J+4) = DP(NEL,1,2)*TYPEB(I,J)+DP(NEL,3,3)*TYPEC(I,J,I)
000 STIFF(J+4,I) = STIFF(I,J+4)
000 STIFF(I+4,J+4) = LP(NEL,2,2)*TYPEC(I,J)+DP(NEL,3,3)*TYPEA(I,J)
000 C
000 CM(I,J) = DFNS * TYPED(I,J)
000 CM(I+4,J+4) = CM(I,J)
000 C
000 VE(I,J) = ATD(1,1)*TYPEA(I,J) + ATD(3,3)*TYPEC(I,J)
000 VE(I,J+4) = ATD(3,3)*TYPEB(I,J)
000 VE(I+4,J) = ATD(3,3)*TYPEB(I,J)
000 VE(I+4,J+4) = ATD(2,2)*TYPEC(I,J) + ATD(3,3)*TYPEA(I,J)
000 C
000 300 CONTINUE
000 C
000 DO 110 I = 1,8
000 I1 = FK(NEL,I)
000 DO 110 J = 1,8
000 J1 = FK(NEL,J)
000 IF(I1.EQ.0.OR.J1.EQ.0) GO TO 110
000 IF(I1.LT.J1) GO TO 110
000 IF(I1.GT.IHR1) GO TO 104
000 L = J1 + (I1-J1) * I1 / 2
000 GO TO 105
000 104 L = J1 + I1 + (I1-IHR1) * IHR1
000 105 XKL(L) = XKL(L) + STIFF(I,J)
000 XM(L) = XM(L) + CM(I,J)
000 CHAR(L) = CHAR(L) + VE(I,J)
000 110 CONTINUE
000 C
000 900 CONTINUE
000 C
000 C
000 666 FORMAT(8E15.7)
000 600 FORMAT(/' SUB STIFF1'/)
000 610 FORMAT(4F16.6)
000 RETURN
000 END
000 BELT,SIH NASA*IPFS,DISPL,,,163543132410
000 SUBROUTINE DISPL(N1)
000 PARAMETER NFT=150,NFLS=150,MX= 5000,NF=NFT*2
000 COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NHC,MACT,ISW(5)
000 COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFT),Z(NFT),KK(NELS,8)
000 COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHB,IHBI,LT,LAST,NFREE
000 C
000 WRITE(6,620) MACT,NI
000 620 FORMAT(/' FORCE VECTOR SIZE =',I5,' NI =',I4/)
000 NF = LAST + 1
000 NM = LAST + NFREE
000 WRITE(6,667) (XKL(I),I=MM,NN)
000 667 FORMAT(9E14.6)
000 C
000 CALL FACIOR(XKL,IHB,IHBI,LT,LAST,NFREE)
000 C
000 CALL SOLIN (XKL,IHB,IHBI,LT,LAST,NFREE)
000 C

```

```

000      WRITE(6,600)                                00001800
000      DO 280 I = 1,NBC                             00001900
000      J = (I-1)*2 + 1 + LAST                       00002000
000      JJ = J + 1                                    00002100
000      280 WRITE(6,610) I,XKL(J),XKL(JJ)           00002200
000      600 FORMAT(/// 10X,'GLOBAL DISPLACEMENT'//5X,'NODE', 00002300
000      110X,'Z-DISPL',10X,'P-DISPL')              00002400
000      610 FORMAT(5X,14,2E20.7)                   00002500
000      RETURN                                        00002600
000      END                                           00002700
000  DELT,SIH NASA*IPF9,SIRAIN,,,226151130610
000      SUBROUTINE SIRAIN(NT)                          00000100
000      PARAMETER NF=150,NFLS=150,MX= 5000,NF=NFT*2  00000200
000      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5) 00000300
000      COMMON /BLK1/ W(4),F(4),AR(4),RR(4),CR(4),AZ(4),BZ(4),CZ(4), 00000400
000      * HQ(4),CIN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4), 00000500
000      * AU,B0,C0,IC,JC,KC,LC,NFL                   00000600
000      COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFT),7(NFT),KK(NELS,8) 00000700
000      COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHP,IHRI,LT,LAST,NFREE 00000800
000      COMMON /BLK4/ DE(3,3),AII(3,3),AD(3,3),RD(3,3),STIFF(8,8),CM(8,8), 00000900
000      * VE(8,8),AI(NFLS) \                          00001000
000      COMMON /BLK5/ GR(NELS,3),STRAIR(NELS,3),ALPHA,BETA,GAMMA,DELT,XX, 00001100
000      * AT,XNU,E,SM,SMS,VOIDI,CAPA,HAMD,BET        00001200
000      COMMON /BLK6/ BT(NELS,8,3),ARM(NELS,4),AZM(NELS,4),AOJ(NELS) 00001300
000      COMMON /BLK8/ LP(NELS,3,3),PRINS(NELS,3)     00001400
000      DIMENSION U(4),V(4),L(3,3)                  00001500
000      C                                             00001600
000      WRITE(6,600) N1                               00001700
000      C                                             00001800
000      DO 900 NEL = 1,NELEM                          00001900
000      DO 200 I = 1,4                                 00002000
000      IN = IJKL(NEL,1)                              00002100
000      II = (IN-1) * 2 + 1 + LAST                    00002200
000      V(I) = XKL(II)                                00002300
000      200 U(I) = XKL(II+1)                          00002400
000      620 FORMAT(4E15.6)                           00002500
000      C                                             00002600
000      DO 210 I = 1,3                                 00002700
000      DO 210 J = 1,3                                 00002800
000      210 D(I,J) = DP(NEL,I,J)                     00002900
000      C                                             00003000
000      C      COMPUTE STRESSES AT THE CTR.           00003100
000      C                                             00003200
000      EZ = 0.                                       00003300
000      ER = 0.                                       00003400
000      GM = 0.                                       00003500
000      DO 100 I = 1,4                                 00003600
000      EZ = FZ - ARM(NEL,I) * V(I) / AOJ(NEL)       00003700
000      ER = FR - ARM(NEL,I) * U(I) / AOJ(NEL)       00003800
000      100 GM = CM - (ARM(NEL,I) * U(I) + AZM(NEL,I) * V(I)) / AOJ(NEL) 00003900
000      SIGMZ = D(1,1) * EZ + D(1,2) * ER            00004000
000      SIGMR = D(2,2) * ER + D(2,1) * FZ            00004100
000      SHEAR = D(3,3) * GM                          00004200
000      650 FORMAT(15,7F15.6)                        00004300
000      C                                             00004400
000      GR(NEL,1) = SIGMZ + GR(NEL,1)                00004500
000      GR(NEL,2) = SIGMR + GR(NEL,2)                00004600

```

```

000      GP(NEL,3) = SHEAR + GB(NEL,3)          00004700
000      STRAIN(NEL,1) = S10M2                  00004800
000      STRAIN(NEL,2) = S10MR                 00004900
000      STRAIN(NEL,3) = SHEAR                  00005000
000      C                                       00005100
000      C      COMPUTE PRINCIPAL STRESS. (COMP. IS POSITIVE HERE) 00005200
000      C                                       00005300
000      SGR = ((GB(NEL,1) - GB(NEL,2)) / 2.) **2 + GP(NEL,3)**2)**.5 00005400
000      SGR = (GB(NEL,1) + GB(NEL,2)) / 2.     00005500
000      PRINS(NEL,1) = SGR + SGR               00005600
000      PRINS(NEL,2) = SGR - SGR               00005700
000      PRINS(NEL,3) = SGR                     00005800
000      900 CONTINUE                             00005900
000      C                                       00006000
000      WRITE(6,010) (I,GB(I,J),J=1,3),(PRINS(I,K),K=1,3),I=1,NELEM 00006100
000      000 FORMAT(1H1,10X,'LOCAL STRESS AT THE END OF',14,' LOAD INCREMENT',/ 00006200
000      1 5X,'ELEM',10X,'STRESS AT CTR',30X,'PRINCIPAL STRESS',/ 00006300
000      610 FORMAT(10,3F13.5,3X,3E13.5)        00006400
000      RETURN                                    00006500
000      END                                        00006600
000      *ELI*SIH *NASA*IPF9,PLASTIC,,,163544132410
000      SUBROUTINE PLASTC                               00006700
000      PARAMETER NF=150,NFLE=150,MX= 500,NF=NFT*2 00006800
000      COMMON /BLK0/ IJLL(20),INODE,NFLEM,NAPC,NPC,MACT,IS#(5) 00006900
000      COMMON /BLK2/ IF(NF,2),IJKL(NF(5,4)),K(NF1),Z(NFT),KK(NELS,8) 00007000
000      COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHF,IHF1,LT,LAST,NFREL 00007100
000      COMMON /BLK4/ U(3,3),A1(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8), 00007200
000      * VE(8,8),A1(NFLE)                             00007300
000      COMMON /BLK7/ UDB(NF),UDBF(NF),FOV(NF),HLD(NF),PB(NF),DISP(NF), 00007400
000      * DDBF(NF)                                       00007500
000      COMMON /DYN/ CBAR(MX),XF(MX),AA(MX)           00007600
000      C                                       00007700
000      CALL ZERO(AA,MX,1)                             00007800
000      NLOAD = -ISP(1)                                00007900
000      SCAL = NLOAD                                   00008000
000      DO 100 I = 1,MAC1                              00008100
000      F(1) = P(1) / SCAL                             00008200
000      100 XKL(LAST+I) = F(1)                         00008300
000      C                                       00008400
000      DO 900 N1 = 1,NLOAD                            00008500
000      C                                       00008600
000      IF(N1.EQ.1) GO TO 800                          00008700
000      CALL DIAGNL(XPL,P,NFREL,IHB,IHF1,LT,LAST)     00008800
000      C                                       00008900
000      CALL STIFFZ                                    00009000
000      C                                       00009100
000      200 CALL DISPL(P,1)                             00009200
000      DO 120 I = 1,NFREL                             00009300
000      AA(1) = AA(1) + XKL(LAST+I)                   00009400
000      120 DISP(I) = AA(1)                            00009500
000      C                                       00009600
000      WRITE(6,000) N1                                00009700
000      DO 130 I = 1,INODE                             00009800
000      JU = I + 2                                     00009900
000      II = JU - 1                                    00010000
000      130 WRITE(6,010) I,AA(II),AA(JU)             00010100
000      C                                       00010200

```

