N73-15368

NASA CR-124006

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

g1

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

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

TABLE OF CONTENTS (cont.)

LIST OF FIGURES

Page

PART I.

PART II·

LIST OF FIGURES (Cont.)

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 κ 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,κ) 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 cohesionless 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,

$$\frac{p}{p_o} - \frac{M^2}{M^2 + \eta^2} = 0 \tag{1}$$

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

$$M = \frac{6 \sin \varphi}{3 - \sin \varphi} \tag{2}$$

in which φ is the angle of internal friction.

The incremental plastic (irrecoverable) volumetric strain is

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

The overall void ratio change along the isotropic compression curve is

$$de = -\lambda \frac{dp_o}{p_o} \tag{4}$$

whereas the incremental recoverable void ratio is given by

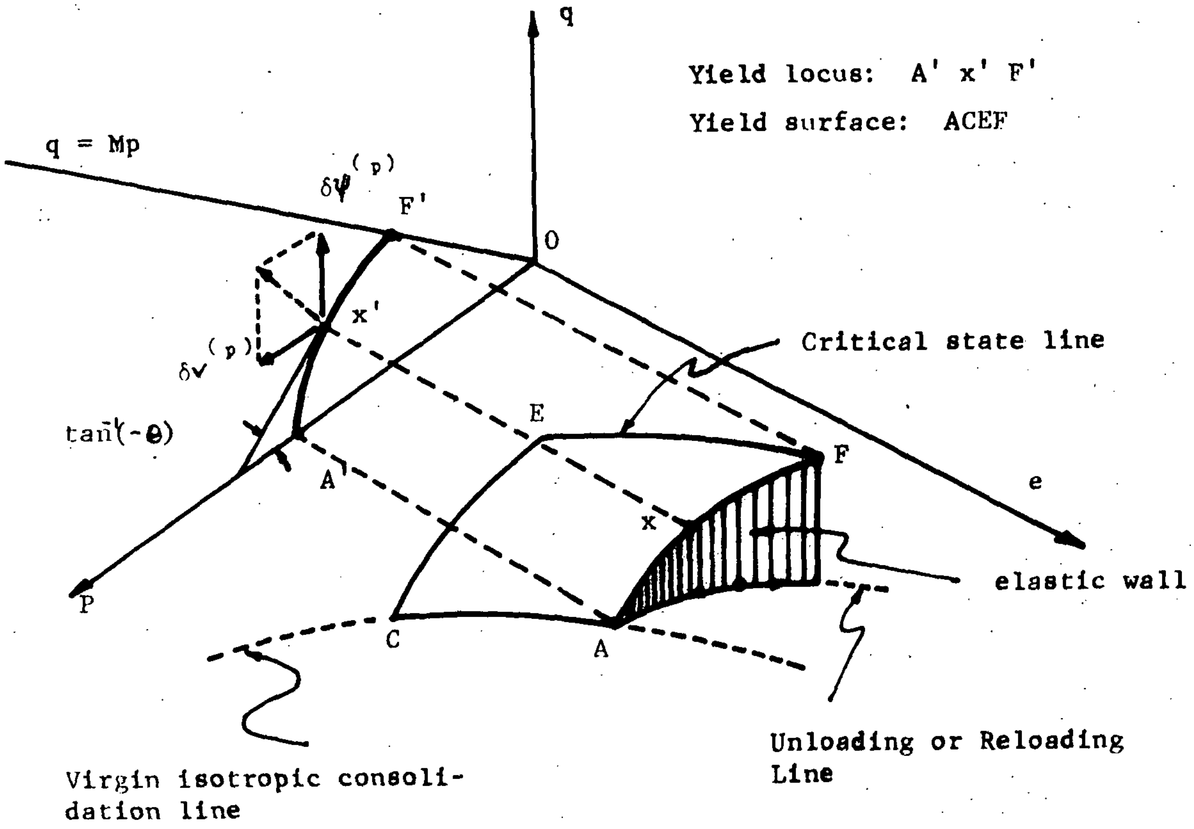


Figure 1: Yield Surface and Yield Locus (after Roscoe and Burland)

$$de^{(r)} = -\kappa \frac{dp_0}{p_0} \tag{5}$$

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

$$de^{(p)} = -(\lambda - \kappa)\frac{dp_0}{p_0} \tag{6}$$

At this point we introduce the equation of yield surface in the form [1]

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

in which p_e 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 $p_e = p_0$ in (7) leads to

$$p_0 = p\left(1 + \frac{\eta^2}{M^2}\right)^{(1-\kappa/\lambda)} \tag{8}$$

Under triaxial compression, the second deviatoric stress invariant becomes

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

which gives

$$q = \sqrt{3J} \tag{10}$$

Substituting (10) into (1) and rearranging yields

$$3J + pM^2\ (p - p_0) = 0$$

or

$$3J - A^2 = 0 \tag{11}$$

where

$$A^2 = -pM^2(p - p_0) \tag{12}$$

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,

$$F(J, A) = 3J - A^2 = 0 \tag{13}$$

$\overline{\partial A}\ \overline{\partial p_o}$

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,

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

$$d\psi_{mn}^{(p)} = \frac{\partial F}{\partial J}\frac{\partial J}{\partial \sigma_{mn}}\ d\lambda \tag{18}$$

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

$$dv^{(p)} = \frac{\lambda - \kappa}{1+e}\quad \frac{dp_o}{p_o} \tag{19}$$

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

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

The incremental total plastic strain tensor is given by

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

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

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

where

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

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

also,

$$\frac{\partial F}{\partial J}\, dJ = 3\,\frac{\partial J}{\partial \sigma_{mn}}\, d\sigma^{mn} = S_{mn}\, d\sigma^{mn} \tag{25a}$$

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

$$\frac{\partial F}{\partial A}\frac{\partial A}{\partial p_0} = -M^2\, p \tag{25c}$$

Substituting (25) into (23) gives

$$B_{mn} = \frac{S_{mn} + a\,\delta_{mn}}{b} \tag{26}$$

in which

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

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

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

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

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 $d\gamma_{mn}$ is the sum of the incremental elastic strain tensor $d\gamma^{(e)}_{mn}$ and the incremental plastic strain tensor $d\gamma^{(p)}_{mn}$. Therefore,

$$d\gamma^{(e)}_{mn} = d\gamma_{mn} - d\gamma^{(P)}_{mn} \tag{30}$$

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

$$d\sigma^{\alpha\beta} = D^{\alpha\beta mn}_{(e)}\, d\gamma^{(e)}_{mn} \tag{31}$$

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

$$d\sigma^{\alpha\beta} = D^{\alpha\beta mn}(d\gamma_{mn} - B_{mn} R_{ij}\ d\sigma^{ij}\ ) \tag{32}$$

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

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

or

$$R_{rs}\left[D^{rsmn}_{(e)}\ (d\gamma_{mn} - B_{mn}\ R_{\alpha\beta}\, d\sigma^{\alpha\beta})\right] - R_{\alpha\beta}\, d\sigma^{\alpha\beta} = 0$$

from which

$$R_{\alpha\beta}\, d\sigma^{\alpha\beta} = \frac{R_{rs}\quad D^{rsmn}\ d\gamma_{mn}}{1 + R_{rs}\ B_{mn}\ D^{rsmn}} \tag{33}$$

Substituting (33) into (32) gives

$$d\sigma^{\alpha\beta} = \left(D^{\alpha\beta mn}_{(e)} + D^{\alpha\beta mn}_{(P)}\right) d\gamma_{mn} \tag{34}$$

where

$$D^{\alpha\beta mn}_{(P)} = - \frac{D^{\alpha\beta kl}\quad B_{kl}\ R_{ij}\ D^{ijmn}}{1 + B_{rs}\ R_{tu}\ D^{rstu}} \tag{35}$$

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

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

$$dF = R_{\alpha\beta} d\sigma^{\alpha\beta} - M^2 p \; dp_0 \tag{36}$$

where dp_0 can be determined from (8),

$$dp_0 = g \; dp + h \; S_{ij} \; d\sigma_{ij} \tag{37}$$

in which

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

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

Substituting these in (36) yields

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

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 axi-symmetric cylindrical box. The material constants given by Namiq are angle of internal friction $\varphi = 35^{\circ}$, initial void ratio $e = 0.875$, initial density $\gamma = 0.0147$ 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 $E_s = 10$ N/cm^2 and Poisson's ratio $\nu_s = .45$ are used.

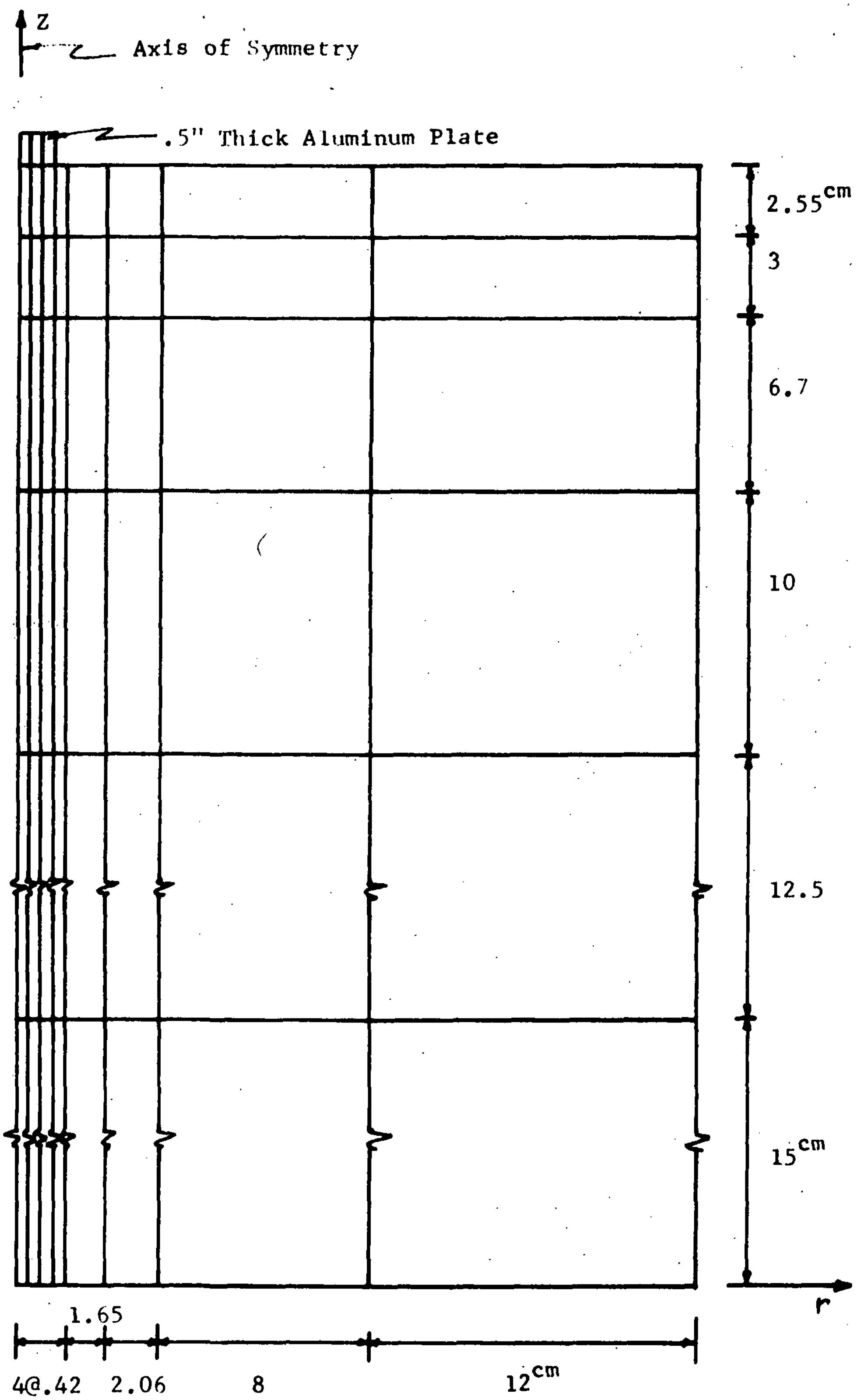


Figure 2: Plate Bearing Geometry

Legend	ϕ	e	ν	$\varkappa$	λ	E_s	ν_s
———	35^0	.875	.0147 N/cc	NA	NA	NA	NA
– – – –	"	"	"	.003	.05	10 N/cm²	.45
—·—	"	"	"	"	.13	"	"

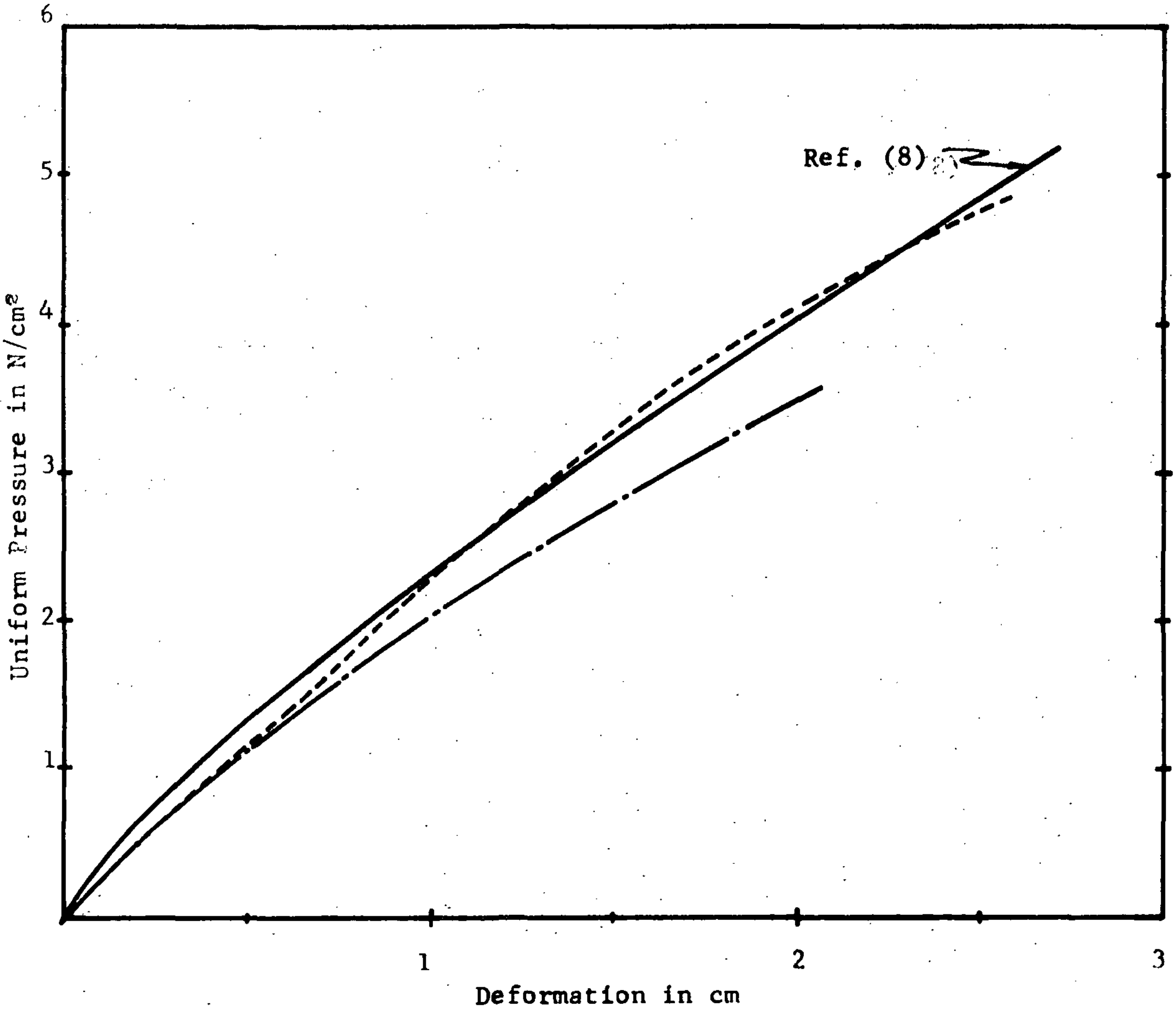


Figure 3: Deformation at Center of Plate

Deformed shapes of the finite elements adjacent to the bearing plate are shown in Figure 4 for the loading increments at F = 2.5 N/cm^8 and F = 5 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 N/cm^2 and Poisson's ration ν_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