```

000      CALL STRAIN(NI)                                00003700
000      C                                              00003800
000      900 CONTINUE                                  00003900
000      C                                              00004000
000      600 FORMAT(1H1,'X',10I, 'DISPL. AT THE END OF',I4,' LOAD INCREMENT',// 00004100
000      1 5X,'NODE',I4,'Z-DISPL',15X,'R-DISPL'//)      00004200
000      610 FORMAT(I10,'E2U.7)                       00004300
000      STOP                                          00004400
000      END                                           00004500
000      WLLT,SIH NASA*1PFF,STIFF2,,163551132410
000      SUBROUTINE STIFF2                                00000100
000      PARAMETER NFI=150,NELS=150,MX= 5000,NF=NFI*2  00000200
000      COMMON /BLK0/ TITLE(20),INODE,NFLEM,NAPC,NHC,MACT,ISW(5) 00000300
000      COMMON /BLK1/ W(4),H(4),AR(4),PR(4),CR(4),A7(4),BZ(4),CZ(4), 00000400
000      * BR(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEU(4,4), 00000500
000      * AO,BO,CO,IC,JC,KC,LC,NEL                    00000600
000      COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFI),Z(NFI),KK(NELS,8) 00000700
000      COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHR,IHRI,LT,LAST,NFREE 00000800
000      COMMON /BLK4/ U(3,3),AID(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8), 00000900
000      * VE(8,8),A1(NELS)                            00001000
000      COMMON /BLK5/ GR(NELS,3),STRAIB(NELS,3),ALPHA,BETA,GAMMA,DELTA,XK, 00001100
000      * AT,XIU,E,SM,SMS,UTDI,CAPA,RAMP,BET          00001200
000      COMMON /BLK6/ PRINS(NELS,3)                  00001300
000      C                                              00001400
000      REWIND 2                                       00001500
000      C                                              00001600
000      XNUS = XNU * 1.0                               00001700
000      VA = YNU + 1.0                                 00001800
000      VB = 1.+XNUS-1.0                              00001900
000      VC = 2.*(XNUS-1.0) - 1.0                     00002000
000      C                                              00002100
000      DO 900 NEL = 1,NLEEM                          00002200
000      READ(2) TYPEA,TYPEB,TYPEC,TYPEU              00002300
000      IF(NEL.LE.15W(5)) GO TO 800                  00002400
000      DO 200 I = 1,3                                00002500
000      DO 200 J = 1,3                                00002600
000      200 DP(NEL,I,J) = U(I,J)                     00002700
000      C                                              00002800
000      SIGZ = QB(NEL,1) + STRAIB(NEL,1)/2.          00002900
000      SIGR = QB(NEL,2) + STRAIB(NEL,2)/2.          00003000
000      TAUZ = QB(NEL,3) + STRAIB(NEL,3)/2.          00003100
000      DELP = VA * (STRAIB(NEL,1)+STRAIB(NEL,2))/3. 00003200
000      PP = VA * (SIGZ+SIGR) / 3.                   00003300
000      PSW = PP * PP                                 00003400
000      SZZ = 2.*VB*SIGZ + VC*SIGR                   00003500
000      SRK = 2.*VB*SIGR + VC*SIGZ                   00003600
000      SZR = 6.*TAUZ                                 00003700
000      TJ = (VB*SIGZ*SIGZ+VB*SIGR*SIGR+VC*SIGZ*SIGR)/3. + TAUZ*TAUZ 00003800
000      CALL AREAA(NEL,AREA)                          00003900
000      RATIO = AREA / A1(NEL)                       00004000
000      VOIDR = RATIO * (1.+VOIDR) - 1.               00004100
000      VDR = VOIDR + 1.                              00004200
000      TTJ = 3.*TJ                                   00004300
000      ETM = 1.+TTJ/(PSW+SMS)                       00004400
000      POW = 1.-CAPA/RAMP                             00004500
000      PU = PP*ETM*POW                               00004600
000      AA = SMS * (2.*PP-PO) / 3.                   00004700

```



```

000      B6 = 3.*AA*VDR*SMS*PP*PO / BET          00004800
000      AR(1) = SZZ + AA                        00004900
000      AR(2) = SRK + AA                        00005000
000      AR(3) = SZK                             00005100
000      BR(1) = SZZ                             00005200
000      BR(2) = SRK                             00005300
000      BR(3) = SZK                             00005400
000      DF1 = 0.                                00005500
000      DFK = 0.                                00005600
000      DO 220 I = 1,3                          00005700
000      DF1 = DF1 + AR(I) * STRA1B(NEL,I)        00005800
000      220 DFK = DFK + BR(I) * STRA1B(NEL,I)    00005900
000      DFJ = PO * SMS * DELP                   00006000
000      POW = -CAPA/RAMD                         00006100
000      DFK = ETM **POW * (DFK-2.*TTJ/PP*DELP)  00006200
000      DF = DFI - DFJ - DFK * (1.-CAPA/RAMD)   00006300
000      ASQ = SMS * PP * (PO-PP)               00006400
000      ETA = SQRT(TTJ) / PP                    00006500
000      WRITE(6,620) VOIDR,TTJ,ASQ,DF,NEL,SM,ETA 00006600
000      620 FORMAT(5X,'VOIDR=',F10.5,' 3J=',E12.5,' ASQ=',E12.5,' DF=', 00006700
000      1 E12.5,' 17,' M=',E12.5,' ETA=',E12.5) 00006800
000      IF(DFK.LT.0.) GO TO 800                 00006900
000      DO 300 I = 1,3                          00007000
000      BR(I) = 0.                              00007100
000      CR(I) = 0.                              00007200
000      DO 300 J = 1,3                          00007300
000      BR(I) = BR(I) + D(I,J)*AR(J)           00007400
000      300 CR(I) = CR(I) + AR(J)*D(J,I)       00007500
000      DEN = 0.                                00007600
000      DO 310 I = 1,3                          00007700
000      310 DEN = DEN + AR(I)*BR(I)            00007800
000      DEN = DEN + BR                             00007900
000      DO 320 I = 1,3                          00008000
000      DO 320 J = 1,3                          00008100
000      320 DP(NEL,I,J) = D(I,J) - BR(I)*CR(J) /DEN. 00008200
000      C                                         00008300
000      800 CONTINUE                            00008400
000      DO 100 I = 1,4                          00008500
000      DO 100 J = 1,4                          00008600
000      STIFF(I,J) = DP(NEL,1,1)*TYPEA(I,J)+DP(NEL,3,3)*TYPEC(I,J) 00008700
000      STIFF(I,J+4) = DP(NEL,1,2)*TYPER(I,J)+DP(NEL,3,3)*TYPEB(J,I) 00008800
000      STIFF(J+4,1) = STIFF(I,J+4)            00008900
000      100 STIFF(I+4,J+4) = DP(NEL,2,2)*TYPEC(I,J)+DP(NEL,3,3)*TYPEA(I,J) 00009000
000      C                                         00009100
000      DO 210 I = 1,8                          00009200
000      II = KK(NEL,I)                          00009300
000      DO 210 J = 1,8                          00009400
000      JJ = KK(NEL,J)                          00009500
000      IF(II.EQ.0.OR.JJ.EQ.0) GO TO 210        00009600
000      IF(II.LT.JJ) GO TO 210                 00009700
000      IF(II.GT.IHBI) GO TO 214               00009800
000      L = JJ + (II-1) * II / 2               00009900
000      GO TO 215                               00010000
000      214 L = JJ + LT + (II-IHB) * IHBI      00010100
000      215 XKL(L) = XKL(L) + STIFF(I,J)       00010200
000      210 CONTINUE                            00010300
000      900 CONTINUE                            00010400

```