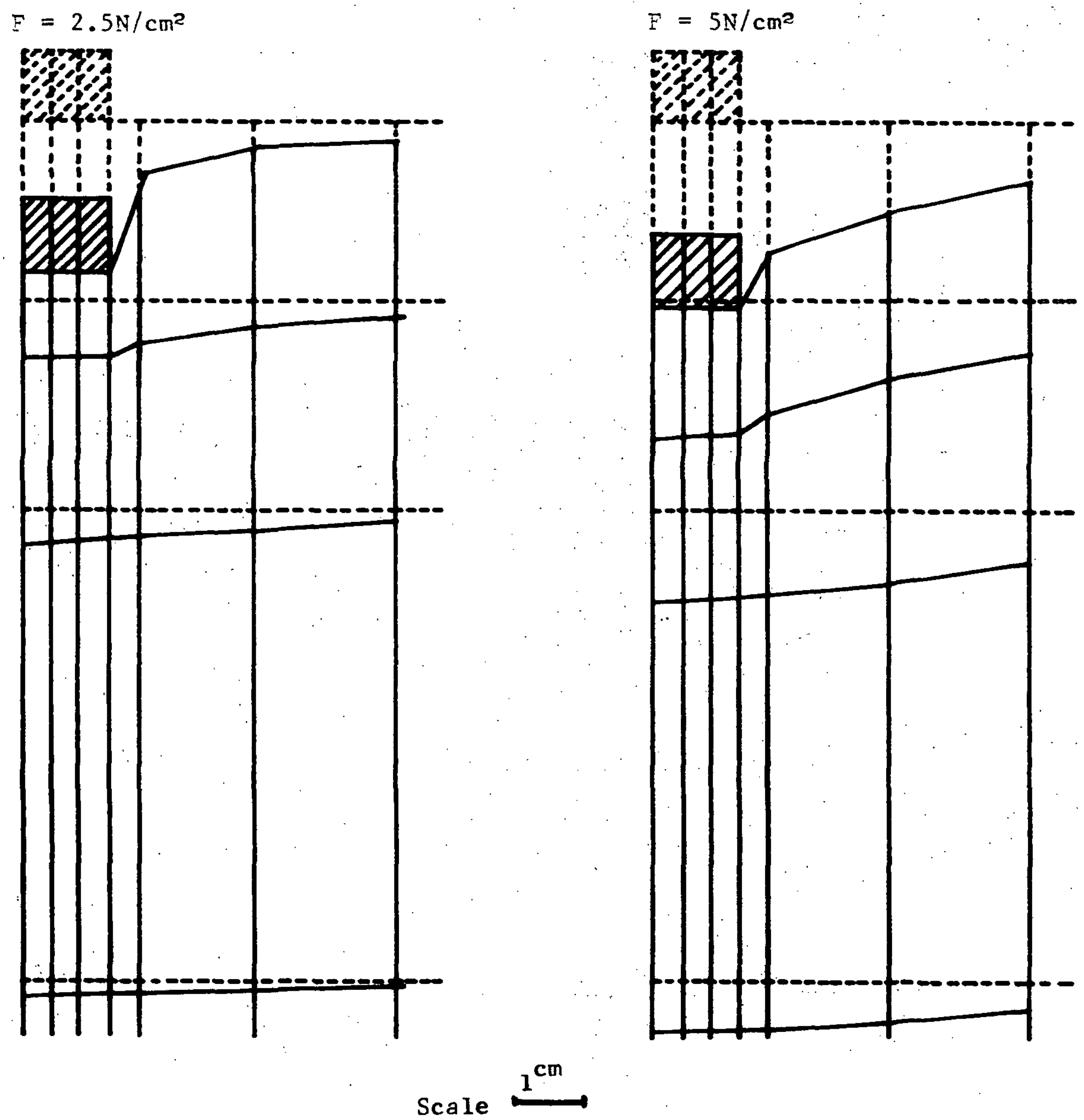


Figure 4: Deformed Geometry

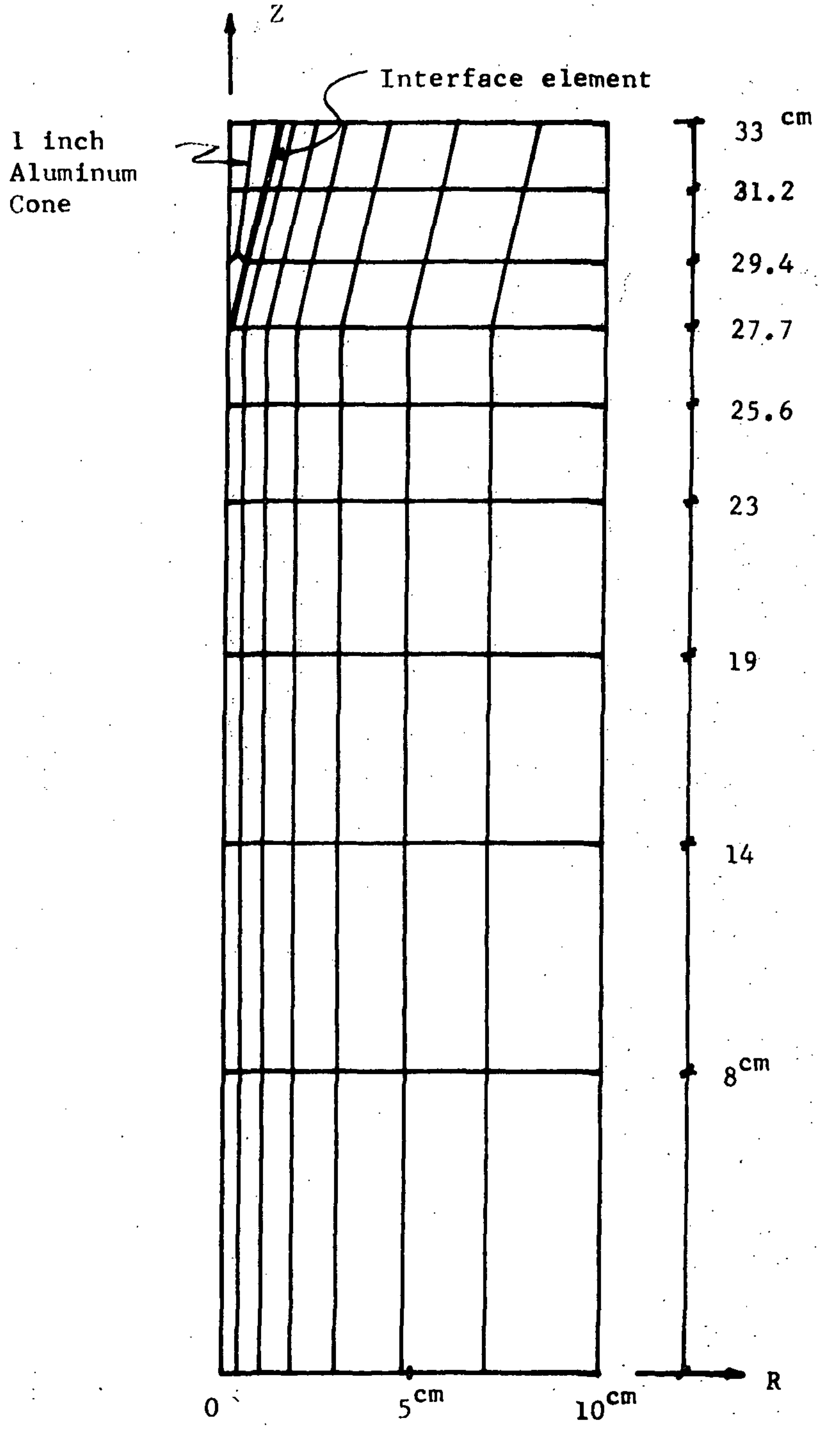
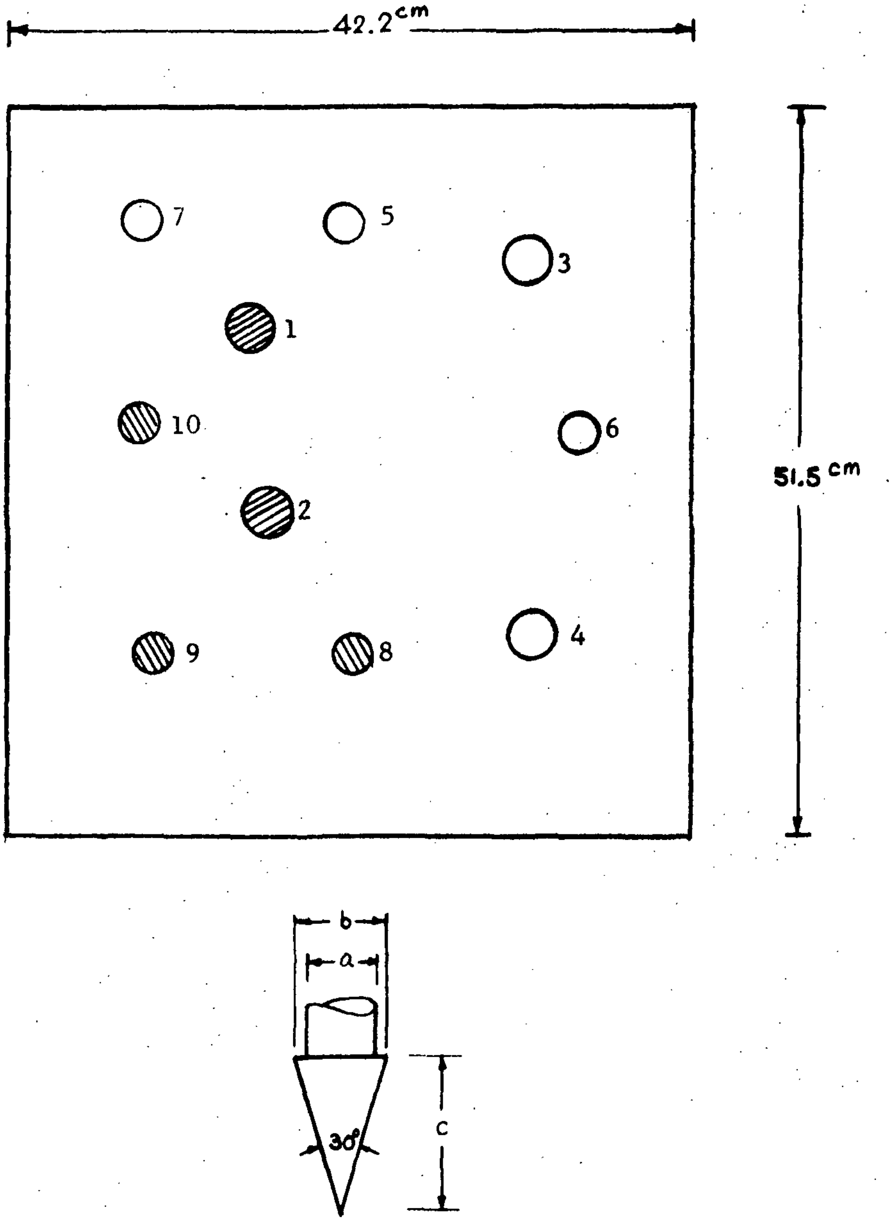


Figure 5:Cone-Penetrometer Geometry



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 ½ inch Smooth Cone } $a=1.7^{cm}$, $b=2^{cm}$, $c=3.9^{cm}$
8,9,10 ½ inch Rough Cone

Figure 6: Cone-Penetrometer Tests

Legend	Description
—◇—	Average of ½" Rough Cone Tests
—⊙—	" 1" " "
—▽—	" ½ Smooth Cone Tests
—△—	" 1" " " "
············	Finite Element Solution (1), λ=.07 *
— — — —	" " (2), λ=.13

* Other constants used are:
k = .006, ϕ = 35°, e = .76, γ = .0157 N/cc

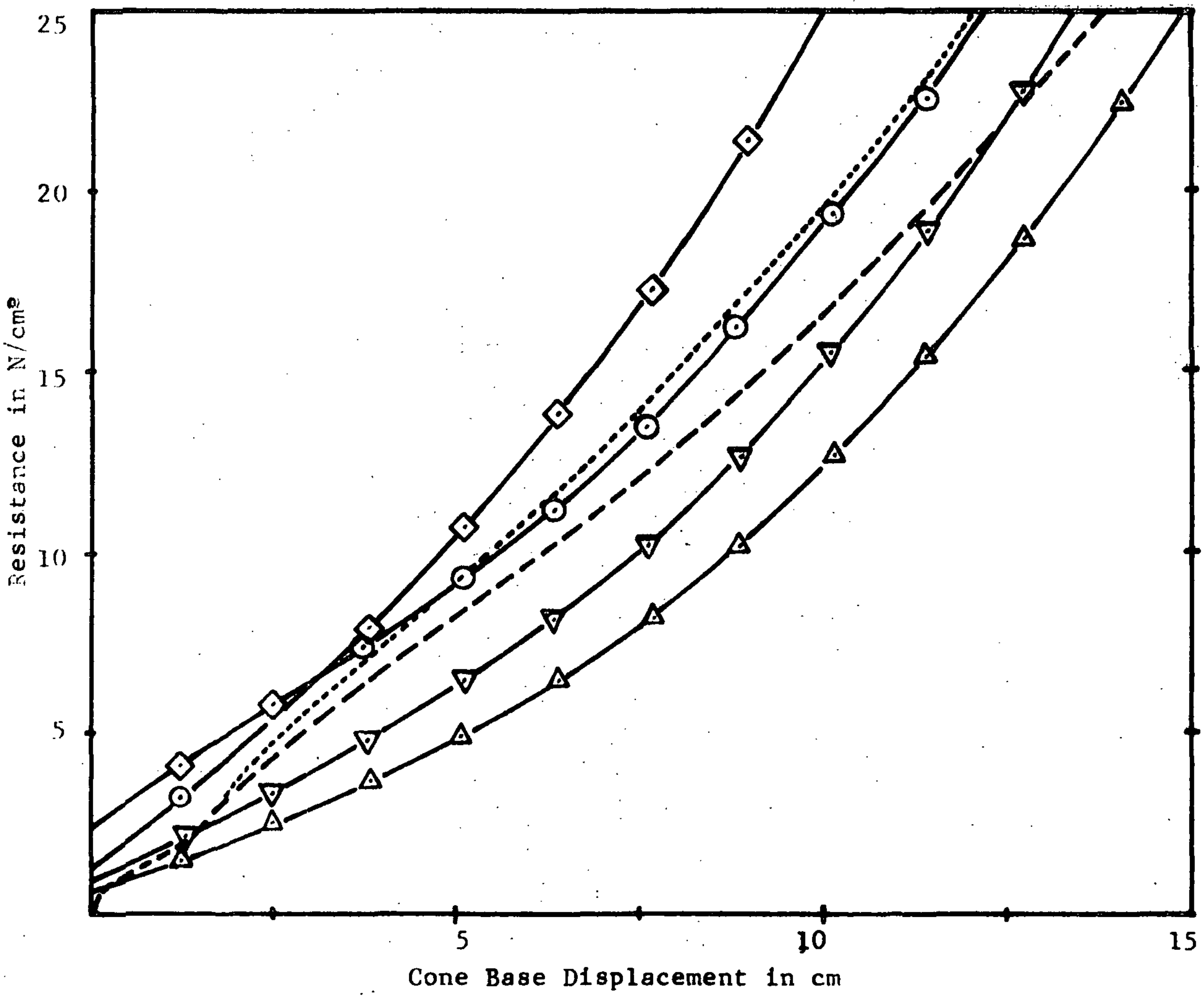


Figure 7: Force-Displacement Curves for Cone Penetrometer

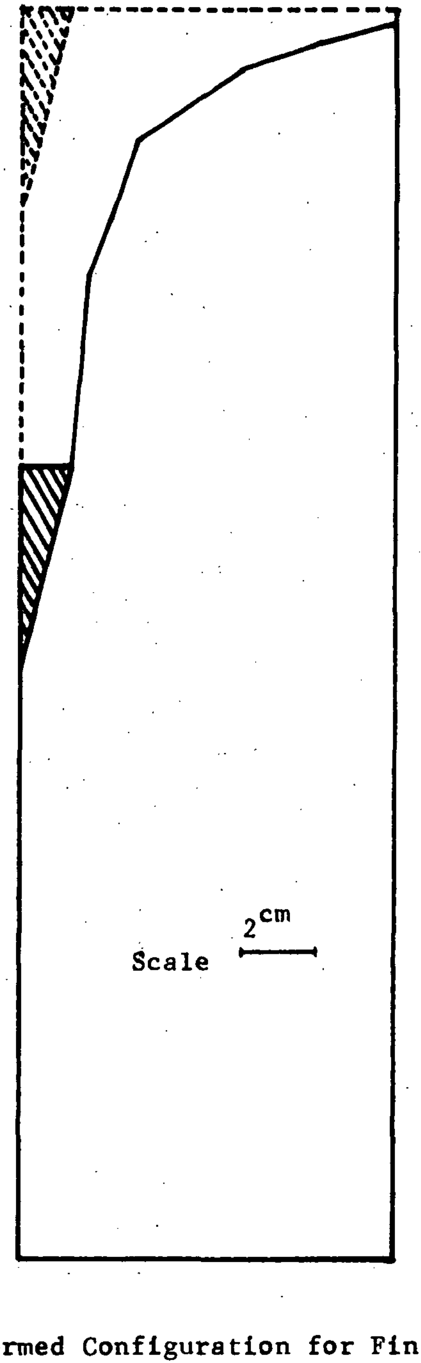


Figure 8: Deformed Configuration for Finite Element Solution (1) at F = 25 N/cm^2

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 paramenter 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, 1050, 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)

```
@ELT,IDL DECK,,SYSTOP
ELT 005-11/24-14:39
@ELT,SIH NASA*TPF$.MAIN,,,132656133010
C-----------------------------------------------------------------------
C
C          THE FINITE ELEMENT ANALISYS OF AXISYMMETRIC SOIL MEDIUM
C                        BY A SOIL PLASTICITY THEORY
C
C-----------------------------------------------------------------------
C
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,NCYCL ,EPSLON
      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
     * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL
      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DEI,DE2
      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST
      COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP
      COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS,
     1 FINC,FN,ULOAD,FEL,PMAX,DLMAX
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
      COMMON /BLK7/ DSTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS),
     1 AOJ(NELS)
      COMMON/BLK8/INCR,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,PP
      COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRIGD
      COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20),
     1 POR(20),DELTS,EC,XNUC,DELD
      COMMON /BLK11/ VOID(NELS),DGAM(NELS,4)
      COMMON /BLK12/ SIGMX(NELS),DEP(NELS),EP(NELS),DEQST2(NELS),ES
      COMMON /GMTRY/ RO(NODS),ZO(NODS)
C-----------------------------------------------------------------------
C
      NTAPE = 2
C
C-----------------------------------------------------------------------
C
      CALL SETUP
C
C           INITIALIZES NECESSARY CONSTANTS FOR INTEGRATION SCHEME.
C-----------------------------------------------------------------------
C
      CALL INPUT(NTAPE)
      ISHEAR = 1
C
C
      FN = FEL
C
C-----------------------------------------------------------------------
C
C           START MAIN ITERATION LOOP.
C
C-----------------------------------------------------------------------
      DO 990 NI = 1,NINCR
C
      FN = FN + FINC
```

```
000        ITER = 0
000  C
000        IF(NI.EQ.1) GO TO 950
000        ISHEAR = NI
000        CALL ZERO(XK,NF,1)
000        REWIND NTAPE
000  C
000        DO 940 NEL = 1,NELEM
000         IC = IJKL(NEL,1)
000         JC = IJKL(NEL,2)
000         KC = IJKL(NEL,3)
000         LC = IJKL(NEL,4)
000        NFT = NEL - NRIGD
000        IF(NEL.LE.NRIGD) GO TO 938
000        SIGZ = STRS(NEL,1) + DSTRS(NEL,1) / 2.
000        SIGR = STRS(NEL,2) + DSTRS(NEL,2) / 2.
000        SIGT = STRS(NEL,3) + DSTRS(NEL,3) / 2.
000        TAUZR= STRS(NEL,4) + DSTRS(NEL,4) / 2.
000        IF(NBC.NE.0.AND.NFT.LE.NBC) GO TO 899
000  C------------------------------------------------------------------
000  C
000        CALL DMATRX(NI)
000  C
000  C           CALCULATE STRESS DEPENDENT MATERIAL PROPERTY MATRIX (D).
000  C------------------------------------------------------------------
000  C
000        GO TO 837
000  C
000    938 DO 400 I = 1,4
000        DO 400 J = 1,4
000        DMAT(NEL,I,J) = DE(NEL,I,J)
000    400 D(I,J) = DMAT(NEL,I,J)
000    837 CALL STIFF2(NI,NTAPE)
000        GO TO 838
000  C
000    899 CALL FRICTN(IC,KC,NEL,NFT,ISHEAR,VOIDI)
000  C
000    838 CALL ASSEMB(NEL,NFT)
000    940 CONTINUE
000    950 CONTINUE
000        ITER = ITER + 1
000        INCR = ITER
000  C
000        CALL DISPL(NFREE,NI,INODE,INCR)
000  C
000        DO 340 I = 1,INODE
000        JJ = I * 2
000        II = JJ - 1
000        Z(I) = Z0(I) + TOTDIS(II) + XK(LAST+II)
000    340 R(I) = R0(I) + TOTDIS(JJ) + XK(LAST+JJ)
000  C
000        CALL STRAIN(NI,ITER)
000  C
000  C        SUMMING OF STRESSES AND DISPLACEMENTS FOR EACH INCREMENTAL STEP
000        DO 310 ITT = 1,NFREE
000    310 TOTDIS(ITT) = TOTDIS(ITT) + XK(ITT+LAST)
000        DO 329 I = 1,NELEM
```

```
      DO 329 J = 1,4
  329 STRS(I,J) = STRS(I,J) + DSTRS(I,J)
C
      WRITE(6,689) NINCR,NI,FN
      DO 320 I = 1,INODE
       II = (I-1) * 2 + 1
       JJ = II + 1
  320 WRITE(6,690) I,(TOTDIS(LX),LX = II,JJ)
      WRITE(6,691)
      DO 330 I = 1,NELEM
  330 WRITE(6,692) I,(STRS(I,J),J=1,4)
      IF(FN.GT.PMAX) STOP LOADMX
  990 CONTINUE
C
C
  500 FORMAT(10I5)
  600 FORMAT(//'  ITER',I5,'  DMAX',E12.5,'  DE2',E12.5)
  689 FORMAT(1H1,10X,'  TOTAL DISPLACEMENT     NO. OF INCREMENTAL STEPS
     *=',2I5//5X,'NODE',5X,'Z - DISPL',20X,'R - DISPL',5X,'FN =',E12.5/)
  690 FORMAT(4X,I5,E15.7,10X,E15.7)
  691 FORMAT(///,10X,' TOTAL STRESSES'//5X,'ELEM',5X,'SIGMA - Z',T28,
     1'SIGMA - R',T42,'TANGENTIAL',T58,'TAU - ZR')
  692 FORMAT(I8,4F14.6)
  693 FORMAT(//'     TAU, SIG, RAT, DELTS, ISHEAR, ',4E12.5,I5//)
  694 FORMAT(//20X,'NEW GEOMETRY AT THE END OF LOAD INCR.',I5//)
  695 FORMAT(I10,2F15.6)
      STOP
      END
@ELT,SIH NASA*TPF$.INPUT,,,132661133010
      SUBROUTINE INPUT(NTAPE)
C-----------------------------------------------------------------------
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C-----------------------------------------------------------------------
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,NCYCL ,EPSLON
      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
     * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL
      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2
      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST
      COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP
      COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS,
     1 FINC,FN,ULOAD,FEL,PMAX,DLMAX
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
      COMMON/BLK8/INCR,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,XMS
      COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRIGD
      COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20),
     1 POR(20),DELTS,EC,XNUC,DELD
      COMMON /BLK11/ VOID(NELS),DGAM(NELS,4)
      COMMON /BLK12/ SIGMX(NELS),DEP(NELS),EP(NELS),DEQST2(NELS),ES
      COMMON /GMTRY/ RO(NODS),ZO(NODS)
C-----------------------------------------------------------------------
      REWIND NTAPE
      READ(5,510) (TITLE(I),I=1,20)
      READ(5,500) INODE,NELEM,NAPC,NBC,NINCR,NCYCL,ICASE,NRIGD,NULOAD
      READ(5,530) YSTRS,DELL,ZETA,PMAX,DLMAX
      READ(5,511) DZI,DRI
```

```
000   C      DZI = SHEAR MOD. FOR INTERFACE ELEMENTS.
000   C      DRI = NORMAL     FOR INTERFACE ELEMENTS.
000         READ(5,511) EC,XNUC,ES,XNUS
000         READ(5,530) PI,SMALLK,XI,VOIDI,PO,DEPTH,ALAMDA,EPSLON
000         WRITE(6,530)PI,SMALLK,XI,VOIDI,PO,DEPTH,ALAMDA,EPSLON
000         WRITE(6,630) YSTRS,DELL
000         WRITE(6,631) ZETA,DZI,DRI
000     631 FORMAT(//'     ZETA, MOD. FOR INTERFACE ELEM., SHEAR, NORMAL,
000        1  3F15.6)
000         POP = EXP(1.-SMALLK/ALAMDA)
000         II = 0
000         DO 101 I = 1,INODE
000         DO 101 J = 1,2
000          II = II + 1
000     101 ID(I,J) = II
000         BETA=ALAMDA-SMALLK
000         PI=PI*3.14159/180.
000         EPSLON = EPSLON * 3.14159 / 180.
000         DELTS = TAN(EPSLON)
000         PI=SIN(PI)
000   C        PI IS SINE(PI)
000         XM = 6.*PI/(3.-PI)
000         XMS = XM * XM
000         WRITE(6,600) (TITLE(I),I=1,20)
000         DO 100 I = 1,INODE
000         READ(5,520) Z(I),R(I),IZ,IR
000         ZO(I) = Z(I)
000         RO(I) = R(I)
000         IF(IZ.NE.0) ID(I,1) = 0
000         IF(IR.NE.0) ID(I,2) = 0
000     100 WRITE(6,620) I,Z(I),R(I) ,IZ,IR
000         WRITE(6,501)  INODE,NELEM,NAPC,NINCR,NCYCL
000         NFREE = INODE * 2
000   C
000         WRITE(6,651)
000         READ(5,540) ((IJKL(NEL,J),J=1,4),NEL=1,NELEM)
000         WRITE(6,650) (NEL,(IJKL(NEL,J),J=1,4),NEL=1,NELEM)
000   C
000   C     FIND HALF BAND WIDTH AND ACTUAL SIZE OF MATRIX (XK)
000   C
000         IMAX = 0
000         DO 800 NEL = 1,NELEM
000         DO 700 I = 1,4
000         IN = IJKL(NEL,I)
000         KK(NEL,I) = ID(IN,1)
000     700 KK(NEL,I+4) = ID(IN,2)
000         DO 7999 I = 1,2
000         II = I + 1
000         DO 7999 J = II,4
000         IDIF = IJKL(NEL,I) - IJKL(NEL,J)
000         IF(IDIF.LT.0) IDIF = -IDIF
000    7999 IF(IDIF.GT.IMAX) IMAX = IDIF
000     800 CONTINUE
000         IF(NBC.NE.0) READ(5,500) ((ID(I,J),J=1,2),I=1,NBC)
000   C     IMAX = MAX DIFFERENCE IN ADJACENT NODE NO.
000         IHB = (IMAX + 1) * 2
000         IHBI = IHB - 1
```

```
000          LT = IHB * IHBI / 2                                          00008600
000          LAST = LT + (NFREE - IHBI) * IHB                             00008700
000          WRITE(6,640) IMAX,IHB,LT,LAST                                00008800
000    C-----------------------------------------------------------------00008900
000          IF(NRIGD.NE.0) CALL ELASTC(D,NRIGD,EC,XNUC)                  00009000
000    C-----------------------------------------------------------------00009100
000          NLST = NELEM - NRIGD - NBC                                   00009200
000          DO 900 NEL = 1,NELEM                                         00009300
000          IC = IJKL(NEL,1)                                             00009400
000          JC = IJKL(NEL,2)                                             00009500
000          KC = IJKL(NEL,3)                                             00009600
000          LC = IJKL(NEL,4)                                             00009700
000          IF(NEL.LE.NRIGD)  GO TO 898                                  00009800
000          NFT = NEL - NRIGD                                            00009900
000          IF(NFT.GT.NBC) GO TO 896                                     00010000
000          DZZ(NFT) = DZI                                               00010100
000          DRR(NFT) = DRI                                               00010200
000          CALL FRICTN(IC,KC,NEL,NFT,0,VOIDI)                           00010300
000          GO TO 899                                                    00010400
000      896 IF(ICASE. NE. 0) GO TO 897                                   00010500
000    C                                                                  00010600
000          CALL AREAA(IC,JC,KC,LC,AREA)                                 00010700
000          VOIDR = VOIDI                                                00010800
000          VOID(NEL) = VOIDR                                            00010900
000          DELNM1(NEL) = AREA                                           00011000
000    C                                                                  00011100
000      897 NELST = NRIGD + NBC + 1                                      00011200
000          IF(NEL.EQ.NELST)CALL ELASTC(D,NLST,ES,XNUS)                  00011300
000    C                                                                  00011400
000      898 DO 111 I = 1,4                                               00011500
000          DO 111 J = 1,4                                               00011600
000          DE(NEL,I,J) = D(I,J)                                         00011700
000      111 DMAT(NEL,I,J) = D(I,J)                                       00011800
000    C                                                                  00011900
000          CALL STIFF1(NTAPE)                                           00012000
000          CALL STIFF2(0,NTAPE)                                         00012100
000      899 CALL ASSEMB(NEL,NFT)                                         00012200
000    C                                                                  00012300
000    C-----------------------------------------------------------------00012400
000      900 CONTINUE                                                     00012500
000          IF(NAPC.NE.0) CALL PTLOAD(NAPC,ULOAD)                        00012600
000      901 IF(NULOAD.NE.0) CALL EQLOAD(ULOAD)                           00012700
000    C                                                                  00012800
000          SCAL = NINCR                                                 00012900
000          DO 200 I = 1,NFREE                                           00013000
000      200 APF(I) = APF(I) / SCAL                                       00013100
000          FEL = 0.                                                     00013200
000          FINC = ULOAD / SCAL                                          00013300
000    C                                                                  00013400
000      500 FORMAT(10I5)                                                 00013500
000      501 FORMAT(//'     NUMBER OF NODES =',I5/'     NUMBER OF ELEMENTS =',I00013600
000         15/'     NUMBER OF APLIED CONCENTRATED LOADS =',I5/'     NUMBER OF 00013700
000         2INCREMENTAL LOAD STEPS =',I5/'     NUMBER OF ITERATIONS PER EACH I00013800
000         3NCREMENTAL LOADING =',I5/)                                   00013900
000      510 FORMAT(20A4)                                                 00014000
000      511 FORMAT(4F20.5)                                               00014100
000      520 FORMAT(2F10.4,2I5)                                           00014200
```

```
  530 FORMAT(8F10.4)
  531 FORMAT(/'     SINE PHI =',F10.4,'     SMALLK =',F10.4,'     KI =',
     *F10.4,'     POROSITY =',F10.4/'     BETA =',F10.4,'     OVERBURD
     *EN PRESSURE =',F10.4,'     DESIRED ACCURACY =',F10.4,' PERCENT.'/
     * '     DEPTH OF SOIL MEDIA =',F10.4/)
  540 FORMAT(4I5)
  600 FORMAT(1H1,20X,20A4///30X,'COORDINATE VALUES'//T11,'NODE',T30,'Z-C
     *OORD',T50,'R-COORD',T65,'0,IF FREE TO Z',3X,'0,IF FREE TO R'//)
  620 FORMAT(T10,I5,T25,F10.4,T45,F10.4,T65,2(I8,5X))
  630 FORMAT(//'     YIELD STRESS =',E12.5,'     DELL =',E12.5//)
  640 FORMAT(//'   IMAX,IHB,LT,LAST',4I10//)
  650 FORMAT(5I7)
  651 FORMAT(1H1,10X,'CNNECTIVITY'/)
      RETURN
      END
WELT,SIH NASA*TPF$.DMATRX,,,132671133010
      SUBROUTINE DMATRX(NI)
C-----------------------------------------------------------------------
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C-----------------------------------------------------------------------
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON
      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
     * TYPEF(4,4),TYPEG(4,4),A0,B0,C0,RT,RB,RA,RC,IC,JC,KC,LC,NEL
      COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP
      COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS,
     1 FINC,FN,ULOAD,FEL,PMAX,DLMAX
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
      COMMON /BLK7/ USTRS(NELS,4),DUM(NELS,10)
      COMMON/BLK8/INCR,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,XMS
      COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRIGD
      COMMON /BLK11/ VOID(NELS),DGAM(NELS,4)
      COMMON/BLK12/ SIGMX(NELS),DEP(NELS),EP(NELS),DEQST2(NELS),ES
      DIMENSION        RLB(4),DB(4),RD(4)
C-----------------------------------------------------------------------
      VOIDR = VOID(NEL)
      P = (SIGZ + SIGT + SIGR) / 3.
      TJ = ((SIGZ-SIGR)**2+(SIGR-SIGT)**2+(SIGT-SIGZ)**2)/6.+TAUZR**2
      PSQ = P*P
      ETS = 3.*TJ/PSQ