```

000 C
000 RETURN 00010500
000 END 00010600
000 DELT,SIH,NASA*IPF,STFP,,,163560132410 00010700
000 SUBROUTINE STEP 00000100
000 PARAMETER (NFI=150,NELS=150,MX= 5000,NF=NFI*2 00000200
000 COMMON /BLK1/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5) 00000300
000 COMMON /BLK2/ AKL(MX),P(NF),IMAX,IHB,IHB1,LT,LAST,NFREE 00000400
000 COMMON /BLK3/ GH(NELS,3),STRAIB(NELS,3),ALPHA,BETA,GAMMA,DELT,XK, 00000500
000 * AI,XMU,E,SI,SPS,VOIDI,CAPA,AMD,BET 00000600
000 COMMON /BLK7/ UDB(NF),UDD(NF),FOV(NF),UDD(NF),DB(NF),DISP(NF), 00000700
000 * DDO1(NF) 00000800
000 COMMON /BLK8/ UP(NELS,3,3),SIGB(NELS,3) 00000900
000 COMMON /DYN/ CBAR(MX),XM(MX),AA(MX) 00001000
000 NTIME=1 00001100
000 NDELT = ISW(1) 00001200
000 N = MACT 00001300
000 NPRNT = ISW(5) 00001400
000 IKOUNT = 250 00001500
000 KOUNT = IKOUNT 00001600
000 IPLT = 1 00001700
000 CALL ZERO(SIGB,NELS,3) 00001800
000 C 00001900
000 DO 2 I = 1, LAST 00002000
000 2 AA(I) = XM(I) + CBAR(I) * UDELT / 2. + XKL(I) * DELT * DELT / 4. 00002100
000 DO 150 I = 1, NFREE 00002200
000 150 AA(I+LAST) = P(I) 00002300
000 C 00002400
000 CALL FACTOR(AA,IHB,IHB1,LT,LAST,NFREE) 00002500
000 CALL SOLTN (AA,IHB,IHB1,LT,LAST,NFREE) 00002600
000 C 00002700
000 DO 3 I=1,N 00002800
000 UDB(I)=0. 00002900
000 UDDR(I)=0. 00003000
000 FOV(I)=0. 00003100
000 UDD(I) = AA(LAST+1) 00003200
000 3 DB(I)=0. 00003300
000 DO 4 I=1,N 00003400
000 DISP(I)= DB(I) + DELT*UDB(I) +(UDDR(I) +UDD(I))*DELT**2/4. 00003500
000 4 DDO1(I)= UDR(I) + (UDDR(I) +UDD(I))*DELT /2. 00003600
000 WRITE(6,100) NTIME 00003700
000 100 FORMAT (25X,'DISPLACEMENTS FOR TIME INCREMENT',I5,5X,'LINEAR') 00003800
000 WRITE (6,6997) (DISP(I),I=1,N) 00003900
000 6997 FORMAT(8E15.6) 00004000
000 101 FORMAT(8E15.6) 00004100
000 WRITE(6,102) 00004200
000 102 FORMAT('VELOCITY') 00004300
000 WRITE(6,101) (DDO1(I),I=1,N) 00004400
000 C 00004500
000 C INITIALIZE DB,STRAIB 00004600
000 C 00004700
000 CALL ZERO(STRAIB,NELS,3) 00004800
000 CALL ZERO(GH,NELS,3) 00004900
000 C 00005000
000 CALL STAN(NTIME) 00005100
000 C 00005200
000 C NOW READY FOR NEW INCREMENTS.. 00005300

```

```

000 C 00005400
000 C 10 CONTINUE 00005500
000 C 00005600
000 JJ = IHBI 00005700
000 DO 200 I = 1,NFREE 00005800
000 JJ = JJ + 1 00005900
000 SUN = 0. 00006000
000 SUM = 0. 00006100
000 IF(I.GE.IHB) GO TO 180 00006200
000 L = (I+1)*I / 2 - 1 00006300
000 DO 170 J = 1,JJ 00006400
000 IF(J.LE.1) L = L + 1 00006500
000 IF(J.GT.1.AND.J.LE.IHB) L = L + J - 1 00006600
000 IF(J.GT.IHB) L = L + IHB1 00006700
000 SUN = SUN + XM(L)*UB(J) 00006800
000 170 SUM = SUM + CBAR(L)*(UB(J)+UDDR(J)*DELT/2.) + XKL(L)*(DB(J)+DELT* 00006900
000 1UB(J)+UDDR(J)*DELT*DELT/4.) 00007000
000 GO TO 199 00007100
000 180 II = I - IHB + 1 00007200
000 L = LT + (II-1) * IHB 00007300
000 IF(JJ.GT.NFREE) JJ = NFREE 00007400
000 DO 190 J = II,JJ 00007500
000 IF(J.LE.1) L = L + 1 00007600
000 IF(J.GT.1) L = L + IHB1 00007700
000 SUN = SUN + XM(L)*UB(J) 00007800
000 190 SUM = SUM + CBAR(L)*(UB(J)+UDDR(J)*DELT/2.) + XKL(L)*(DB(J)+DELT* 00007900
000 1UB(J)+UDDR(J)*DELT*DELT/4.) 00008000
000 199 XM(LAST+1) = SUM 00008100
000 CHAR(LAST+1) = SUN 00008200
000 200 CONTINUE 00008300
000 C 00008400
000 IF(NTIME.GT.NDELT) GO TO 113 00008500
000 DO 12 I = 1,NFREE 00008600
000 12 AA(I+LAST) = P(1) - FOV(I) -XM(LAST+I) - CBAR(LAST+I) 00008700
000 C 00008800
000 300 CALL SOLTN (AA,IHB,IHB1,LT,LAST,NFREE) 00008900
000 C 00009000
000 DO 13 I=1,N 00009100
000 FOV(I)=0. 00009200
000 UDD(I) = AA(LAST+1) 00009300
000 DISP(I)= DB(I) + DELT*UB(I) + (UDD(I) + UDD(I))*DELT**2/4. 00009400
000 13 DDOT(I)= UDR(I) + (UDD(I) + UDD(I))*DELT /2. 00009500
000 C 00009600
000 CALL STAN(NTIME) 00009700
000 C 00009800
000 IF(NTIME.NE.NPKNT) GO TO 310 00009900
000 NPKNT = NPKNT + ISW(3) 00010000
000 WRITE(6,105) NTIME 00010100
000 105 FORMAT(10X,'DISPLACEMENT AT NTIME =',I5/5X,'NODE NO 2-DISPL 00010200
000 1 R-DISPL') 00010300
000 DO 106 I = 1,INODE 00010400
000 JJ = I * 2 00010500
000 II = JJ - 1 00010600
000 106 WRITE(6,620) I,DISP(II),DISP(JJ) 00010700
000 620 FORMAT(I9,2E15.6) 00010800
000 II = LAST + 1 00010900
000 JJ = LAST + NFREE 00011000

```

```

000      WRITE(6,600)
000      WRITE(6,101) (AA(I),I=1I,JJ)
000      WRITE(6,610)
000      WRITE(6,101) (CPAK(I),I=1I,JJ)
000      600 FORMAT(/5X,'(0),FORCE'/)
000      610 FORMAT(/5X,'FP'/)
000      310 CONTINUE
000      GO TO 10
000
000      C
000      113 WRITE(6,1968)
000      1968 FORMAT(/2X,'END OF ANALYSIS'///)
000      RETURN
000      END
000      WELT,SIH NASA*1PF$,STAN,,,163570132410
000      SUBROUTINE STAN(NTIME)
000      PARAMETER NFI=150,NELS=150,MX= 5000,NF=NFI*2
000      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5)
000      COMMON /BLK2/ IN(NF,2),IJKL(NELS,4),R(NFI),Z(NFI),KK(NELS,8)
000      COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHR,IHRI,LT,LAST,NFRFE
000      COMMON /BLK4/ D(3,3),ATD(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),
000      * VE(8,8),A1(NELS)
000      COMMON /BLK5/ GP(NELS,3),STRAIR(NELS,3),ALPHA,BETA,GAMMA,DELI,XK,
000      * AT,XMU,E,SM,SMS,VOIDI,CAPA,RAND,BET
000      COMMON /BLK6/ BT(NELS,8,3),ARM(NELS,4),AZM(NELS,4),AOJ(NELS)
000      COMMON /BLK7/ UDB(NF),UDUH(NF),FOV(NF),UDD(NF),DB(NF),DISP(NF),
000      * UDOT(NF)
000      COMMON /BLK8/ DP(NELS,3,3),SIGR(NELS,3)
000      COMMON /UYN/ CBAR(MX),XM(MX),AA(MX)
000      DIMENSION V(4),H(4),VD(4),EPS(3),EPSD(3),SIGE(3),SIGV(3),
000      * SIGF(3),SIGT(3),FV(8)
000
000      C
000      CALL ZERO(XM,MX,1)
000      REWIND 2
000      NN = NTIME / ISW(3)
000      MM = NN * ISW(3)
000      IRITE = 0
000      IF(MM.EQ.NTIME) IRITE = 1
000      IF(MM.EQ.NTIME) WRITE(6,600) NTIME
000      DO 900 NEL = 1,NELEM
000      DO 100 I = 1,4
000      IN = IJKL(NEL,I)
000      II = (IN-1)*2 + 1
000      V(1) = DISP(II)
000      U(1) = DISP(II+1)
000      VD(I) = UDOT(II)
000      UD(I) = UDOT(II+1)
000      100 CONTINUE
000      CALL ZERO(EPS,3,1)
000      CALL ZERO(EPSD,3,1)
000      DO 110 I = 1,4
000      EPS(1) = EPS(1) + ARM(NEL,I)*V(I) / AOJ(NEL)
000      EPS(2) = EPS(2) + AZM(NEL,I)*U(I) / AOJ(NEL)
000      EPS(3) = EPS(3)+(ARM(NEL,I)*U(I)+AZM(NEL,I)*V(I)) /AOJ(NEL)
000      EPSD(1) = EPSD(1) + ARM(NEL,I)*VD(I) / AOJ(NEL)
000      EPSD(2) = EPSD(2) + AZM(NEL,I)*UD(I) / AOJ(NEL)
000      110 EPSD(3) = EPSD(3)+(ARM(NEL,I)*UD(I)+AZM(NEL,I)*VD(I)) /AOJ(NEL)
000      DO 120 I = 1,3

```