      POW = 1.-SMALLK/ALAMDA
      SIGMX(NEL) = P* ((XMS+ETS) /XMS) ** POW
      POP = SIGMX(NEL)
      AA = XMS * (2.*P-POP) / 3.
      BB = 3. * AA * XMS * P * POP * (1.+VOIDR) / BETA
      DO 100 I = 1,4
      DO 100 J = 1,4
  100 D(I,J) = DE(NEL,I,J)
      SZZ = 2.*SIGZ-SIGR-SIGT
      SRR = 2.*SIGR-SIGZ-SIGT
      STT = 2.*SIGT-SIGZ-SIGR
      SZR = 6.*TAUZR
      RLB(1) = SZZ + AA
      RLB(2) = SRR + AA
      RLB(3) = STT + AA
      RLB(4) = SZR
      DELP = (DSTRS(NEL,1)+DSTRS(NEL,2)+DSTRS(NEL,3)) / 3.
      DB(1) = SZZ
      DB(2) = SRR
      DB(3) = STT
      DB(4) = SZR
      DFI = 0.
      DFK = 0.
      DO 200 I = 1,4
      DFI = DFI + RLB(I) * DSTRS(NEL,I)
  200 DFK = DFK + DB(I) * DSTRS(NEL,I)
      DFJ = DELP * POP * XMS
      POW = -SMALLK / ALAMDA
      DEN = (1.+ETS/XMS) ** POW
      DFK = DEN * (DFK-6.*TJ/P*DELP)
      DF = DFI - DFJ - DFK * (1.-SMALLK/ALAMDA)
      ASQ = XMS * P * (POP-P)
      TTJ = TJ * 3.
      SIGBA(NEL) = ASQ
      EP(NEL) = DF
      WRITE(6,620) NEL,VOIDR,TTJ,ASQ,POP,DF,XMS,ETS
  620 FORMAT(I5,'  VOIDR=',F10.4,'  3J=',F10.4,'  ASQ=',F10.4,'  PO=',
     1F10.4,'  DF=',E12.5,'  XMS=',E12.5,'  ETS=',E12.5)
C
      IF(DF.LT.0.) GO TO 764
C
      DO 110 I = 1,4
      DB(I) = 0.
      RD(I) = 0.
      DO 110 J = 1,4
      DB(I) = DB(I) + D(I,J) * RLB(J)
  110 RD(I) = RD(I) + RLB(J) * D(J,I)
      DEN = 0.
      DO 120 I = 1,4
  120 DEN = DEN + RLB(I) * DB(I)
      DEN = DEN + BB
      DO 130 I = 1,4
      DO 130 J = 1,4
  130 D(I,J) = D(I,J) - DB(I) * RD(J) / DEN
  764 DO 111 I = 1,4
      DO 111 J = 1,4
  111 DMAT(NEL,I,J) = D(I,J)
  600 FORMAT(4E20.7)
      RETURN
      END
@ELT,SIH NASA*TPF$.STIFF1,,,113762121110
      SUBROUTINE STIFF1(NTAPE)
C-----------------------------------------------------------------------
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C-----------------------------------------------------------------------
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON
      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
     * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
      COMMON /BLK7/ DSTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS),
     1 AOJ(NELS)
```

```
000          COMMON /BLKA/M,N
000     C-----------------------------------------------------------------------
000     C
000     C        DIFFERENT TYPE OF INTEGRATIONS ARE PERFORMED AND SAVED ON FILE
000     C         NO.2 FOR LATER USE.
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          BZ(1) = Z(KC) - Z(LC)
000          BZ(2) = -BZ(1)
000          BZ(3) = Z(JC) - Z(IC)
000          BZ(4) = -BZ(3)
000          CZ(1) = Z(KC) - Z(JC)
000          CZ(2) = Z(IC) - Z(LC)
000          CZ(3) = -CZ(2)
000          CZ(4) = -CZ(1)
000          AR(1) = R(JC) - R(LC)
000          AR(2) = R(KC) - R(IC)
000          AR(3) = -AR(1)
000          AR(4) = -AR(2)
000          BR(1) = R(LC) - R(KC)
000          BR(2) = -BR(1)
000          BR(3) = R(IC) - R(JC)
000          BR(4) = -BR(3)
000          CR(1) = R(JC) - R(KC)
000          CR(2) = R(LC) - R(IC)
000          CR(3) = -CR(2)
000          CR(4) = -CR(1)
000          AO =-AR(3)*AZ(2) + AR(4)*AZ(1)
000          BO =-BR(2)*BZ(4) + BR(3)*BZ(1)
000          CO = CR(3)*CZ(1) - CR(4)*CZ(2)
000     C
000          RT = R(IC) + R(JC) + R(KC) + R(LC)
000          RA =-R(KC) + R(IC) - R(LC) + R(JC)
000          RB = R(JC) - R(IC) + R(KC) - R(LC)
000          RC =-R(IC) + R(JC) - R(KC) + R(LC)
000          AOJ(NEL) = AO
000          RTT(NEL) = RT
000     C
000          DO 200 M = 1,4
000          ARM(NEL,M) = AR(M)
000          AZM(NEL,M) = AZ(M)
000          DO 200 N = 1,4
000          CALL GAUSS(1,AA)
000          CALL GAUSS(2,BB)
000          CALL GAUSS(3,CC)
000          CALL GAUSS(4,EE)
000          CALL GAUSS(5,FF)
000          CALL GAUSS(6,GG)
000          TYPEA(M,N) = AA
000          TYPEB(M,N) = BB
000          TYPEC(M,N) = CC
000          TYPEE(M,N) = EE
000          TYPEF(M,N) = FF
000          TYPEG(M,N) = GG
```

```
  200 CONTINUE
      WRITE(NTAPE) TYPEA,TYPEB,TYPEC,      TYPEE,TYPEF,TYPEG
      RETURN
      END
WELT,SIH NASA*TPF$.STIFF2,,,113771121110
      SUBROUTINE STIFF2(NI,NTAPE)
C-----------------------------------------------------------------------
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C-----------------------------------------------------------------------
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON
      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
     * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
C-----------------------------------------------------------------------
C
C          FORM STIFFNESS MATRIX.
C
      IF(NI.NE.0)
     1READ (NTAPE) TYPEA,TYPER,TYPEC,      TYPEE,TYPEF,TYPEG
      DO 200 I = 1,4
      DO 200 J = 1,4
C
      STIFF(I,J) = TYPEA(I,J)*D(1,1)/8.+TYPEB(I,J)*D(4,1)/ 8.+
     1 TYPER(J,I)*D(1,4)/ 8.+TYPEE(I,J)*D(4,4)/ 8.
      STIFF(J+4,I) = TYPER(J,I)*D(2,1)/8.+TYPEC(J,I)*D(3,1)+TYPEA(J,I)*
     1 D(4,1)/ 8.+TYPEE(J,I)*D(2,4)/ 8.+TYPEF(J,I)*D(3,4)*2+TYPEB(I,J)*
     2 D(4,4)/ 8.
      STIFF(I,J+4) = STIFF(J+4,I)
      STIFF(I+4,J+4) = TYPEE(I,J)*D(2,2)/8.+(TYPEB(I,J)+TYPEB(J,I))*
     1 D(2,4)/ 8.+TYPEA(I,J)*D(4,4)/ 8.+2.*(TYPEF(I,J)+TYPEF(J,I))*
     2 D(2,3)+(TYPEC(I,J)+TYPEC(J,I))*D(3,4)    +TYPEG(I,J)*D(3,3)
  200 CONTINUE
      RETURN
      END
WELT,SIH NASA*TPF$.FRICTN,,,132675133010
      SUBROUTINE FRICTN(IC,KC,NEL,NFT,ISHEAR,VOIDI)
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2
      COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP
      COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS,
     1 FINC,FN,ULOAD,FEE,PMAX,DLMAX
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
      COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRIGD
      COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20),
     1 POR(20),DELTS,EC,XNUC,DELD
      DIMENSION         TS(8,8)
      DZ = DZZ(NFT)
      DR = DRR(NFT)
      PPI = 3.14159
      RB = (R(IC)+R(KC)) / 2.
      PIR = PPI * RB / 3.
      BASE = R(KC) - R(IC)
      HIGH = Z(IC) - Z(KC)
      XL = (BASE*BASE + HIGH*HIGH)**.5
```

```
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 = ID(NFT,1)                                                      00003900
000531      000          JJ = ID(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,I) = 2.*DIIF                                                00005600
000548      000          STIFF(I+4,I+4) = 2.*DJJF                                            00005700
000549      000          STIFF(I,I+4) = 2.*DIJF                                              00005800
000550      000      190 STIFF(I,9-I) = DIJF                                                 00005900
000551      000          DO 191 I = 1,2                                                      00006000
000552      000          STIFF(I,I+2) = -DIIF                                                00006100
000553      000          STIFF(I,5-I) = DIIF                                                 00006200
000554      000          STIFF(I,7-I) = -2.*DIJF                                             00006300
000555      000          STIFF(I,I+6) = -DIJF                                                00006400
000556      000          STIFF(I+4,I+6) = -DJJF                                              00006500
000557      000          STIFF(I+4,9-I) = DJJF                                               00006600
000558      000          STIFF(I+2,I+4) = -DIJF                                              00006700
000559      000          STIFF(I+2,9-I) = -2.*DIJF                                           00006800
000560      000          STIFF(2*I-1,2*I) = -2.*DIIF                                         00006900
000561      000      191 STIFF(2*I+3,2*I+4) = -2.*DJJF                                       00007000
000562      000          DO 210 I = 1,8                                                      00007100
000563      000          DO 210 J = I,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
```

```
  630 FORMAT('   NEL   DZ   DR   DII   DIJ   DJJ   LENGTH',I5,6E12.5)
      WRITE(6,610) STIFF
  610 FORMAT(8E15.6)
      DO 140 I = 1,8
      DO 140 J = 1,8
      SUM = 0.
      DO 144 K = 1,8
  144 SUM = SUM + TR(NFT,K,I) * STIFF(K,J)
  140 TS(I,J) = SUM
      DO 150 I = 1,8
      DO 150 J = 1,8
      SUM = 0.
      DO 155 K = 1,8
  155 SUM = SUM + TS(I,K) * TR(NFT,K,J)
  150 STIFF(I,J) = SUM
      DO 200 I = 1,8
      DO 200 J = 1,8
  200 FRCK(NFT,I,J) = STIFF(I,J)
      WRITE(6,610) STIFF
      RETURN
      END
@ELT,SIH NASA*TPF$.ASSEMB,,,114006121110
      SUBROUTINE ASSEMB(NEL,NFT)
C-----------------------------------------------------------------------
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C-----------------------------------------------------------------------
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON
      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2
      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
      COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20),
     1 POR(20),DELTS,EC,XNUC,DELD
C-----------------------------------------------------------------------
C
       NFREE = INODE * 2
      IF(NFT.LT.1.OR.NFT.GT.NBC) GO TO 120
      DO 130 I = 1,8
      DO 130 J = 1,8
  130 STIFF(I,J) = FRCK(NFT,I,J)
  120 CONTINUE
C
      DO 110 I = 1,8
      II = KK(NEL,I)
      DO 110 J = 1,8
      JJ = KK(NEL,J)
      IF(II.EQ.0.OR.JJ.EQ.0) GO TO 110
      IF(II.LT.JJ) GO TO 110
      IF(II.GT.IHBI) GO TO 104
      L = JJ + (II-1) * II / 2
      GO TO 105
  104 L = JJ + LT + (II-IHB) * IHBI
  105 XK(L) = XK(L) + STIFF(I,J)
  110 CONTINUE
      RETURN
      END
@ELT,SIH NASA*TPF$.DISPL,,,114011121110
```

```
      SUBROUTINE DISPL(NFREE,NI,INODE,INCR)
C-----------------------------------------------------------------------
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C-----------------------------------------------------------------------
      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2
      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST
      COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
     1 R(NODS),Z(NODS),TOTDIS(MAX)
C-----------------------------------------------------------------------
C
      N = NFREE
      ZTEST = 0.000001
C
      DO 100 J = 1,NFREE
      IF(J.GT.IHBI) GO TO 108
      L = (J+1) * J / 2
      GO TO 109
  108 L = LT + IHB * (J - IHBI)
  109 XTEST = ABS(XK(L))
      IF(XTEST.LT.ZTEST) XK(L) = 1.
  100 CONTINUE
C
      DO 110 I = 1,NFREE
  110 XK(LAST+I) = APF(I)
C
C
      CALL FACTOR(NFREE)
C
C
      CALL SOLTN(NFREE)
C
C
      WRITE(6,600) INCR,NI
      NN = 10
      DO 280 J = 1,NN
        II = (J-1) * 2 + 1 + LAST
        JJ = II + 1
  280 WRITE(6,610) J,XK(II),XK(JJ)
  600 FORMAT(/// 20X,'DISPLACEMENTS FOR CYCLE NO.',I4//18X,'Z - DISPL',
     * 15X,'R - DISPL',15X,'LOAD INCREMENT STEP =',I5//)
  610 FORMAT(I7,2E20.7)
      RETURN
      END
@ELT,SIH NASA*TPF$.FACTOR,,,114013121110
      SUBROUTINE FACTOR(NFREE)
C     THIS SUBROUTINE PERFORMS FACTORING
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST
      N = NFREE
      IHB1 = IHB1
      DO 8 I=1,N
      IF(I.GT.IHB1) GO TO 2
      K=1
      M=K+(I-1)*I/2
      GO TO 3
    2 K=I-IHB1
      M=K+LT+(I-IHB)*IHB1
```

```
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            B = 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     @ELT,SIH NASA*TPF$.SOLTN,,,114016121110
000726     000           SUBROUTINE SOLTN(NFREE)                                              00000100
000727     000           PARAMETER NODS=300,NELS=260,NT=20000,MAX=600                         00000200
000728     000           COMMON /BLK3/ XK(NT),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 = IHB1                                                          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
```

```
      M1=MM*K/2
      GO TO 3
    2 M=K-IHB
      MM=IHB1
      M1=M*IHB1+LT
    3 SUM=0.0
      DO 4 L=1,MM
      LL=L+M
      JJ=LL+M1
      LL=LL+NF-1
    4 SUM=SUM+XK(JJ)*XK(LL)
    1 XK(LL+1)=XK(LL+1)-SUM
      J = NF+N-1
C     THIS PORTION OF SUBROUTINE PERFORMS BACK-SUBSTITUTION
      NF=NF+N-1
      XK(NF)=XK(NF)/XK(LAST)
      DO 5 K=2,N
      L=N-K+1
      IF(L.GT.IHB1) GO TO 6
      I=L+(L-1)*L/2
      GO TO 7
    6 I=L+(L-IHB)*IHB1+LT
   7  IR=N-IHB
      IF(L.GT.IR) GO TO 8
      J=IHB1
      GO TO 9
    8 J=K-1
    9 SUM=0.0
      DO 10 M=1,J
      MM=L+M
      IF(MM.GT.IHB1) GO TO 11
      NN=L+(MM-1)*MM/2
      GO TO 12
   11 NN=L+(MM-IHB)*IHB1+LT
   12 MM=NF-N+MM
   10 SUM=SUM+XK(NN)*XK(MM)
      MM=NF-N+L
    5 XK(MM)=XK(MM)/XK(1)-SUM
      RETURN
      END
@ELT,SIH NASA*TPF$.STRAIN,,,132705133010
      SUBROUTINE STRAIN(NI,INCR)
C-----------------------------------------------------------------------
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
C-----------------------------------------------------------------------
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,IPRINT,EPSLON
      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2
      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHB1,LT,LAST
      COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP
      COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS,
     1 FINC,FN,ULOAD,FEL,PMAX,DLMAX
      COMMON /BLK7/ DSTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS),
     1 ADJ(NELS)
      COMMON /BLK8/ INCNN,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,XMS
      COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NEREE,NELST,ICASE,NRIGD
      COMMON /BLK11/ VOID(NELS),DGAM(NELS,4)
      COMMON /BLK12/ SIGMX(NELS),DEP(NELS),EP(NELS),DEQST2(NELS),ES
```

```
000       DIMENSION          U(4),V(4),EPS(4),SIG(4)
000   C
000   C
000         SUMA = 0.
000         SUMB = 0.
000         IF(NEC.NE.0) CALL INTFAC(DEI,NI)
000         WRITE(6,666) NI,INCR
000     666 FORMAT(///10X,'STRAINS AND STRESSES FOR INCREMENT NO. =',I4,
000        * '  NO. OF CYCLES = ',I4,//5X,'ELEM',T14,'SIG-Z',T30,'SIG-R',T45,
000        1'SIG-T',T60,'TAU-ZR',T75,'AREA',T90,'VOID RATIO')
000         DO 900 NEL = 1, NELEM
000         NFT = NEL - NRIGD
000         IF(NFT.GT.0.AND.NFT.LE.NBC) GO TO 900
000         DO 100 I = 1,4
000         IN = IJKL(NEL,I)
000         II = (IN-1)*2 + LAST + 1
000         JJ = II + 1
000         V(I) = XK(II)
000     100 U(I) = XK(JJ)
000   C        FIND STRAINS AND SRESSES AT THE CENTROID.
000          EZ = 0.
000          ER = 0.
000          GM = 0.
000          SUM = 0.
000         DO 111 I = 1,4
000         EZ = EZ + ARM(NEL,I) * V(I)
000         ER = ER + AZM(NEL,I) * U(I)
000         GM = GM + ARM(NEL,I) * U(I) + AZM(NEL,I) * V(I)
000     111 SUM = SUM + U(I)
000   C        COMPRESSION POSITIVE.
000         EPS(1) = -EZ / AOJ(NEL)
000         EPS(2) = -ER / AOJ(NEL)
000         EPS(3) = -SUM / RIT(NEL)
000         EPS(4) = -GM / AOJ(NEL)
000   C        EPS(1) = STRAIN IN Z - DIRECTION.
000   C        EPS(2) = STRAIN IN R - DIRECTION.
000   C        EPS(3) = TANGENTIAL STRAIN.
000   C        EPS(4) = SHEAR STRAIN
000         DO 200 I = 1,4
000         DGAM(NEL,I) = EPS(I)
000          SIG(I) = 0.
000         DO 200 J = 1,4
000         SIG(I) = SIG(I) + DMAT(NEL,I,J) * EPS(J)
000     200 CONTINUE
000         III=NEL
000         DO 210 I = 1,4
000         DSTRS(III,I)=SIG(I)
000     210 CONTINUE
000         IF(NEL.LE.NRIGD) GO TO 890
000         CALL AREAA(IJKL(NEL,1),IJKL(NEL,2),IJKL(NEL,3),IJKL(NEL,4),AREA)
000         RATE = AREA / DELNM1(NEL)
000         VOID(NEL) = RATE * (1.+VOIDI) - 1.
000     890 CONTINUE
000         WRITE(6,600) NEL,SIG,AREA,VOID(NEL)
000     900 CONTINUE
000   C
000     600 FORMAT(I9,7F15.6)
```

```
  620 FORMAT(I10,2F15.6)
      RETURN
      END
@ELT,SIH NASA*TPF$.INTFAC,,,132711133010
      SUBROUTINE INTFAC(BEI,NI)
      PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,NINCR,NCYCL ,EPSLON
      COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2
      COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST
      COMMON /BLK4/ STRS(NELS,4),DMAT(NELS,4,4),DELNM1(MAX),POP
      COMMON /BLK5/ DE(NELS,4,4),SIGBA(NELS),DSIGBA(NELS),DELL,YSTRS,
     1 FINC,FN,ULOAD,FEL,PMAX,DLMAX
      COMMON /BLK7/ DSTRS(NELS,4),ARM(NELS,4),AZM(NELS,4),RTT(NELS),
     1 ADJ(NELS)
      COMMON/BLK8/INCR,PDEPTH(NELS),VOIDI,ALAMDA,DEPTH,PP
      COMMON /BLK9/ PI,SMALLK,CK,BETA,PO,NFREE,NELST,ICASE,NRIGD
      COMMON /BLK10/ FRCK(20,8,8),TR(20,8,8),XXL(20),DZZ(20),DRR(20),
     1 POR(20),DELTS,EC,XNUC,DELD
      DIMENSION UG(8),UL(8)
      DE1 = 0.
      DO 900 NEL = 1,NBC
      III = NEL + NRIGD
      DO 100 I = 1,4
      II = IJKL(III,1) * 2  + LAST
      UG(I) = XK(II-1)
  100 UG(I+4) = XK(II)
      DO 110 I = 1,8
      UL(I) = 0.
      DO 110 J = 1,8
  110 UL(I) = UL(I) + TR(NEL,I,J) * UG(J)
      WRITE(6,620) UG
      WRITE(6,620) UL
      EZ = (-UL(1)+UL(2)+UL(3)-UL(4)) / (2.*XXL(NEL))
      ER = (-UL(5)+UL(6)+UL(7)-UL(8)) / (2.*XXL(NEL))
      SIGZ = DMAT(III,1,1) * EZ + DMAT(III,1,2) * ER
      SIGR = DMAT(III,1,2) * EZ + DMAT(III,2,2) * ER
      SIGZ = -SIGZ
      DSTRS(III,1) = SIGZ
      DSTRS(III,2) = SIGR
      NAMELIST/NAME2/ III,SIGZ,SIGR
      WRITE(6,NAME2)
  900 CONTINUE
  620 FORMAT(8F15.5)
      RETURN
      END
@ELT,SIH NASA*TPF$.SETUP,,,114055121110
      SUBROUTINE SETUP
      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
     * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL
        W(1) = .1713244924
        W(2) = .3607615730
        W(3) = .4679139346
        W(4) = W(3)
        W(5) = W(2)
        W(6) = W(1)
        H(1) = .9324695142
        H(2) = .6612093865
        H(3) = .2386191861
        H(4) = -H(3)
        H(5) = -H(2)
        H(6) = -H(1)
      BN(1) = -1.
      BN(2) = 1.
      BN(3) = 1.
      BN(4) = -1.
      CN(1) =  1.
      CN(2) =  1.
      CN(3) = -1.
      CN(4) = -1.
      DN(1) = -1.
      DN(2) =  1.
      DN(3) = -1.
      DN(4) =  1.
      RETURN
      END
@ELT,SIH NASA*TPF$.ELASTC,,,114060121110
      SUBROUTINE ELASTC(D,NRIGD,E,XNU)
      DIMENSION D(4,4)
      WRITE(6,600) NRIGD,E,XNU
      CONST = E*XNU / ((1.+XNU)*(1.-XNU*2.))
      SHEAR = E / (2.*(1.+XNU))
      D(1,1) = CONST + SHEAR*2.
      D(2,2) = D(1,1)
      D(3,3) = D(1,1)
      D(4,4) = SHEAR
      D(1,2) = CONST
      D(1,3) = CONST
      D(2,3) = CONST
      DO 100 I = 1,4
      DO 100 J = I,4
  100 D(J,I) = D(I,J)
  600 FORMAT(//'  FOR FIRST',I4,'  ELEMENTS, THE FOLLOWING MATERIAL PRO
     * PERTIES ARE USED TO FORM ELASTIC MATRIX (D)'/'   E  =',F20.7,
     * '      XNU  =',F10.3//)
      RETURN
      END
@ELT,SIH NASA*TPF$.GAUSS,,,114063121110
      SUBROUTINE GAUSS(IT,AA)
      COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
     * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NEL
      TWOPI = 6.28318531
       IPT = 6
       AA = 0.
      DO 100 I = 1,IPT
       X = H(I)
      DO 100 J = 1,IPT
       Y = H(J)
      AA = AA + W(I) * W(J) * F(X,Y,IT)
  100 CONTINUE
      AA = AA * TWOPI
      RETURN
      END
```

```
000   WELT,SIH NASA*TPF$.F,,,114066121110
000         FUNCTION F(X,Y,IT)
000         COMMON /BLK1/ W(6),H(6),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000        * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPEE(4,4),
000        * TYPEF(4,4),TYPEG(4,4),AO,BO,CO,RT,RB,RA,RC,IC,JC,KC,LC,NFL
000         COMMON /BLKA/M,N
000   C        X STANDS FOR ZHAI IN ZHAI - EITA COORD.
000   C        Y STANDS FOR EITA IN ZHAI - EITA COORD.
000   C        FB = DET. OF JACOBI.
000   C        FC = (N) * (R)
000         FB = AO + BO * X + CO * Y
000         FC = ( RT + RB * X + RA * Y + RC * X * Y) / 4.
000         GO TO (10,20,30,40,50,60),IT
000      10 F = (AR(M)+BR(M)*X+CR(M)*Y) * (AR(N)+BR(N)*X+CR(N)*Y) /FB
000         F = F * FC
000         RETURN
000      20 F = (AZ(M)+BZ(M)*X+CZ(M)*Y) * (AR(N)+BR(N)*X+CR(N)*Y) /FB
000         F = F * FC
000         RETURN
000      30 F = (1.+BN(M)*X+CN(M)*Y+DN(M)*X*Y) * (AR(N)+BR(N)*X+CR(N)*Y) / 32.
000         RETURN
000      40 F = (AZ(M)+BZ(M)*X+CZ(M)*Y) * (AZ(N)+BZ(N)*X+CZ(N)*Y) /FB
000         F = F * FC
000         RETURN
000      50 F = (1.+BN(M)*X+CN(M)*Y+DN(M)*X*Y) * (AZ(N)+BZ(N)*X+CZ(N)*Y) / 64.
000         RETURN
000      60 F = (1.+BN(M)*X+CN(M)*Y+DN(M)*X*Y) *
000        1 (1.+BN(N)*X+CN(N)*Y+DN(N)*X*Y) * FB / (128.*FC)
000         RETURN
000         END
000   WELT,SIH NASA*TPF$.AREAA,,,114072121110
000         SUBROUTINE AREAA(IC,JC,KC,LC,AREA)
000         PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
000         COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),
000        1 R(NODS),Z(NODS),TOTDIS(MAX)
000         AI = (R(JC)-R(IC)) * (Z(LC)-Z(IC)) - (R(LC)-R(IC)) * (Z(JC)-Z(IC))
000         AJ = (R(KC)-R(JC)) * (Z(LC)-Z(JC)) - (R(LC)-R(JC)) * (Z(KC)-Z(JC))
000         IF(AI.LT.0) AI = -AI
000         IF(AJ.LT.0) AJ = -AJ
000         AREA = (AI + AJ) / 2.
000         RETURN
000         END
000   WELT,SIH NASA*TPF$.PTLOAD,,,114074121110
000         SUBROUTINE PTLOAD(NAPC,ULOAD)
000         PARAMETER NODS=300,NELS=260,NF=20000,MAX=600
000         COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST
000   C
000   C
000   C        GET CONCENTRATED LOAD IF THERE IS ANY
000         WRITE(6,677)
000         DO 920 NC = 1,NAPC
000          READ(5,540) NODE,PZ,PR
000          WRITE(6,688)NODE,PZ,PR
000          II = (NODE-1) * 2
000          APF(II+1) = PZ
000          APF(II+2) = PR
000          ULOAD = PZ
```

```
000     920 CONTINUE                                                             00001500
000     540 FORMAT(I5,2F15.6)                                                    00001600
000     677 FORMAT(///10X,'APPLIED C. LOAD'//5X,'NODE',10X,'FORCE TO Z',5X,'     00001700
000        * FORCE TO R'//)                                                      00001800
000     688 FORMAT(5X,I5,2(5X,E12.4))                                            00001900
000         RETURN                                                               00002000
000         END                                                                  00002100
000   @ELT,SIH NASA*TPF$.EQLOAD,,,114076121110
000         SUBROUTINE EQLOAD(ULOAD)                                             00000100
000   C                                                                          00000200
000   C        THIS SUBROUTINE IS TO CALCULATE THE EQUIVALENT NODAL LOADS.       00000300
000   C                                                                          00000400
000         PARAMETER NODS=300,NELS=260,NF=20000,MAX=600                         00000500
000         COMMON /BLK2/ ID(NODS,2),IJKL(NELS,4),DE1,DE2                        00000600
000         COMMON /BLK3/ XK(NF),APF(MAX),IMAX,IHB,IHBI,LT,LAST                  00000700
000         COMMON /BLK6/ SIGR,SIGZ,SIGT,TAUZR,D(4,4),STIFF(8,8),KK(NELS,8),     00000800
000        1 R(NODS),Z(NODS),TOTDIS(MAX)                                         00000900
000         PI = 3.1415926                                                       00001000
000         XXI = 0.                                                             00001100
000         XXJ = 0.                                                             00001200
000         READ(5,500) NLDEL,ULOAD                                              00001300
000         WRITE(6,600) NLDEL                                                   00001400
000         DO 200 I = 1,NLDEL                                                   00001500
000         READ(5,510) NOL,NLFT,NRHT                                            00001600
000         KC = NRHT                                                            00001700
000         LC = NLFT                                                            00001800
000         EQL = (R(KC)**2-R(LC)**2)*PI*ULOAD/2.                                00001900
000         II = (KC-1) * 2 + 1                                                  00002000
000         JJ = (LC-1) * 2 + 1                                                  00002100
000         APF(II) = APF(II) + EQL                                              00002200
000         APF(JJ) = APF(JJ) + EQL                                              00002300
000     200 WRITE(6,620) NOL,LC,KC,EQL,ULOAD                                     00002400
000     620 FORMAT(I6,5X,2I4,2F20.7)                                             00002500
000     500 FORMAT(I5,F20.9)                                                     00002600
000     510 FORMAT(3I5)                                                          00002700
000     600 FORMAT(///'      CALCULATION OF EQUIVALENT NODAL LOAD'/              00002800
000        *   '         NUMBER OF LOADED ELEMENTS = ',I4/                       00002900
000        * '  ELEM NO----LOADED NODE-----EQUIV LOAD-----GIVEN U.LOAD'/)        00003000
000     610 FORMAT(2I10,2F15.7)                                                  00003100
000         RETURN                                                               00003200
000         END                                                                  00003300
000   @ELT,SIH NASA*TPF$.ZERO,,,114100121110
000         SUBROUTINE ZERO(A,N,M)                                               00000100
000         DIMENSION A(1)                                                       00000200
000         K = N * M                                                            00000300
000         DO 100 I = 1,K                                                       00000400
000     100 A(I) = 0.                                                            00000500
000         RETURN                                                               00000600
000         END                                                                  00000700
000   @MAP,IX
000   @XQT
000     CONE PENETROMETER
000       90   71         3  100                5    2
000                                        25.
000
000        7000000.          0.3                   10.                 0.45
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                                                          30
001202     000     44   45   37   36                                                          31
001203     000     45   46   38   37                                                          32
001204     000     46   47   39   38                                                          33
001205     000     47   48   40   39                                                          34
001206     000     48   49   41   40                                                          35
001207     000     49   50   42   41                                                          36
001208     000     51   52   44   43                                                          37
001209     000     52   53   45   44                                                          38
001210     000     53   54   46   45                                                          39
001211     000     54   55   47   46                                                          40
001212     000     55   56   48   47                                                          41
001213     000     56   57   49   48                                                          42
001214     000     57   58   50   49                                                          43
001215     000     59   60   52   51                                                          44
001216     000     60   61   53   52                                                          45
001217     000     61   62   54   53                                                          46
001218     000     62   63   55   54                                                          47
001219     000     63   64   56   55                                                          48
001220     000     64   65   57   56                                                          49
001221     000     65   66   58   57                                                          50
001222     000     67   68   60   59                                                          51
001223     000     68   69   61   60                                                          52
001224     000     69   70   62   61                                                          53
001225     000     70   71   63   62                                                          54
001226     000     71   72   64   63                                                          55
001227     000     72   73   65   64                                                          56
001228     000     73   74   66   65                                                          57
001229     000     75   76   68   67                                                          58
001230     000     76   77   69   68                                                          59
001231     000     77   78   70   69                                                          60
001232     000     78   79   71   70                                                          61
001233     000     79   80   72   71                                                          62
001234     000     80   81   73   72                                                          63
001235     000     81   82   74   73                                                          64
001236     000     83   84   76   75                                                          65
001237     000     84   85   77   76                                                          66
001238     000     85   86   78   77                                                          67
001239     000     86   87   79   78                                                          68
001240     000     87   88   80   79                                                          69
001241     000     88   89   81   80                                                          70
001242     000     89   90   82   81                                                          71
001243     000      2    9    4   10    5    11
001244     000      2   50.
001245     000      1    1    2
001246     000      2    2    3


GFIN
```