```

000 SIGE(I) = 0. 00004400
000 SIGV(I) = 0. 00004500
000 SIGF(I) = 0. 00004600
000 DO 119 J = 1,3 00004700
000 SIGE(I) = SIGE(I) + D(I,J)*FPS(J) 00004800
000 SIGV(I) = SIGV(I) + ATD(I,J)*EPSD(J) 00004900
000 119 SIGF(I) = SIGF(I) + BD(I,J)*STRAIB(NEL,J) + AD(I,J)*GR(NEL,J) 00005000
000 120 SIGT(I) = SIGE(I) + SIGV(I) + SIGF(I) 00005100
000 C 00005200
000 C COMPUTE PRINCIPAL STRESSES 00005300
000 C COMPRESSION IS POSITIVE HERE 00005400
000 C 00005500
000 SQR = (((SIGT(1)-SIGT(2))/2.)**2 + SIGT(3)**2) **.5 00005600
000 SGI = -(SIGT(1) + SIGT(2)) / 2. 00005700
000 EPS(1) = SGI + SQR 00005800
000 EPS(2) = SGI - SQR 00005900
000 EPS(3) = SQR 00006000
000 IF(MM.EQ.NTIME) WRITE(6,610) NEL,SIGE,SIGV,SIGT,EPS 00006100
000 C 00006200
000 IF(ISW(2).EQ.1) CALL PSTIFF(NEL,SIGE,IPLST,IRITF) 00006300
000 IF(ISW(2).EQ.0) GO TO 900 00006400
000 C 00006500
000 C COMPUTE VISC. FORCE FOR EACH ELEMENT 00006600
000 C 00006700
000 DO 130 I = 1,8 00006800
000 FV(I) = 0. 00006900
000 DO 130 J = 1,3 00007000
000 130 FV(I) = FV(I) + BI(NEL,I,J) * SIGF(J) 00007100
000 C NOW ASSEMBLE IN GLOBAL FORM 00007200
000 DO 140 I = 1,8 00007300
000 II = KK(NEL,I) 00007400
000 IF(II.EQ.0) GO TO 140 00007500
000 FOV(II) = FOV(II) + FV(I) 00007600
000 IF(ISW(2).NE.1.OR.IPLST.NE.1) GO TO 140 00007700
000 DO 139 J = 1,8 00007800
000 JJ = KK(NEL,J) 00007900
000 IF(JJ.EQ.0) GO TO 139 00008000
000 IF(II.LT.JJ) GO TO 139 00008100
000 IF(II.GT.IHR1) GO TO 137 00008200
000 L = JJ+(II-1)*IJ/2 00008300
000 GO TO 138 00008400
000 137 L = JJ + LI + (II-IHB)*IHR1 00008500
000 138 XM(L) = XM(L) + STIFF(I,J) 00008600
000 139 CONTINUE 00008700
000 140 CONTINUE 00008800
000 C 00008900
000 C UPDATE (QB) AND (STRAIB). (ATRAIN RATE) 00009000
000 C 00009100
000 DO 150 I = 1,3 00009200
000 SIGR(NEL,I) = -SIGE(I) 00009300
000 GR(NEL,I) = ALPHA*GR(NEL,I)+BETA*STRAIB(NEL,I)+GAMMA*EPSD(I) 00009400
000 150 STRAIB(NEL,I) = EPSD(I) 00009500
000 C 00009600
000 900 CONTINUE 00009700
000 C 00009800
000 C UPDATE DISPL.,VEL., AND ACCELL. 00009900
000 C 00010000

```

```

000      160 NTIME = NTIME + 1                                00010100
000      170 DO 910 I = 1,MACT                                00010200
000          DF(I) = DISP(I)                                  00010300
000          UDB(I) = DDB(I)                                  00010400
000      910 UDB(I) = UDB(I)                                  00010500
000      600 FORMAT(///10X,'NTIME =',I5/2X,'ELEM',T10,'ELASTIC STRESS',T40,
000          1,'VISCOUS STRESS',T70,'TOL STRESS',T100,'PRINCIPAL STRESS'/) 00010600
000      610 FORMAT(I5,12E10.3)                               00010800
000      C                                                     00010900
000      620 FORMAT(//10X,'PRINCIPAL STRESS. NTIME =',I5/) 00011000
000      630 FORMAT(I10,3E13.5)                               00011100
000      C                                                     00011200
000          RETURN                                           00011300
000          END                                             00011400
000      WELT,SIH NASA*TPFS,PSTIFF,,163602132410
000          SUBROUTINE PSTIFF(NEL,SIGT,IPLST,IKITE)           00000100
000          PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2    00000200
000          COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5) 00000300
000          COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000          * FN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
000          * AU,BU,CU,DU,AV,CV,LC,MEL                      00000500
000          COMMON /BL2/ D(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8), 00000700
000          * VE(a,b),AL(NELS)                               00000800
000          COMMON /BLK3/ WB(NELS,3),STRAIR(NELS,3),ALPHA,BETA,GAMMA,DELTA,XK,
000          * AT,XNU,L,SM,SMS,VOIDI,CAPA,RAMD,BET           00001000
000          COMMON /BLK4/ UP(NELS,3,3),SIGB(NELS,3)         00001100
000          DIMENSION SIGT(3)                                00001200
000      C                                                     00001300
000      C      COMPRESSION IS POSITIVE                       00001400
000      C                                                     00001500
000          CALL ZERO(STIFF,8,8)                              00001600
000          READ(2) (TYPEA,TYPEB,TYPEC,TYPED)                00001700
000          IPLST = -1                                        00001800
000          IF (NEL.LE.ISW(5)) GO TO 800                     00001900
000          XNUS = XNU * XNU                                  00002000
000          VA = XNU + 1.                                     00002100
000          VB = 1.+XNUS-XNU                                  00002200
000          VC = 2.*(XNUS-XNU) - 1.                          00002300
000          DSIGZ = -SIGT(1)-SIGB(NEL,1)                     00002400
000          DSIGR = -SIGT(2)-SIGB(NEL,2)                     00002500
000          DTAUZ = -SIGT(3)-SIGB(NEL,3)                     00002600
000          SIGZ = -SIGT(1) + DSIGZ/2.                       00002700
000          SIGR = -SIGT(2) + DSIGR/2.                       00002800
000          TAUZ = -SIGT(3) + DTAUZ/2.                       00002900
000          DELP = VA*(DSIGZ+DSIGR) / 3.                     00003000
000          PP = VA * (SIGZ+SIGR) / 3.                       00003100
000          PSQ = PP * PP                                     00003200
000          SZZ = 2.*VB*SIGZ + VC*SIGR                       00003300
000          SRK = 2.*VB*SIGR + VC*SIGZ                       00003400
000          SZR = 6.*TAUZ                                     00003500
000          TJ = (VB*SIGZ*SIGZ+VB*SIGR*SIGR+VC*SIGZ*SIGR)/3. + TAUZ*TAUZ
000          TJJ = 3.*TJ                                       00003700
000          FTM = 1.+TTJ/(PSQ*SMS)                            00003800
000          POW = 1.-CAPA/RAMD                                00003900
000          PC = PP*ETM**POW                                   00004000
000          CALL AREA(NEL,AREA)                               00004100
000          RATIO = AREA / A1(NEL)                           00004200

```