APPENDIX 2

DATA INPUT FORMAT

Card 1: FORMAT (20A4)

TITLE - Title of the problem

Card 2: FORMAT (1015)

(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(I,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

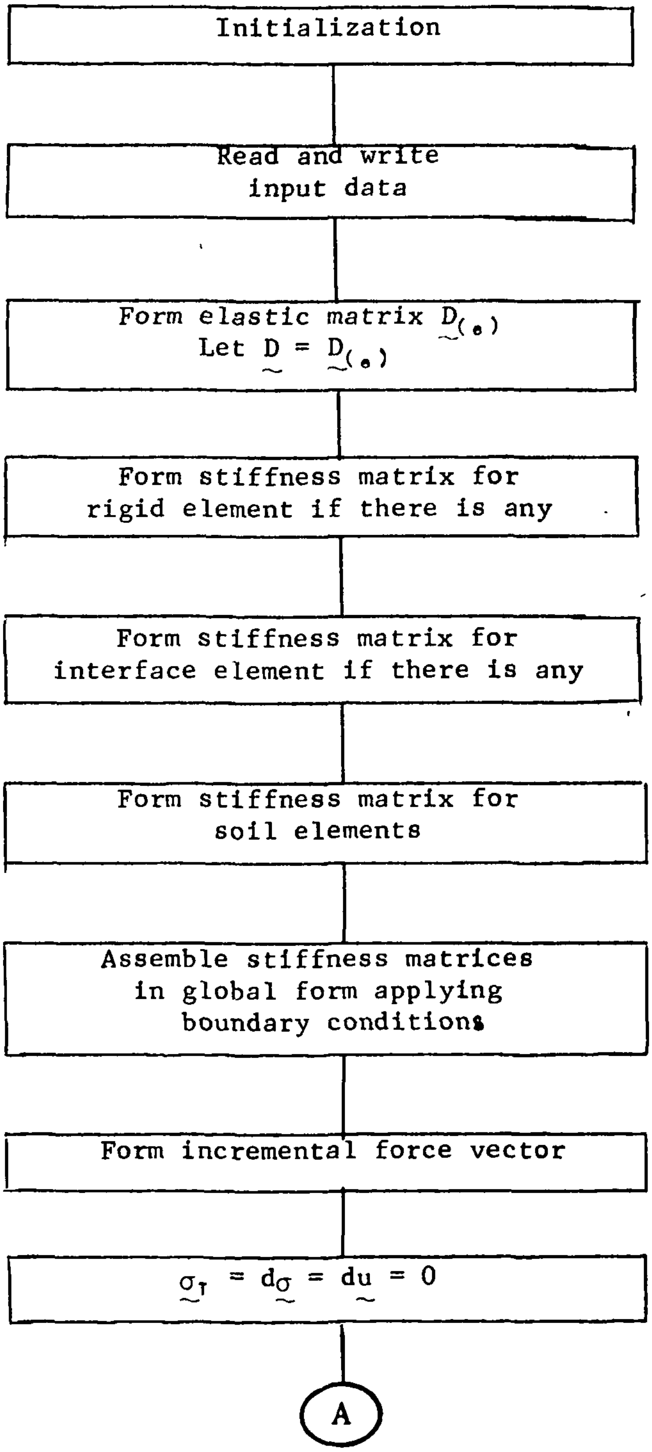
(3) NRHT - Node No. at right

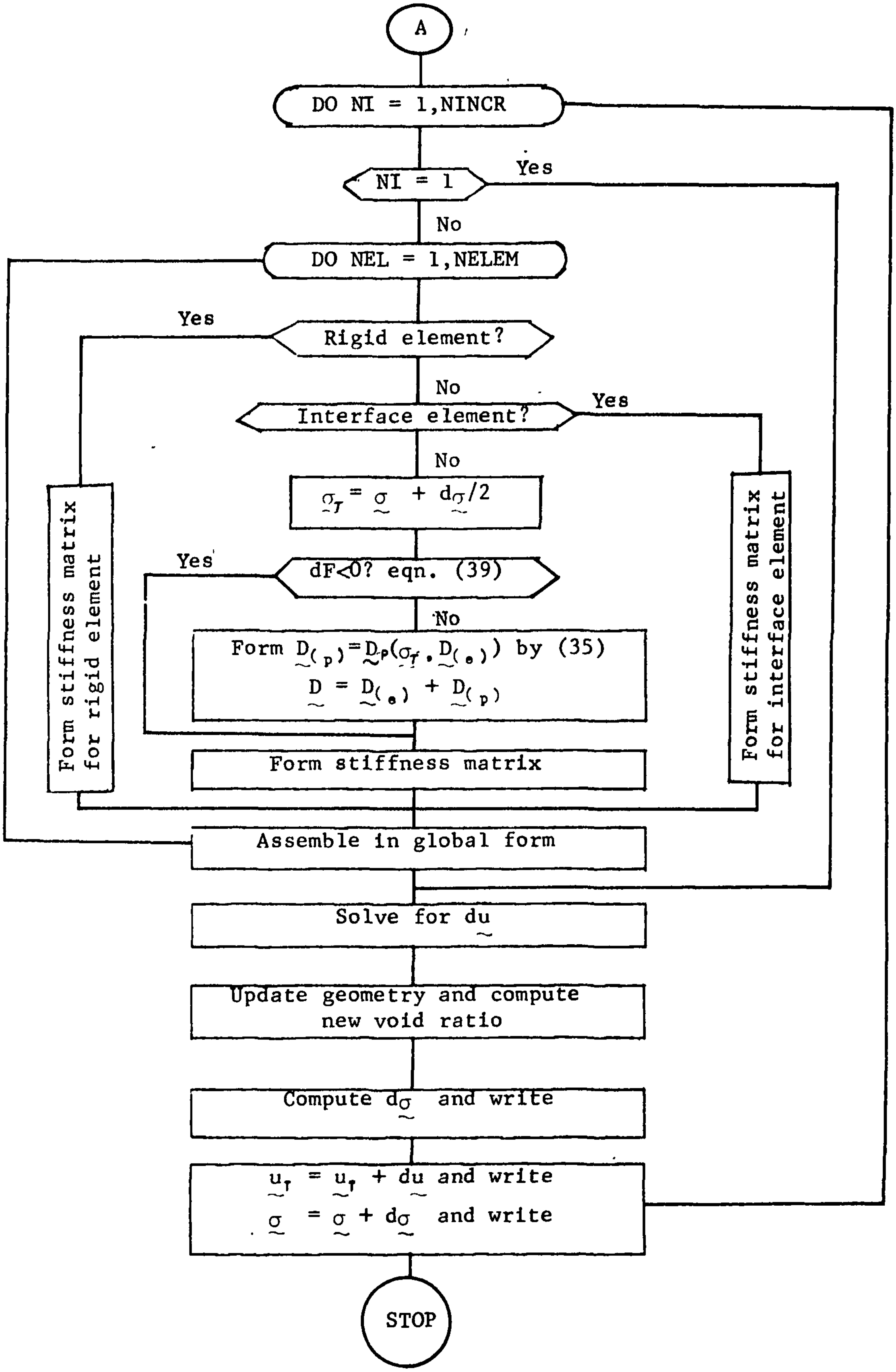
Repeat NLDEL times

** Not reguired if NULOAD=0

APPENDIX 3

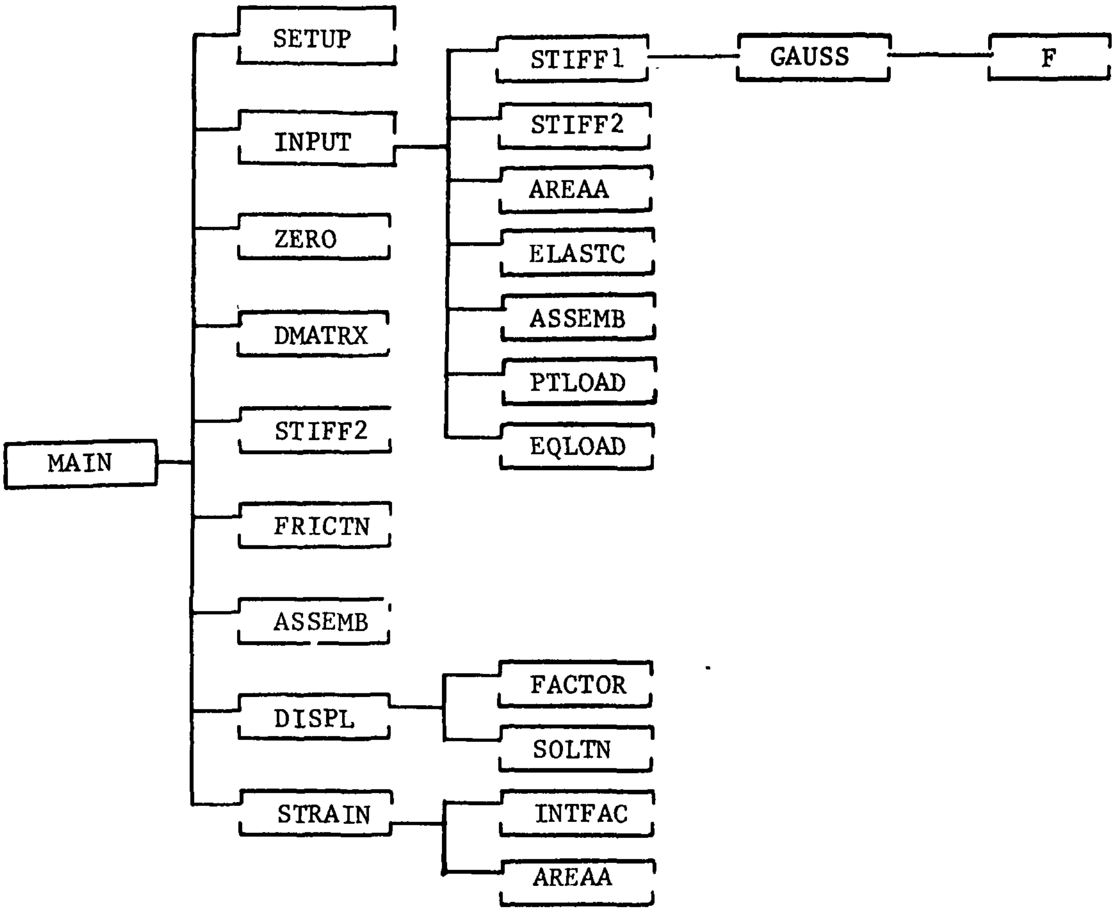
FLOW CHART


Initialization
Read and write input data
Form elastic matrix $\underset{\sim}{D}_{(e)}$ Let $\underset{\sim}{D} = \underset{\sim}{D}_{(e)}$
Form stiffness matrix for rigid element if there is any
Form stiffness matrix for interface element if there is any
Form stiffness matrix for soil elements
Assemble stiffness matrices in global form applying boundary conditions
Form incremental force vector
$\underset{\sim}{\sigma}_T = d\underset{\sim}{\sigma} = d\underset{\sim}{u} = 0$
A


A
DO NI = 1,NINCR
NI = 1
Yes
No
DO NEL = 1,NELEM
Rigid element?
Yes
No
Interface element?
Yes
No
$\sigma_T = \sigma + d\sigma/2$
dF<0? eqn. (39)
Yes
No
Form $D_{(p)} = D_p(\sigma_T, D_{(e)})$ by (35)
$D = D_{(e)} + D_{(p)}$
Form stiffness matrix for rigid element
Form stiffness matrix for interface element
Form stiffness matrix
Assemble in global form
Solve for du
Update geometry and compute new void ratio
Compute $d\sigma$ and write
$u_T = u_T + du$ and write
$\sigma = \sigma + d\sigma$ and write
STOP

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 $D = D_{(e)} + \underset{\sim}{D}_{(p)}$ 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 $\underset{\sim}{D}_{(e)}$
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 $\underset{\sim}{du}$ for first 10 nodes.
FACTOR	Factors the given simultaneous eqns. in one dimensional array.
SOLTN	Backward substituion 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 $\underset{\sim}{D}$.

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], Uffelmann [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.

$$\dot{k} + \dot{U} = R \tag{1}$$

Here the superposed dot indicates a time rate, and

$$k = \frac{1}{2}\int_V \rho v_i v_i dV \tag{2}$$

$$U = \int_V \rho \epsilon dV \tag{3}$$

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

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

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

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

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

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

$$\sigma^{ij}_{,i} + \rho F^j - \rho a^j = 0 \tag{6}$$

and
$$\rho\dot{\epsilon} = \sigma^{ij} v_{j,i} = \sigma^{ij}\dot{\gamma}_{ij} \tag{7}$$

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 $\gamma_{ij}(\Delta t) = \overset{(e)}{\gamma}_{ij}(\Delta t) + \overset{(p)}{\gamma}_{ij}(\Delta t)$ and incremental internal state variables (or hidden variables) $\overset{(r)}{\alpha}_{ij}(\Delta t) = \overset{(r)(e)}{\alpha}_{ij}(\Delta t) + \overset{(r)(p)}{\alpha}_{ij}(\Delta t)$ where $^{(e)}$ and $^{(p)}$ represent elastic and plastic components, respectively. This statement may be given by

$$\Phi(\Delta t) = \hat{\phi}[\overset{(e)}{\gamma}_{ij}(\Delta t), \overset{(p)}{\gamma}_{ij}(\Delta t), \overset{(r)(e)}{\alpha}_{ij}(\Delta t), \overset{(r)(p)}{\alpha}_{ij}(\Delta t)] \tag{8}$$

$$\sigma^{ij}(\Delta t) = \hat{\sigma}[\overset{(e)}{\gamma}_{ij}(\Delta t), \overset{(p)}{\gamma}_{ij}(\Delta t), \overset{(r)(e)}{\alpha}_{ij}(\Delta t), \overset{(r)(p)}{\alpha}_{ij}(\Delta t) \tag{9}$$

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

$$\rho\dot{\Phi} = \rho\dot{e} = \sigma^{ij}\dot{\gamma}_{ij}$$

or for the small time interval Δt,

$$\rho\dot{\Phi}(\Delta t) = \sigma^{ij}(\Delta t)(\dot{\overset{(e)}{\gamma}}_{ij}(\Delta t) + \dot{\overset{(p)}{\gamma}}_{ij}(\Delta t)) \tag{10}$$

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

$$\rho\Phi(\Delta t) = \frac{1}{2}E^{ijk\ell}\gamma_{ij}^{(e)}\gamma_{k\ell}^{(e)} + \frac{1}{2}\overset{*}{E}{}^{ijk\ell}\gamma_{ij}^{(p)}\gamma_{k\ell}^{(p)}$$
$$+ \frac{1}{2}\sum_{r=1}^{n}\xi_{(r)}^{ijk\ell}(\alpha_{ij}^{(r)(e)} + \alpha_{ij}^{(r)(p)})(\alpha_{k\ell}^{(r)(e)} + \alpha_{k\ell}^{(r)(p)})$$
$$+ \sum_{r=1}^{n}F_{(r)}^{ijk\ell}(\alpha_{k\ell}^{(r)(e)} + \alpha_{k\ell}^{(r)(p)})(\gamma_{ij}^{(e)} + \gamma_{ij}^{(p)}) \qquad (11)$$

where $E^{ijk\ell}$ and $\overset{*}{E}{}^{ijk\ell}$ represent tensors of elastic and plastic moduli, respectively; $\xi_{(r)}^{ijk\ell}$ 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 $\gamma_{ij}^{(e)}$ and $\gamma_{ij}^{(p)}$ 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 $\gamma_{ij}^{(e)}$ and $\gamma_{ij}^{(p)}$ is nonexisting.

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

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

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 $\dot{\gamma}_{ij}$ within the time interval Δt given by

$$\dot{\gamma}_{ij}(s) = \dot{\gamma}_{ij}(s-1) + \frac{\tau-(t-\Delta t)}{\Delta t}(\dot{\gamma}_{ij}(s-1) - \dot{\gamma}_{ij}(s)) \qquad (13)$$

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

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

in which

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

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

The derivation of these parameters is given in Appendix 1.

Rewriting (10) for the current time step $^{(s)}$ as

$$ \rho\left\{\frac{\partial\Phi^{(s)}}{\partial\gamma_{ij}^{(e)(s)}}\dot{\gamma}_{ij}^{(e)(s)} + \frac{\partial\Phi^{(s)}}{\partial\gamma_{ij}^{(p)(s)}}\dot{\gamma}_{ij}^{(p)(s)} + \frac{\partial\Phi^{(s)}}{\partial\,{}^{(r)(e)}\alpha_{ij}^{(s)}}\,{}^{(r)}\dot{\alpha}_{ij}^{(s)} + \frac{\partial\Phi^{(s)}}{\partial\,{}^{(r)(p)}\alpha_{ij}^{(s)}}\,{}^{(r)(p)}\dot{\alpha}_{ij}^{(s)}\right\} $$

$$ -\sigma^{ij(s)}\left(\dot{\gamma}_{ij}^{(e)(s)} + \dot{\gamma}_{ij}^{(p)(s)}\right) = 0 \qquad (15) $$

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

$$ \{E^{ijk\ell}\gamma_{k\ell}^{(e)(s)} + \sum_{r=1}^{n}\xi_{(r)}^{ijk\ell}\left({}^{(r)}A\,{}^{(r)}\alpha_{k\ell}^{(s-1)} + {}^{(r)}B\,\dot{\gamma}_{k\ell}^{(e)(s-1)} + {}^{(r)}C\,\dot{\gamma}_{k\ell}^{(s)}\right. $$

$$ -\sigma^{ij(s)}\}\dot{\gamma}_{ij}^{(e)(s)} + \sum_{r=1}^{n}\xi_{(r)}^{ijk\ell}\left({}^{(r)}\alpha_{ij}^{(s)}\,{}^{(r)(p)}\dot{\alpha}_{k\ell}^{(s)} + {}^{(r)(p)}\alpha_{k\ell}^{(s)}\dot{\gamma}_{ij}^{(e)(s)}\right. $$

$$ + {}^{(r)}\alpha_{k\ell}^{(s)}\dot{\gamma}_{ij}^{(s)} + \gamma_{ij}\,{}^{(r)}\dot{\alpha}_{k\ell}^{(s)} - \sigma^{ij(s)}\dot{\gamma}_{ij}^{(p)(s)} = 0 $$

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

$$ \sigma^{ij(s)} = E^{ijk\ell}\gamma_{k\ell}^{(e)(s)} + \sum_{r=1}^{n}\xi_{(r)}^{ijk\ell}\left({}^{(r)}A\,{}^{(r)}\alpha_{k\ell}^{(s-1)} + {}^{(r)}B\,\dot{\gamma}_{k\ell}^{(e)(s-1)} + {}^{(r)}C\,\dot{v}_{k\ell}^{(e)(s)}\right) \qquad (16) $$

$$\overset{*}{E}^{ijk\ell}\dot{\gamma}^{(p)}_{ij}(s) - \sigma^{ij}(s)\dot{\gamma}^{(p)}_{ij} + \sum_{r=1}^{n} \xi^{ijk\ell}_{(r)}\Big[\overset{(r)}{\alpha}_{ij}(s)\overset{(\dot{r})(p)}{\alpha}_{k\ell}(s) + \overset{(r)(p)}{\alpha}_{k\ell}(s)\dot{\gamma}^{(p)}_{ij}(s) + \overset{(r)}{\alpha}_{k\ell}(s)\dot{\gamma}_{ij}(s) + \gamma_1\int^{(s)}\overset{(r)}{\dot{\alpha}}_{k\ell}(\omega)\Big] = 0 \qquad (17)$$

Here (16) represents the relationship

$$\sigma^{ij} = \rho\,\frac{\partial\Phi}{\partial\gamma^{(e)}_{ij}}$$

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,

$$d\sigma^{ij} = (E^{ijk\ell} + \overset{*}{E}{}^{ijk\ell})d\gamma_{k\ell} \tag{18}$$

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 $\overset{*}{E}{}^{ijk\ell}$ within the time interval and if we proceed with (18) with iterative cycling for further updating $\overset{*}{E}{}^{ijk\ell}$ 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,

$$d\sigma^{ij(s)} = E^{ijk\ell}d\gamma_{k\ell}{}^{(s)} + \sum_{r=1}^{n} \xi_{(r)}^{ijk\ell}\,(\overset{(r)}{A}\,d\,\overset{(r)}{\alpha}{}_{k\ell}{}^{(s-1)}$$

$$+ \overset{(r)}{B}\,d\dot{\gamma}_{k\ell}{}^{(s-1)} + \overset{(r)}{C}\,d\dot{\gamma}_{k\ell}{}^{(s)}) + \overset{*}{E}{}^{ijk\ell}d\gamma_{k\ell}{}^{(s)} \tag{19}$$

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

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

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

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

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

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

Inserting (25) in (26) yields

$$\gamma_{ij} = A^k_{Nij} u^N_k \qquad (23)$$

where

$$A^k_{Nij} = \frac{1}{2}(\Psi_{N,i}\delta^k_j + \Psi_{N,j}\delta^k_i) \qquad (24)$$

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

$$\{\int_V \rho\Psi_M\Psi_N dV\, \ddot{u}^M_k + \int_V \sigma^{ij} A^k_{Nij} dV - \int_V \hat{F}^k \Psi_N dV\}\dot{u}^N_k = 0 \qquad (25)$$

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

$$M_{MN}\ddot{u}^M_k + \int_V \sigma^{ij} A^k_{Nij} dV = F_{Nk} \qquad (26)$$

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

$$M_{MN} = \int_V \rho \Psi_M \Psi_N dV \tag{27}$$

$$F_{Nk} = \int_V F^k \psi_N dV \tag{28}$$

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

$$M_{MN} d\ddot{u}^M_k + \int_V d\sigma^{ij} A^k_{Nij} dV = dF_{Nk} \tag{29}$$

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

$$M_{MN} d\ddot{U}^M_k(s) + C^{\ell k}_{MN} d\dot{u}^M_{\ell(s)} + (\overset{(e)}{K}{}^{\ell k}_{MN} + \overset{(p)}{K}{}^{\ell k}_{MN}) du^M_\ell(s) = dF_{Nk}(s) + dF^{(v)}_{Nk}(s) \tag{30}$$

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

$$C^{\ell k}_{MN} = \int_V \sum_{r=1}^{n} \xi_{(r)} \overset{(r)}{C}{}^{ijmn} A^\ell_{Mij} A^k_{Nmn} dV \tag{31}$$

$$\overset{(e)}{K}{}^{\ell k}_{MN} = \int_V E^{ijmn} A^\ell_{Mij} A^k_{Nmn} dV \tag{32}$$

$$\overset{(P)}{K}{}^{\ell k}_{MN} = \int_V \overset{*}{E}{}^{ijmn} A^\ell_{Mij} A^k_{Nmn} dV \tag{33}$$

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

$$dF^{(v)}_{Nk} = \int_V \sum_{r=1}^{n} \xi_{(r)} \overset{(r)}{A}{}^{ijmn} \overset{(r)}{\alpha}{}^{(s-1)}_{mn} A^k_{Nij} dV$$

$$+ \int_V \sum_{r=1}^{n} \xi_{(r)} \overset{(r)}{B}{}^{ijmn} A^\ell_{Nij} A^k_{Mmn} dV \, \{d\dot{u}^M_\ell(s-1)\} \tag{34}$$

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,

$$\{M_{MN} + \frac{\Delta t}{2} C_{MN} + \frac{\Delta t^2}{4} (\overset{(s)\ell k}{K}_{MN} + \overset{\ell k}{K}_{MN})\} d\ddot{u}^{M}_{\ell}{}^{(s)} = dF^{(s)}_{Nk} + dF^{(v)}_{Nk}{}^{(s)} - Q_{Nk}{}^{(s)} \tag{35}$$

where

$$Q_{Nk}{}^{(s)} = C_{MN} \{d\dot{u}^{M}_{k}{}^{(s-1)} + \frac{\Delta t^2}{2} d\ddot{u}^{M}_{k}{}^{(s-1)}\} + (\overset{(s)\ell k}{K}_{MN} + \overset{(p)\ell k}{K}_{MN}) \{du^{M}_{\ell}{}^{(s-1)} + \frac{\Delta t^2}{4} d\ddot{u}^{M}_{\ell}{}^{(s-1)}\} \tag{36}$$

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

$$du^{M}_{\ell}{}^{(s)} = du^{M}_{\ell}{}^{(s-1)} + \frac{\Delta t^2}{4} d\ddot{u}^{M}_{\ell}{}^{(s-1)} + \frac{\Delta t^2}{4} d\ddot{u}^{M}_{\ell}{}^{(s)} + \Delta t d\dot{u}^{M}_{\ell}{}^{(s-1)} \tag{38}$$

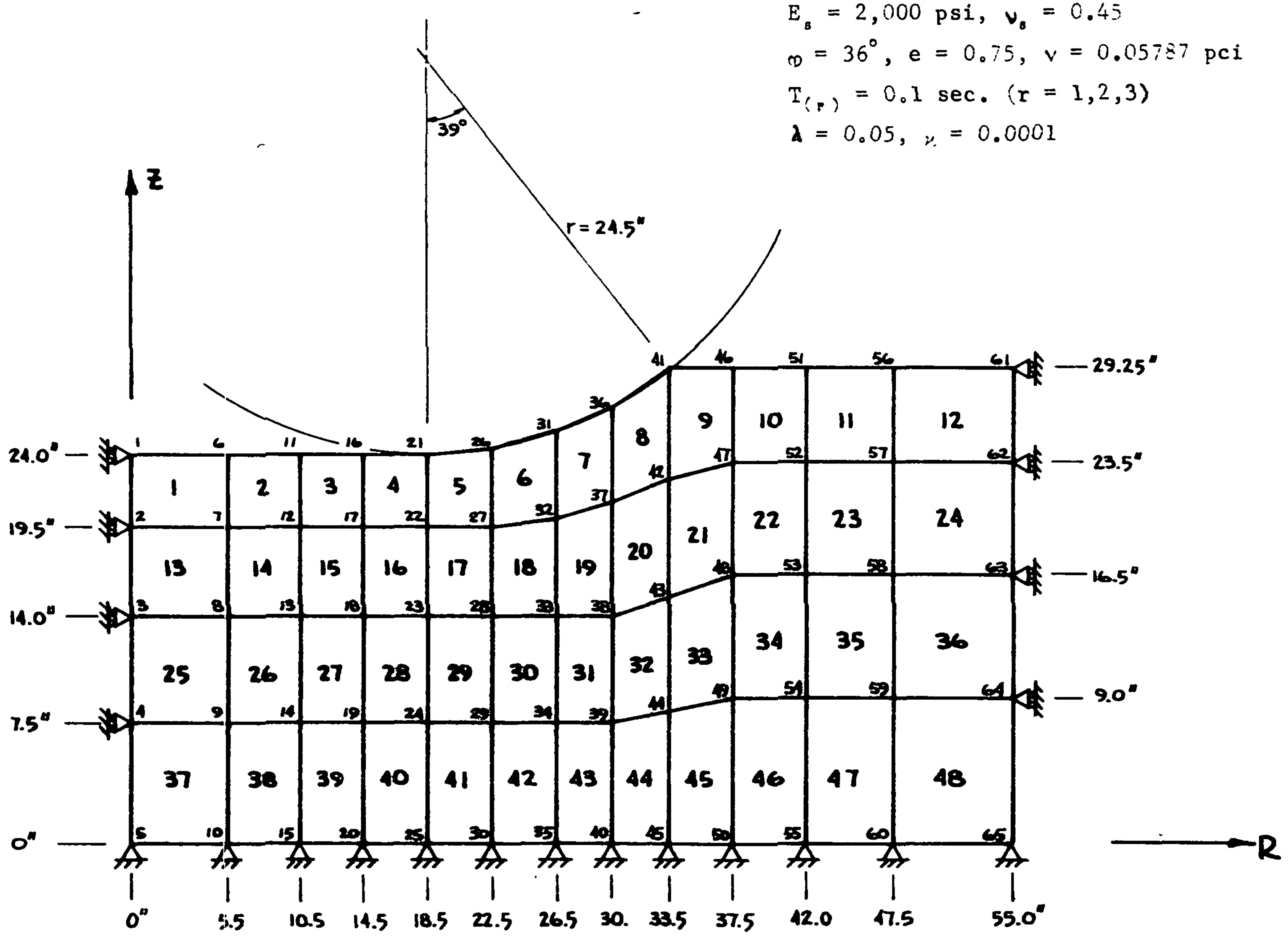
Initially all terms associated with ${}^{(s-1)}$ are zero and $d\ddot{u}^{M}_{\ell}{}^{(s)}$ in (35) can be solved from given initial and boundary conditions. Subsequently, $d\dot{u}^{M}_{\ell}{}^{(s)}$ and $du^{M}_{\ell}{}^{(s)}$ are calculated from (38). These responses or histories are then carried to the next time increment and back to (35). However, for the second increment it is necessary to check yield conditions 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 psi, ν_s = 0.45

φ = 36°, e = 0.75, γ = 0.05787 pci

$T_{(r)}$ = 0.1 sec. (r = 1,2,3)

λ = 0.05, ν = 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 ν_s = 0.45, angle of internal friction $\varphi = 36^{\circ}$, density γ = 0.05787 pci, relaxation time $T_{(r)}$ = 0.1 sec (r = 1, 2, 3), compression index λ = 0.05, and swelling index κ = 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.

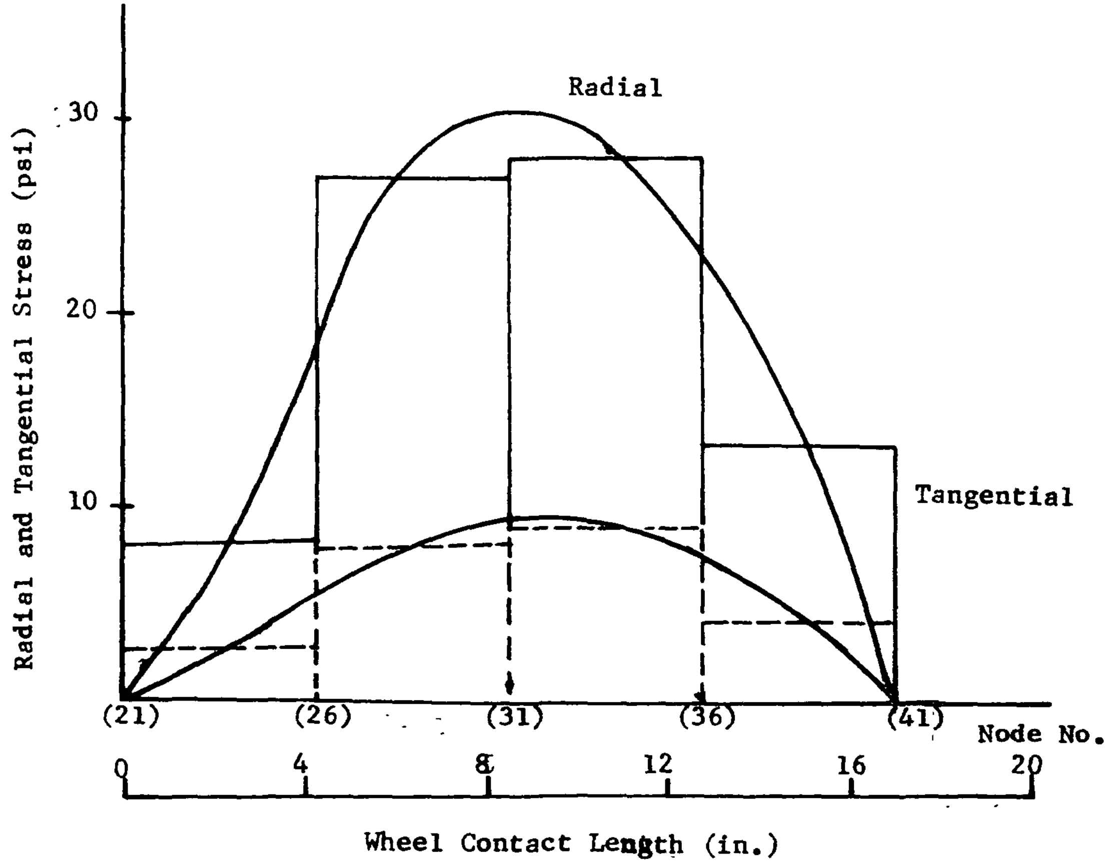


Figure 2: - Radial and Tangential Stress Distribution at the Interface for 3.1% Slip on Compact Sand. Ref. [6]