```

000      VOIDK = RATIO * (1.+VOIDI) - 1.          00004300
000      VLK = VOIDK + 1.                        00004400
000      AA = SMS * (2.*PP-PO) / 3.              00004500
000      PH = 3.*AA+VDR*SMS*PP*PO / RET         00004600
000      AR(1) = SZZ + AA                        00004700
000      AR(2) = SRK + AA                        00004800
000      AR(3) = SZK                             00004900
000      PR(1) = SZZ                             00005000
000      PR(2) = SRK                             00005100
000      PR(3) = SZK                             00005200
000      DF1 = 0.                                00005300
000      DFK = 0.                                00005400
000      DO 220 I = 1,3                          00005500
000      DF1 = DF1 + AR(I) * STRAIB(NEL,I)       00005600
000      220 DFK = DFK + PR(I) * STRAIB(NEL,I)   00005700
000      DFJ = PO * SMS * DELP                  00005800
000      PGOV = -CAPA/RAMP                      00005900
000      DFK = ETM **POW * (DFK-2.*TTJ/PP*DELP) 00006000
000      DF = DF1 - DFJ - DFK * (1.-CAPA/RAMP)   00006100
000      ASG = SMS * PP * (PO-PP)              00006200
000      ETA = SQRT(TTJ) / PP                  00006300
000      IF(DF.LT.0.) GO TO 800                 00006400
000      IPLSI = 1                              00006500
000      IF(IRITE.EQ.1) WR1IF(6,620) VOIDK,TTJ,ASG,DF,NFL,SM,ETA 00006600
000      620 FORMAT(5X,'VOIDR=',F10.5,' 3J=',E12.5,'  ASG=',E12.5,'  DF=', 00006700
000      1 E12.5,' 17,' M=',E12.5,'  FTA=',E12.5) 00006800
000      DO 300 I = 1,3                          00006900
000      PR(I) = 0.                              00007000
000      CR(I) = 0.                              00007100
000      DO 300 J = 1,3                          00007200
000      BR(I) = BR(I) + D(1,J)*AR(J)           00007300
000      300 CR(I) = CR(I) + AR(J)*D(J,1)       00007400
000      DEN = 0.                                00007500
000      DO 310 I = 1,3                          00007600
000      310 DEN = DEN + AR(I)*BR(I)            00007700
000      DEN = DEN + BR                             00007800
000      DO 320 I = 1,3                          00007900
000      DO 320 J = 1,3                          00008000
000      320 DP(NEL,I,J) = -PR(1) * CR(J) / DEN 00008100
000      600 FORMAT(15,' AREA=', VOIDR, PO, UPU,,',4E15.6) 00008200
000      610 FORMAT(3E15.6)                    00008300
000      C                                       00008400
000      DO 100 I = 1,4                          00008500
000      DO 100 J = 1,4                          00008600
000      STIFF(I,J) = DP(NEL,1,1)*TYPEA(I,J)+DP(NEL,3,3)*TYPEC(I,J) 00008700
000      STIFF(I,J+4) = DP(NEL,1,2)*TYPER(I,J)+DP(NEL,3,3)*TYPEB(J,I) 00008800
000      STIFF(J+4,I) = STIFF(I,J+4)           00008900
000      100 STIFF(I+4,J+4) = DP(NEL,2,2)*TYPEC(I,J)+DP(NEL,3,3)*TYPEA(I,J) 00009000
000      C                                       00009100
000      800 RETURN                               00009200
000      END                                       00009300
000      WELT,SIH NASA*TYPE$.FACTOR,,226224130610
000      SUBROUTINE FACTOR(XK,IHB,IHB1,LT,LAST,NFREE) 00000100
000      C THIS SUBROUTINE PERFORMS FACTORING 00000200
000      DIMENSION XK(1) 00000300
000      N = NFREE 00000400
000      IHB1 = IHB1 00000500

```

| | | |
|-----|--|----------|
| 000 | DO 4 I=1,N | 00000600 |
| 000 | IF(I.GT.IHB1) GO TO 2 | 00000700 |
| 000 | K=1 | 00000800 |
| 000 | M=K+(I-1)*1/2 | 00000900 |
| 000 | GO TO 3 | 00001000 |
| 000 | 2 K=I-IHB1 | 00001100 |
| 000 | M=K+LT+(1-IHB)*IHB1 | 00001200 |
| 000 | 3 J=I+IHB1 | 00001300 |
| 000 | IF(J.GT.N) GO TO 4 | 00001400 |
| 000 | JJ=I+IHB1 | 00001500 |
| 000 | GO TO 5 | 00001600 |
| 000 | 4 JJ=N | 00001700 |
| 000 | 5 H=0.0 | 00001800 |
| 000 | LA=1-1 | 00001900 |
| 000 | LI=I+1 | 00002000 |
| 000 | IF(LA.EQ.0) GO TO 6 | 00002100 |
| 000 | DO 7 L=K,LA | 00002200 |
| 000 | IF(L.GT.IHB1) GO TO 50 | 00002300 |
| 000 | J = (L+1)*L/2 | 00002400 |
| 000 | GO TO 51 | 00002500 |
| 000 | 50 J = LT + IHB*(L-IHB1) | 00002600 |
| 000 | 51 A = XK(M) | 00002700 |
| 000 | H = H+A*A*XK(J) | 00002800 |
| 000 | 7 M=M+1 | 00002900 |
| 000 | 6 A=XK(M) | 00003000 |
| 000 | XK(M)=A-B | 00003100 |
| 000 | IF(I.FQ.N) GO TO 6 | 00003200 |
| 000 | DO 9 J=LB,JJ | 00003300 |
| 000 | SUM=0.0 | 00003400 |
| 000 | IF(J.GT.IHB1) GO TO 10 | 00003500 |
| 000 | K=1 | 00003600 |
| 000 | MM=K+(J-1)*J/2 | 00003700 |
| 000 | GO TO 11 | 00003800 |
| 000 | 10 K=J-IHB1 | 00003900 |
| 000 | MM=K+LT+(J-IHB)*IHB1 | 00004000 |
| 000 | 11 IF(LA.EQ.0) GO TO 9 | 00004100 |
| 000 | IF(K.GT.LA) GO TO 9 | 00004200 |
| 000 | DO 12 JA=K,LA | 00004300 |
| 000 | L=M-1+JA | 00004400 |
| 000 | IF(JA.GT.IHB1) GO TO 13 | 00004500 |
| 000 | L1=(JA+1)*JA/2 | 00004600 |
| 000 | GO TO 14 | 00004700 |
| 000 | 13 L1=LT+IHB*(JA-IHB1) | 00004800 |
| 000 | 14 SUM=SUM+XK(MM)*XK(L)*XK(L1) | 00004900 |
| 000 | 12 MM=MM+1 | 00005000 |
| 000 | 9 XK(MM)=(XK(MM)-SUM)/XK(M) | 00005100 |
| 000 | 8 CONTINUE | 00005200 |
| 000 | RETURN | 00005300 |
| 000 | END | 00005400 |
| 000 | WELT,SIH NASA*TPF,SOLTN,,,226226130610 | |
| 000 | SUBROUTINE SOLTN(XK,IHB,IHBI,LT,LAST,NFREE) | 00000100 |
| 000 | C THIS PORTION OF SUBROUTINE PERFORMS FORWARD-SUBSTITUTION | 00000200 |
| 000 | 000 DIMENSION XK(1) | 00000300 |
| 000 | C | 00000400 |
| 000 | N = NFREE | 00000500 |
| 000 | IHB1 = IHB | 00000600 |
| 000 | NF = LAST + 1 | 00000700 |

| | | | |
|-----|---|--|----------|
| 000 | C | | 00000800 |
| 000 | | 14 DO 1 K = 2,N | 00000900 |
| 000 | C | | 00001000 |
| 000 | | IF(K.GT.IHB1) GO TO 2 | 00001100 |
| 000 | | M=0 | 00001200 |
| 000 | | MM=K-1 | 00001300 |
| 000 | | M1=MM*K/2 | 00001400 |
| 000 | | GO TO 3 | 00001500 |
| 000 | | 2 M=K-IHB | 00001600 |
| 000 | | MM=IHB1 | 00001700 |
| 000 | | M1=M*IHB1+LT | 00001800 |
| 000 | | 3 SUM=0.0 | 00001900 |
| 000 | | DO 4 L=1,MM | 00002000 |
| 000 | | LL=L+M | 00002100 |
| 000 | | JJ=LL+M1 | 00002200 |
| 000 | | LL=LL+NF-1 | 00002300 |
| 000 | | 4 SUM=SUM+XK(JJ)*XK(LL) | 00002400 |
| 000 | | 1 XK(LL+1)=XK(LL+1)-SUM | 00002500 |
| 000 | | J = NF+N-1 | 00002600 |
| 000 | C | THIS PORTION OF SUBROUTINE PERFORMS BACK-SUBSTITUTION | 00002700 |
| 000 | | NF=NF+N-1 | 00002800 |
| 000 | | XK(NF)=XK(NF)/XK(LAST) | 00002900 |
| 000 | | DO 5 K=2,N | 00003000 |
| 000 | | L=N-K+1 | 00003100 |
| 000 | | IF(L.GT.IHB1) GO TO 6 | 00003200 |
| 000 | | I=L+(L-1)*L/2 | 00003300 |
| 000 | | GO TO 7 | 00003400 |
| 000 | | 6 I=L+(L-IHB)*IHB1+LT | 00003500 |
| 000 | | 7 IR=N-IHB | 00003600 |
| 000 | | IF(L.GT.1R) GO TO 8 | 00003700 |
| 000 | | J=IHB1 | 00003800 |
| 000 | | GO TO 9 | 00003900 |
| 000 | | 8 J=K-1 | 00004000 |
| 000 | | 9 SUM=0.0 | 00004100 |
| 000 | | DO 10 M=1,J | 00004200 |
| 000 | | MM=L+M | 00004300 |
| 000 | | IF(MM.GT.IHB1) GO TO 11 | 00004400 |
| 000 | | NN=L+(MM-1)*MM/2 | 00004500 |
| 000 | | GO TO 12 | 00004600 |
| 000 | | 11 NN=L+(MM-IHB)*IHB1+LT | 00004700 |
| 000 | | 12 MM=NF-N+MM | 00004800 |
| 000 | | 10 SUM=SUM+XK(NN)*XK(MM) | 00004900 |
| 000 | | MM=NF-N+L | 00005000 |
| 000 | | 5 XK(MM)=XK(MM)/XK(1)-SUM | 00005100 |
| 000 | | RETURN | 00005200 |
| 000 | | END | 00005300 |
| 000 | | WELT,SIH NASA*IPF%.SETUP,,,226231130610 | |
| 000 | | SUBROUTINE SETUP | 00000100 |
| 000 | | COMMON /BLK1/ W(4),H(4),AR(4),RR(4),CR(4),AZ(4),BZ(4),CZ(4), | 00000200 |
| 000 | | * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEU(4,4), | 00000300 |
| 000 | | * AO,BO,CO,IC,JC,KC,LC,NEL | 00000400 |
| 000 | | W(1) = 0.3478548 | 00000500 |
| 000 | | W(2) = 0.6521452 | 00000600 |
| 000 | | W(3) = W(2) | 00000700 |
| 000 | | W(4) = W(1) | 00000800 |
| 000 | | H(1) = 0.8611363 | 00000900 |
| 000 | | H(2) = 0.3399810 | 00001000 |

```

000      H(3) = -H(2)                                00001100
000      H(4) = -H(1)                                00001200
000      RN(1) = -1.                                  00001300
000      RN(2) = 1.                                    00001400
000      RN(3) = 1.                                    00001500
000      RN(4) = -1.                                  00001600
000      CN(1) = -1.                                  00001700
000      CN(2) = -1.                                  00001800
000      CN(3) = 1.                                    00001900
000      CN(4) = 1.                                    00002000
000      DN(1) = 1.                                    00002100
000      DN(2) = -1.                                  00002200
000      DN(3) = 1.                                    00002300
000      DN(4) = -1.                                  00002400
000      TROPI = 1.                                    00002500
000      RETURN                                         00002600
000      END                                             00002700
000  *ELT,SIH NASA*IPF$,GAUSS$,226232130610
000      SUBROUTINE GAUSS (K,AA)
000      COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000      * RN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEU(4,4),
000      * AO,B0,C0,IC,JC,KC,LC,NFL
000      IPT = 4
000      AA = 0.
000      DO 100 I = 1,IPT
000      X = F(I)
000      DO 100 J = 1,IPT
000      Y = H(J)
000      AA = AA + W(I) * W(J) * F(K,X,Y)
000 100 CONTINUE
000      RETURN
000      END
000      FUNCTION F(K,X,Y)
000      COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000      * RN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEU(4,4),
000      * AO,B0,C0,IC,JC,KC,LC,NEL
000      COMMON /BLK9/ M,N
000      FC = AO + B0*X + C0*Y
000  C
000      GO TO (10,20,30,40,50,60), K
000 10 CONTINUE
000      AMAN = AR(M)*AR(N)
000      BMAN = BR(M)*AR(N) + BR(N)*AR(M)
000      AMCN = AR(M)*CR(N) + AR(N)*CR(M)
000      BMBN = BR(M)*BR(N)
000      BMCN = BR(M)*CR(N) + CR(M)*BR(N)
000      CMCN = CR(M)*CR(N)
000      GO TO 100
000 20 CONTINUE
000      AMAN = AR(M)*AZ(N)
000      BMAN = BZ(N)*AR(M) + BR(M)*AZ(N)
000      AMCN = AR(M)*CZ(N) + CR(M)*AZ(N)
000      BMBN = BR(M)*RZ(N)
000      BMCN = BR(M) * CZ(N) + CR(M) * BZ(N)
000      CMCN = CZ(N)*CR(M)
000      GO TO 100
000 30 CONTINUE

```

| | | |
|-----|--|----------|
| 000 | AMAN = AZ(M)*AZ(N) | 00002600 |
| 000 | BMAN = BZ(M)*AZ(N) + BZ(N)*AZ(M) | 00002700 |
| 000 | AMCN = AZ(M)*CZ(N) + AZ(N)*CZ(M) | 00002800 |
| 000 | BMBN = BZ(M)*BZ(N) | 00002900 |
| 000 | BMCN = BZ(M)*CZ(N) + CZ(M)*BZ(N) | 00003000 |
| 000 | CMCN = CZ(M)*CZ(N) | 00003100 |
| 000 | GO TO 100 | 00003200 |
| 000 | 40 CONTINUE | 00003300 |
| 000 | XY = X * Y | 00003400 |
| 000 | FA = 1. + BN(M)*X + CN(M)*Y + DN(M)*XY | 00003500 |
| 000 | FB = 1. + BN(N)*X + CN(N)*Y + DN(N)*XY | 00003600 |
| 000 | F = FA*FB*FC/128. | 00003700 |
| 000 | RETURN | 00003800 |
| 000 | 50 F = (AR(M) + BK(M)*X + CK(M)*Y) / FC | 00003900 |
| 000 | RETURN | 00004000 |
| 000 | 60 F = (AZ(M) + BZ(M)*X + CZ(M)*Y) / FC | 00004100 |
| 000 | RETURN | 00004200 |
| 000 | 100 FA = AMAN + BMAN*X + AMCN*Y + BMBN*X*X + BMCN*X*Y + CMCN*Y*Y | 00004300 |
| 000 | F = FA/FC*.125 | 00004400 |
| 000 | RETURN | 00004500 |
| 000 | END | 00004600 |
| 000 | WELT,SIH NASA*TPF5,DIAGNL,,,226257130610 | |
| 000 | SUBROUTINE DIAGNL(XK,APF,NFREE,IHR,IHRI,LT,LAST) | 00000100 |
| 000 | DIMENSION XK(1),APF(1) | 00000200 |
| 000 | C | 00000300 |
| 000 | DO 200 I = 1, LAST | 00000400 |
| 000 | 200 XK(I) = 0. | 00000500 |
| 000 | C | 00000600 |
| 000 | DO 100 J = 1, NFREL | 00000700 |
| 000 | IF(J.GT.IHRI) GO TO 108 | 00000800 |
| 000 | L = (J+1) * J / 2 | 00000900 |
| 000 | GO TO 109 | 00001000 |
| 000 | 108 L = LT + IHR * (J - IHRI) | 00001100 |
| 000 | 109 XK(L) = 1. | 00001200 |
| 000 | 100 CONTINUE | 00001300 |
| 000 | C | 00001400 |
| 000 | DO 110 I = J, NFREL | 00001500 |
| 000 | 110 XK(LAST+I) = APF(I) | 00001600 |
| 000 | C | 00001700 |
| 000 | RETURN | 00001800 |
| 000 | END | 00001900 |
| 000 | WELT,SIH NASA*TPF5,ARFAA,,,163614132410 | |
| 000 | SUBROUTINE ARFAA(NEL,AREA) | 00000100 |
| 000 | PARAMETER NFT=150,NFLS=150,MX= 5000,NF=NFT*2 | 00000200 |
| 000 | COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFT),Z(NFT),KK(NELS,8) | 00000300 |
| 000 | COMMON /BLK7/ UDB(NF),UDDB(NF),FOV(NF),IUD(NF),DB(NF),DISP(NF), | 00000400 |
| 000 | * DOUT(NF) | 00000500 |
| 000 | DIMENSION RR(4),ZZ(4) | 00000600 |
| 000 | DO 100 I = 1,4 | 00000700 |
| 000 | IN = IJKL(NEL,I) | 00000800 |
| 000 | JJ = IN * 2 | 00000900 |
| 000 | II = JJ - 1 | 00001000 |
| 000 | PR(I) = R(IN) + DISP(JJ) | 00001100 |
| 000 | 100 ZZ(I) = Z(IN) + DISP(II) | 00001200 |
| 000 | AI = (RR(2)-RR(1))*(ZZ(4)-ZZ(1))-(RR(4)-RR(1))*(ZZ(2)-ZZ(1)) | 00001300 |
| 000 | AJ = (RR(3)-RR(2))*(ZZ(4)-ZZ(2))-(RR(4)-RR(2))*(ZZ(3)-ZZ(2)) | 00001400 |
| 000 | IF(AI.LT.0) AI = -AI | 00001500 |

```

000      IF(AJ.LT.0) AJ = -AJ
000      AREA = (AI + AJ) / 2.
000      RETURN
000      END
000  WELT,SIH NASA*TPFS,ZERO,,,22b242130610
000      SUBROUTINE ZERO(A,N,M)
000      DIMENSION A(1)
000      NM = N * M
000      DO 100 I = 1,NM
000  100  A(I) = 0.
000      RETURN
000      END
000  WXQT
000      MOVING WHEEL ON SOIL
000      65  48  5  65
000  1000  1  5
000  2000. 0.45 0.05787 29.25 2000. 0.45
000  200. 2000. 0.0006
000  36. 0.75 0.0001 0.05
000  1 24.0000 .0000 0 1
000  2 19.5000 .0000 0 1
000  3 14.0000 .0000 0 1
000  4 7.5000 .0000 0 1
000  5 .0000 .0000 1 1
000  6 24.0000 5.5000 0 0
000  7 19.5000 5.5000 0 0
000  8 14.0000 5.5000 0 0
000  9 7.5000 5.5000 0 0
000  10 .0000 5.5000 1 1
000  11 24.0000 10.5000 0 0
000  12 19.5000 10.5000 0 0
000  13 14.0000 10.5000 0 0
000  14 7.5000 10.5000 0 0
000  15 .0000 10.5000 1 1
000  16 24.0000 14.5000 0 0
000  17 19.5000 14.5000 0 0
000  18 14.0000 14.5000 0 0
000  19 7.5000 14.5000 0 0
000  20 .0000 14.5000 1 1
000  21 24.0000 18.5000 0 0
000  22 19.5000 18.5000 0 0
000  23 14.0000 18.5000 0 0
000  24 7.5000 18.5000 0 0
000  25 .0000 18.5000 1 1
000  26 24.4000 22.5000 0 0
000  27 19.5000 22.5000 0 0
000  28 14.0000 22.5000 0 0
000  29 7.5000 22.5000 0 0
000  30 .0000 22.5000 1 1
000  31 25.4500 26.5000 0 0
000  32 20.0000 26.5000 0 0
000  33 14.0000 26.5000 0 0
000  34 7.5000 26.5000 0 0
000  35 .0000 26.5000 1 1
000  36 26.8000 30.0000 0 0
000  37 21.0000 30.0000 0 0
000  38 14.0000 30.0000 0 0

```

```

00001600
00001700
00001800
00001900
00000100
00000200
00000300
00000400
00000500
00000600
00000700

```

| | | | | | |
|-----|----|---------|---------|----|----|
| 000 | 39 | 7.5000 | 30.0000 | 0 | 0 |
| 000 | 40 | .0000 | 30.0000 | 1 | 1 |
| 000 | 41 | 29.2500 | 33.5000 | 0 | 0 |
| 000 | 42 | 22.5000 | 33.5000 | 0 | 0 |
| 000 | 43 | 15.2500 | 33.5000 | 0 | 0 |
| 000 | 44 | 8.2500 | 33.5000 | 0 | 0 |
| 000 | 45 | .0000 | 33.5000 | 1 | 1 |
| 000 | 46 | 29.2500 | 37.5000 | 0 | 0 |
| 000 | 47 | 23.5000 | 37.5000 | 0 | 0 |
| 000 | 48 | 16.5000 | 37.5000 | 0 | 0 |
| 000 | 49 | 9.0000 | 37.5000 | 0 | 0 |
| 000 | 50 | .0000 | 37.5000 | 1 | 1 |
| 000 | 51 | 29.2500 | 42.0000 | 0 | 0 |
| 000 | 52 | 23.5000 | 42.0000 | 0 | 0 |
| 000 | 53 | 16.5000 | 42.0000 | 0 | 0 |
| 000 | 54 | 9.0000 | 42.0000 | 0 | 0 |
| 000 | 55 | .0000 | 42.0000 | 1 | 1 |
| 000 | 56 | 29.2500 | 47.5000 | 0 | 0 |
| 000 | 57 | 23.5000 | 47.5000 | 0 | 0 |
| 000 | 58 | 16.5000 | 47.5000 | 0 | 0 |
| 000 | 59 | 9.0000 | 47.5000 | 0 | 0 |
| 000 | 60 | .0000 | 47.5000 | 1 | 1 |
| 000 | 61 | 29.2500 | 55.0000 | 0 | 1 |
| 000 | 62 | 23.5000 | 55.0000 | 0 | 1 |
| 000 | 63 | 16.5000 | 55.0000 | 0 | 1 |
| 000 | 64 | 9.0000 | 55.0000 | 0 | 1 |
| 000 | 65 | .0000 | 55.0000 | 1 | 1 |
| 000 | 1 | 2 | 7 | 6 | 1 |
| 000 | 2 | 7 | 12 | 11 | 6 |
| 000 | 3 | 12 | 17 | 16 | 11 |
| 000 | 4 | 17 | 22 | 21 | 16 |
| 000 | 5 | 22 | 27 | 26 | 21 |
| 000 | 6 | 27 | 32 | 31 | 26 |
| 000 | 7 | 32 | 37 | 36 | 31 |
| 000 | 8 | 37 | 42 | 41 | 36 |
| 000 | 9 | 42 | 47 | 46 | 41 |
| 000 | 10 | 47 | 52 | 51 | 46 |
| 000 | 11 | 52 | 57 | 56 | 51 |
| 000 | 12 | 57 | 62 | 61 | 56 |
| 000 | 13 | 3 | 8 | 7 | 2 |
| 000 | 14 | 8 | 13 | 12 | 7 |
| 000 | 15 | 13 | 18 | 17 | 12 |
| 000 | 16 | 18 | 23 | 22 | 17 |
| 000 | 17 | 23 | 28 | 27 | 22 |
| 000 | 18 | 28 | 33 | 32 | 27 |
| 000 | 19 | 33 | 38 | 37 | 32 |
| 000 | 20 | 38 | 43 | 42 | 37 |
| 000 | 21 | 43 | 48 | 47 | 42 |
| 000 | 22 | 48 | 53 | 52 | 47 |
| 000 | 23 | 53 | 58 | 57 | 52 |
| 000 | 24 | 58 | 63 | 62 | 57 |
| 000 | 25 | 4 | 9 | 8 | 3 |
| 000 | 26 | 9 | 14 | 13 | 8 |
| 000 | 27 | 14 | 19 | 18 | 13 |
| 000 | 28 | 19 | 24 | 23 | 18 |
| 000 | 29 | 24 | 29 | 28 | 23 |
| 000 | 30 | 29 | 34 | 33 | 28 |

| | | | | | |
|-----|----|----------|----|----------|----|
| 000 | 31 | 34 | 39 | 58 | 55 |
| 000 | 32 | 39 | 44 | 43 | 58 |
| 000 | 33 | 44 | 49 | 48 | 45 |
| 000 | 34 | 49 | 54 | 53 | 48 |
| 000 | 35 | 54 | 59 | 58 | 55 |
| 000 | 36 | 59 | 64 | 63 | 58 |
| 000 | 37 | 5 | 10 | 9 | 4 |
| 000 | 38 | 10 | 15 | 14 | 9 |
| 000 | 39 | 15 | 20 | 19 | 14 |
| 000 | 40 | 20 | 25 | 24 | 19 |
| 000 | 41 | 25 | 30 | 29 | 24 |
| 000 | 42 | 30 | 35 | 34 | 29 |
| 000 | 43 | 35 | 40 | 39 | 34 |
| 000 | 44 | 40 | 45 | 44 | 39 |
| 000 | 45 | 45 | 50 | 49 | 44 |
| 000 | 46 | 50 | 55 | 54 | 49 |
| 000 | 47 | 55 | 60 | 59 | 54 |
| 000 | 48 | 60 | 65 | 64 | 59 |
| 000 | 21 | -13.4152 | | -9.5128 | |
| 000 | 26 | -55.2063 | | -26.6738 | |
| 000 | 31 | -98.2728 | | -27.7647 | |
| 000 | 36 | -89.0538 | | -8.3968 | |
| 000 | 41 | -32.0671 | | 2.8452 | |

LEEJK ACCOUNT: UAH31736502F PROJECT: NASA

APPENDIX 4
DATA INPUT FORMAT

Card 1: FORMAT (20 A4)

TITLE - Title of the problem

Card 2: FORMAT (10I5)

- (1) INODE - No. of nodes
- (2) NELEM - No. of elements
- (3) NAPC - No. of applied point load
- (4) NBC - No. of free nodes

Card 3: FORMAT (10I5)

- (1) ISW(1) = 0, static analysis
= N, dynamic analysis for N steps
= -N, static elasto-plastic analysis for N load increments
- (2) ISW(2) = 0, Elastic analysis
= -1, Viscoelastic analysis
= 1, Viscoelastoplastic analysis
- (3) ISW(3) = M, Print for each Mth time step
- (4) ISW(4) = 0
- (5) ISW(5) = No. of rigid elements

Card 4: FORMAT (6F13.6)

- (1) E - Modulus of elasticity for rigid element
- (2) XNU - Poisson's ratio for rigid element
- (3) DENS - Soil density
- (4) DEPTH - Maximum soil depth

- (5) ES - Modulus of elasticity for soil
- (6) XNUS - Poisson's ratio for soil

Card 5: FORMAT (6F13.6)

- (1) AT = $T_{(r)} \times E_s$, where $T_{(r)}$ is the soil relaxation time in seconds
- (2) XK - Modulus of elasticity for soil
- (3) DELT - Magnitude of time step in seconds

Card 6: FORMAT (6F13.6)

- (1) PHI - Angle of internal friction
- (2) VOIDI - Initial void ratio
- (3) CAPA - Swelling index
- (4) RAMD - Compression index

Cards 7: FORMAT (5x, 2F10.4, 2I5)

- (1) Z(I) - *Z - coordinate value of Node I
- (2) R(I) - R - coordinate value of Node I
- (3) IZ = 0 if free to z-direction
≠ 1 if not
- (4) IR = 0 if free to R-direction
≠ 1 if not

Repeat INODE times in the order of node number

*Upward Z is positive

Cards 8: FORMAT (5x, 5I5)

4 corner nodes of an element in counter clockwise. Repeat NELEM times in the order of element number. Note that rigid element should be numbered first.

Card(s) 9: FORMAT (I5,2F10.4)

- (1) NODE - Node number with applied load

(2) PZ - Z-component

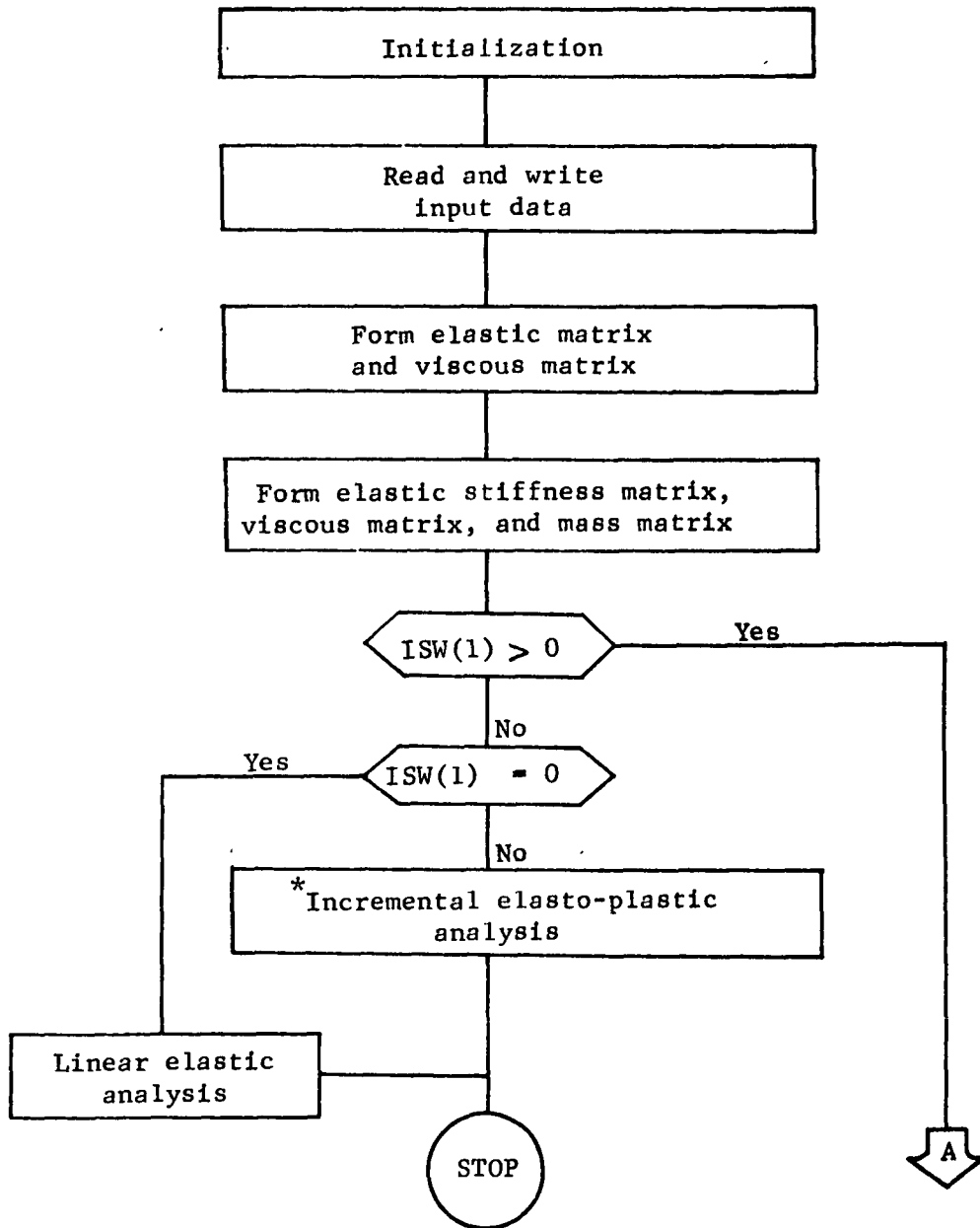
(3) PR - R-component

Note that the regular sign convention of theory of elasticity is used for input quantity.

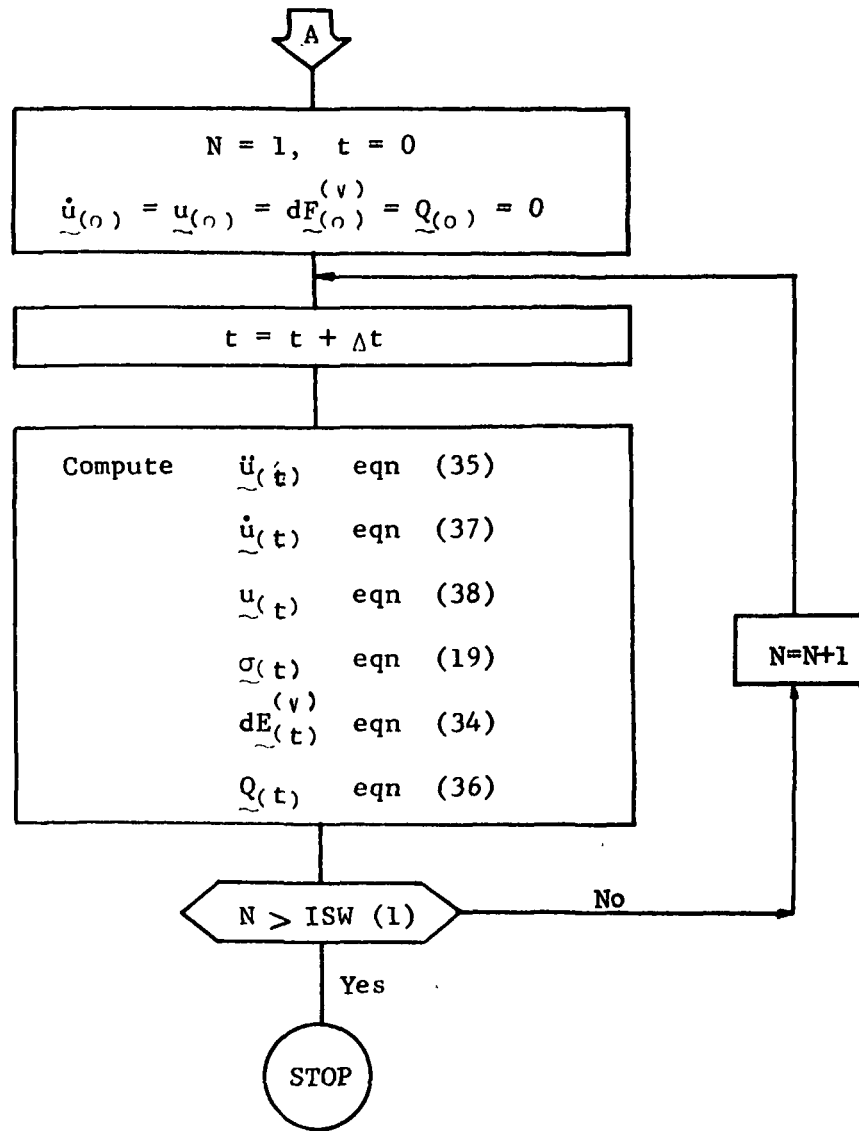
Repeat NAPC times.

APPENDIX 5

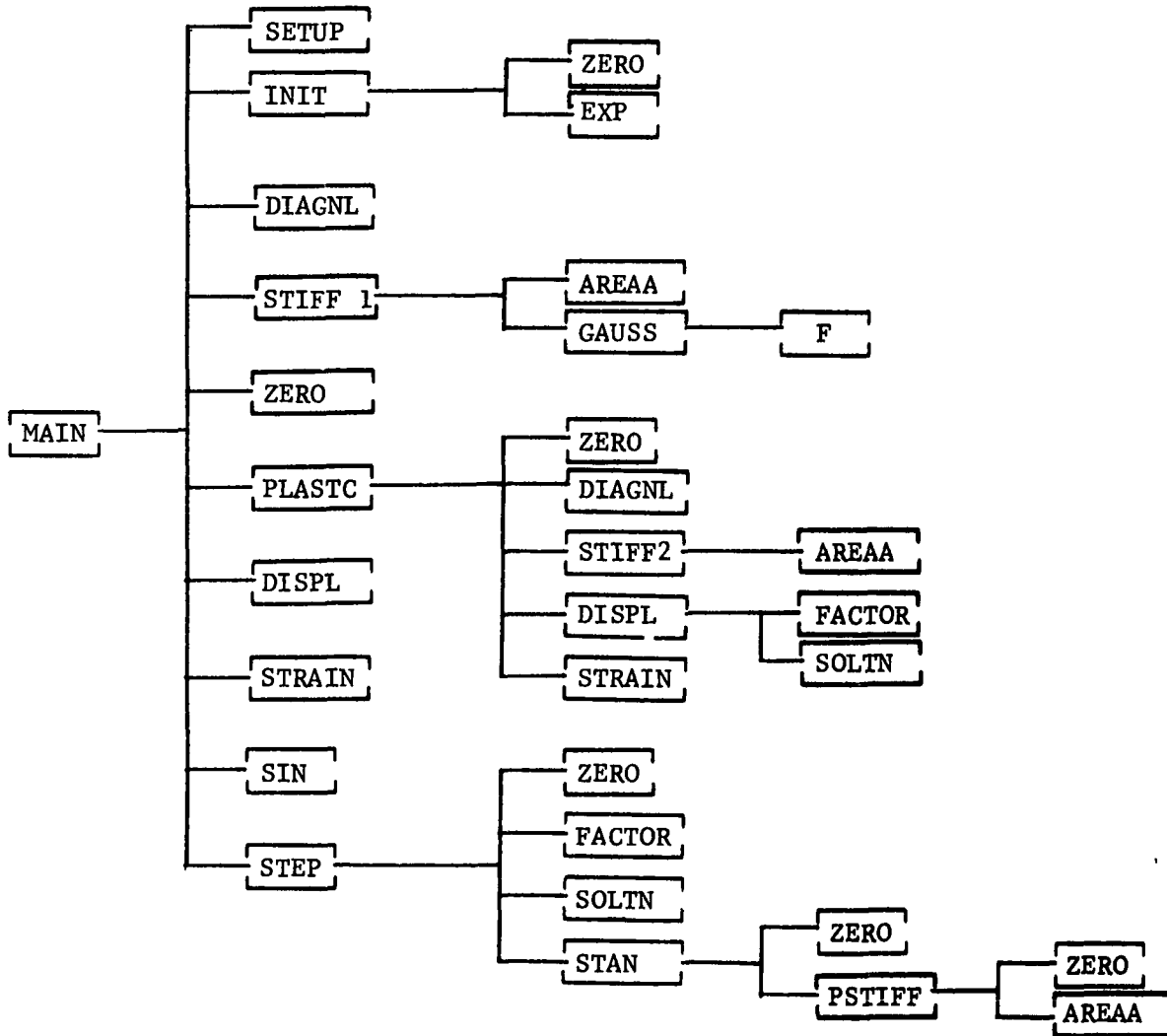
FLOW CHART



*For incremental elasto-plastic analysis, see App. 3 of Part I.



APPENDIX 6
 SUBROUTINE ORGANIZATION CHART



APPENDIX 7

DESCRIPTIONS OF SUBROUTINES

| Subroutine Name | Descriptions |
|--------------------|---|
| AREAA | Computes the cross sectional area of an element |
| DIAGNL | Clears one dimensional array and puts 1's on diagonal. |
| DISPL | Calls FACTOR and SOLTN, and prints displacement vector. |
| F | Functions to be integrated |
| FACTOR | Factors (forward substitution) a given simultaneous equations |
| GAUSS | Integrates by the Gaussian quadrature |
| INIT | Forms elastic matrix and viscous matrix |
| PSTIFF | Checks for yielding and forms plastic stiffness matrices for yielded elements |
| SETUP | Initializes integration constants |
| SOLTN | Backward substitution is performed to give a set of solutions to the given simultaneous equations. |
| STEP | Integrates the equation of motion by step integration scheme for dynamic analysis |
| STIFF 1 | Forms elastic stiffness matrix, consistent mass matrix, and viscous matrix. Also assembles in global form applying the boundary conditions. |
| STIFF 2 | Checks for yielding and forms elasto-plastic stiffness matrix, and assembles in global form applying the boundary conditions. |
| STRAIN | Computes strains, stresses, and principal stresses (compression is positive). |
| ZERO | Clears any given matrix |