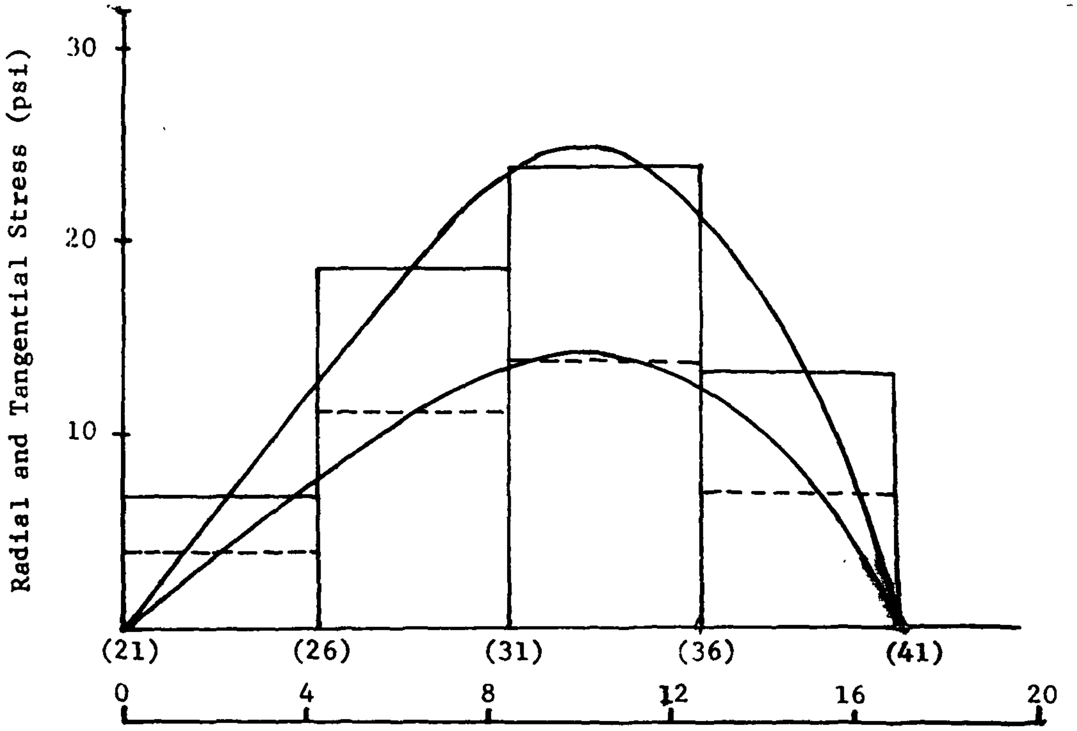
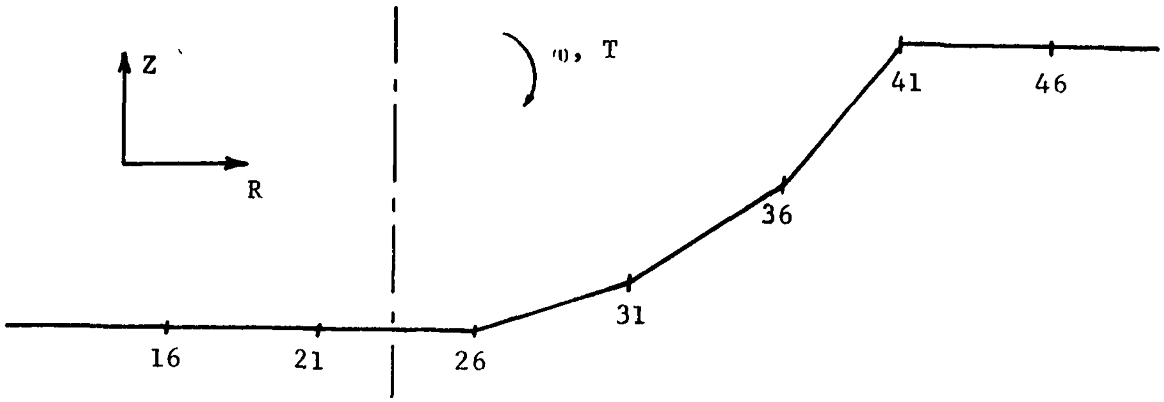


Figure 3: Radial and Tangential Stress Distribution at the Interface for 41.4% Slip on Compact Sand. Ref [6]



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.

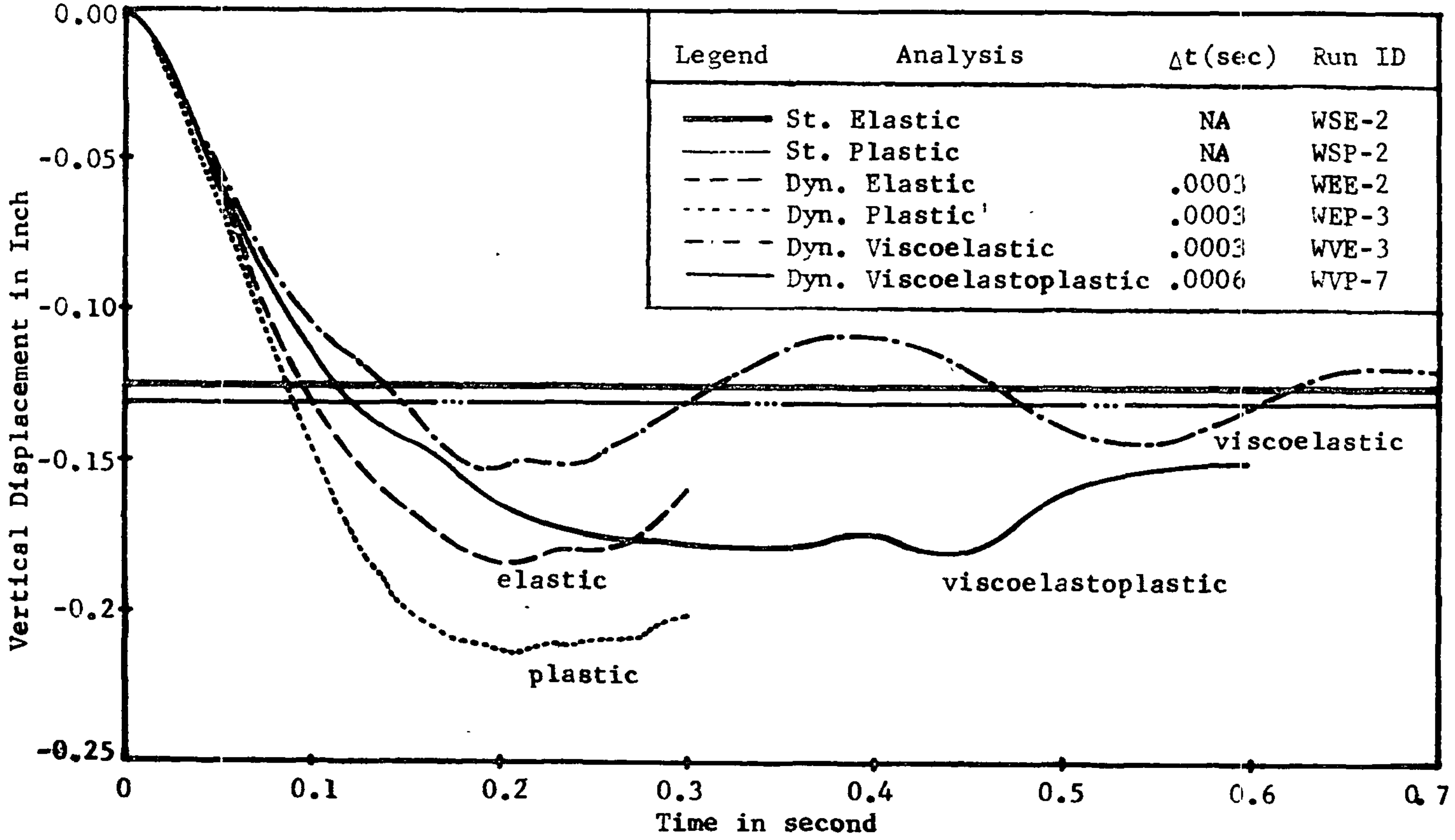


Figure 5: Time-Displacement Curves for 3.1% slip at Node No. 31.

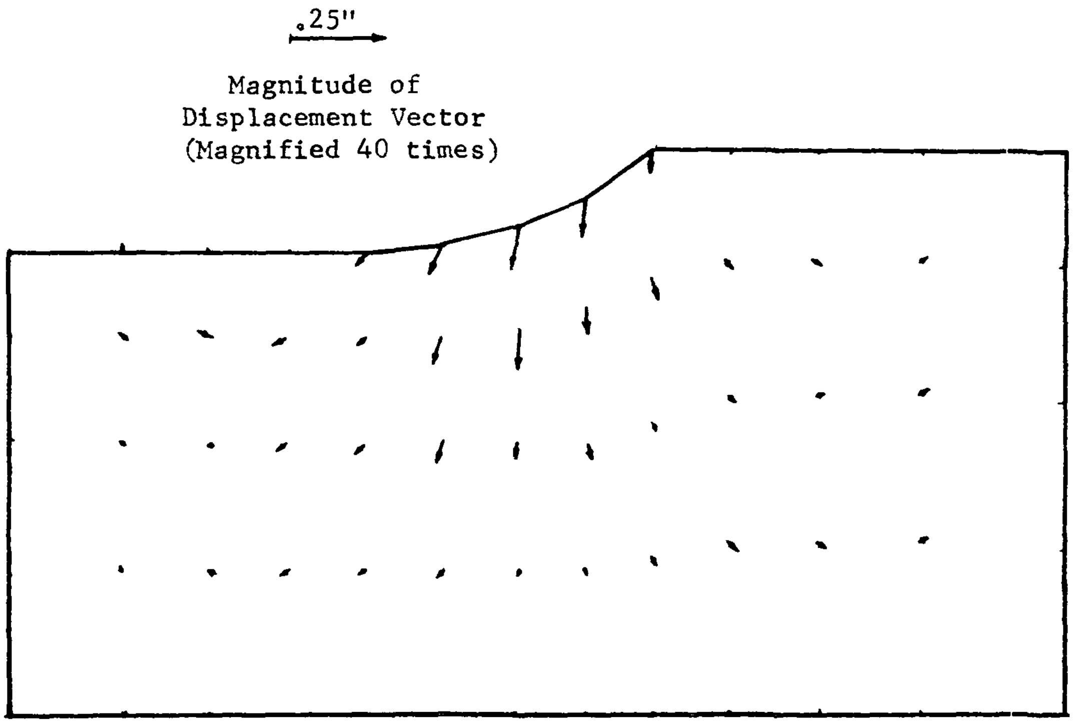


Figure 6: Vector Representation of Displacements (Static Elastoplastic Analysis for 3.1% Slip)

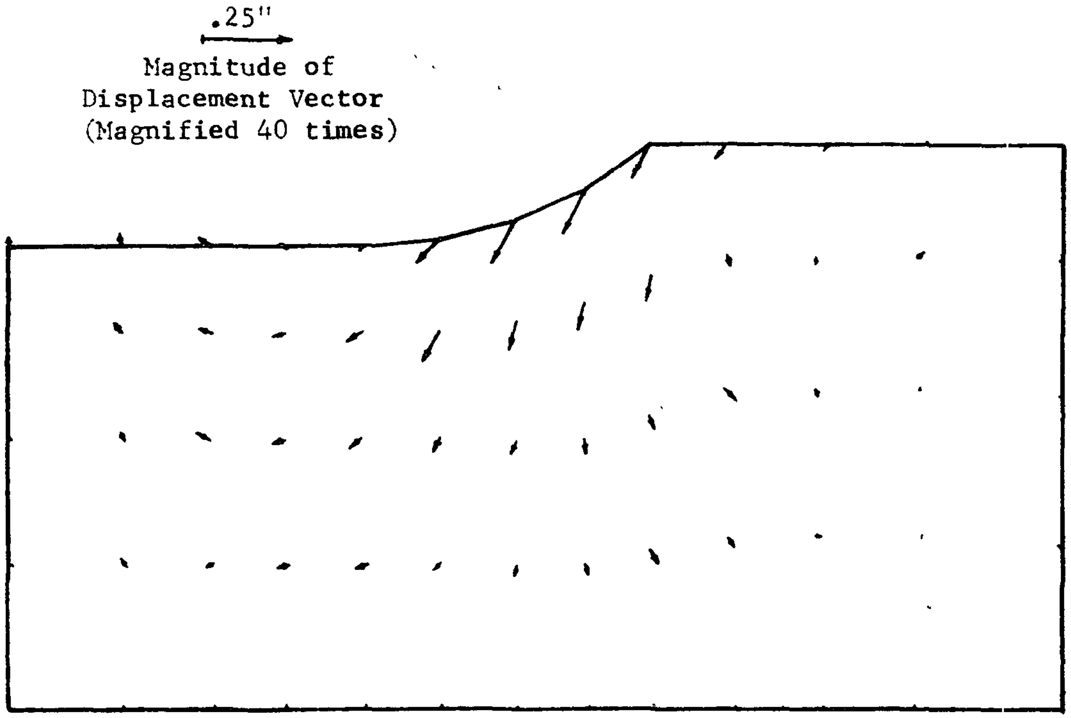


Figure 7: Vector Representation of Displacements (Static Elastoplastic Analysis for 41.4% Slip)

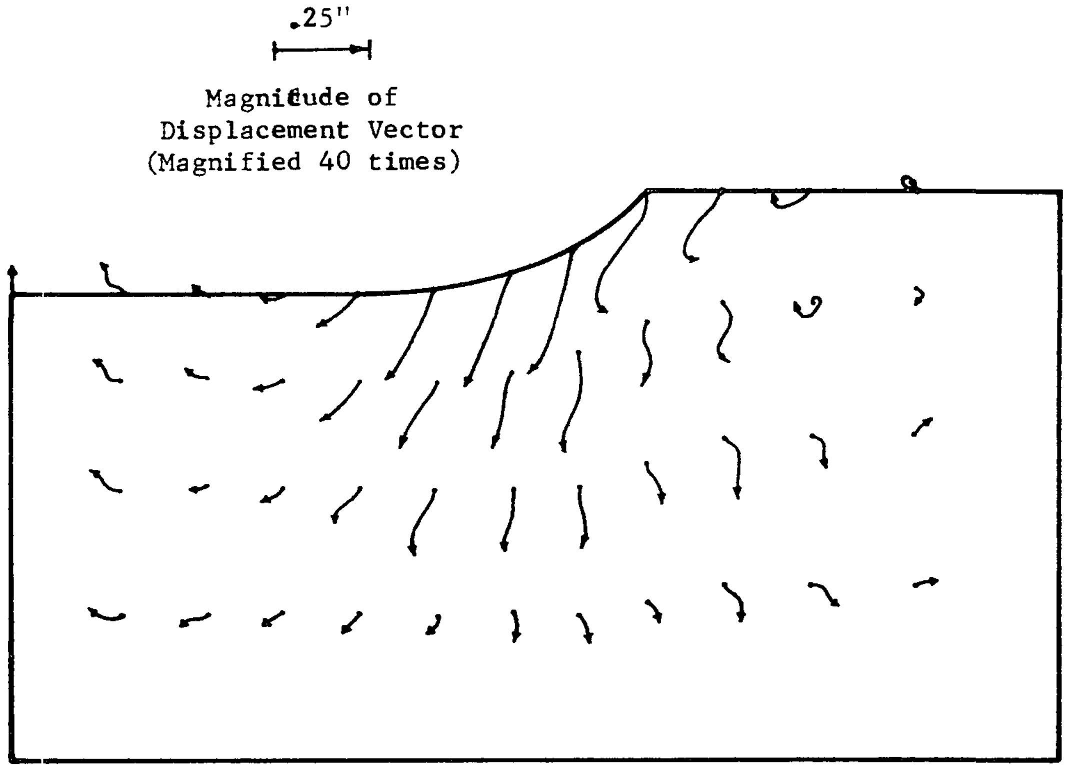


Figure 8: Vector Representation of Deformation (Dynamic Viscoelastoplastic Analysis for 41.4% Slip)

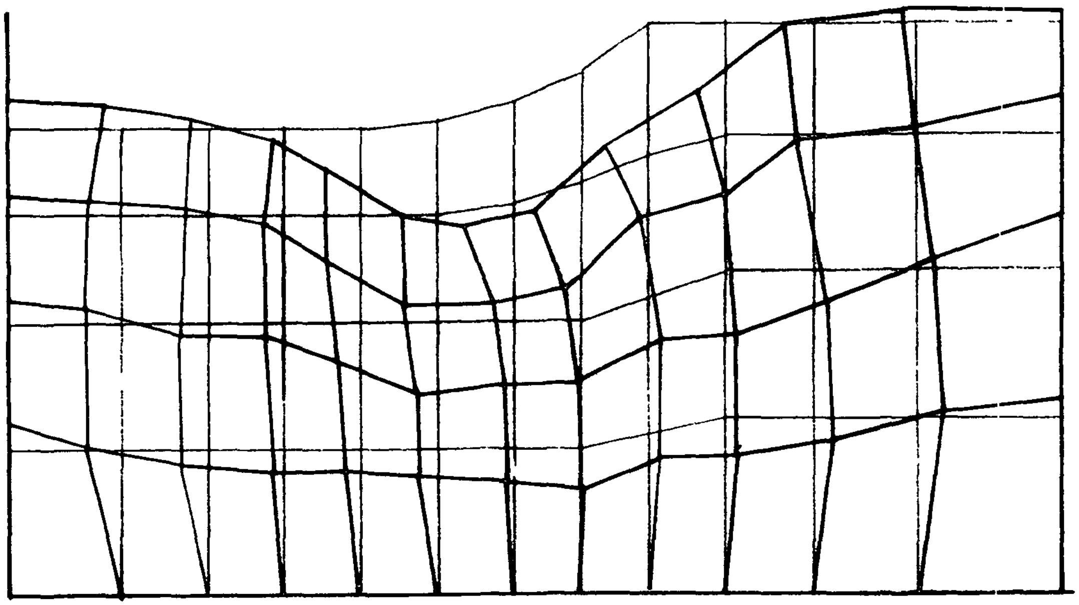


Figure 9: Deformed Configuration at .3 sec,
(Dynamic Viscoplastic Analysis for 41.4% Slip)

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

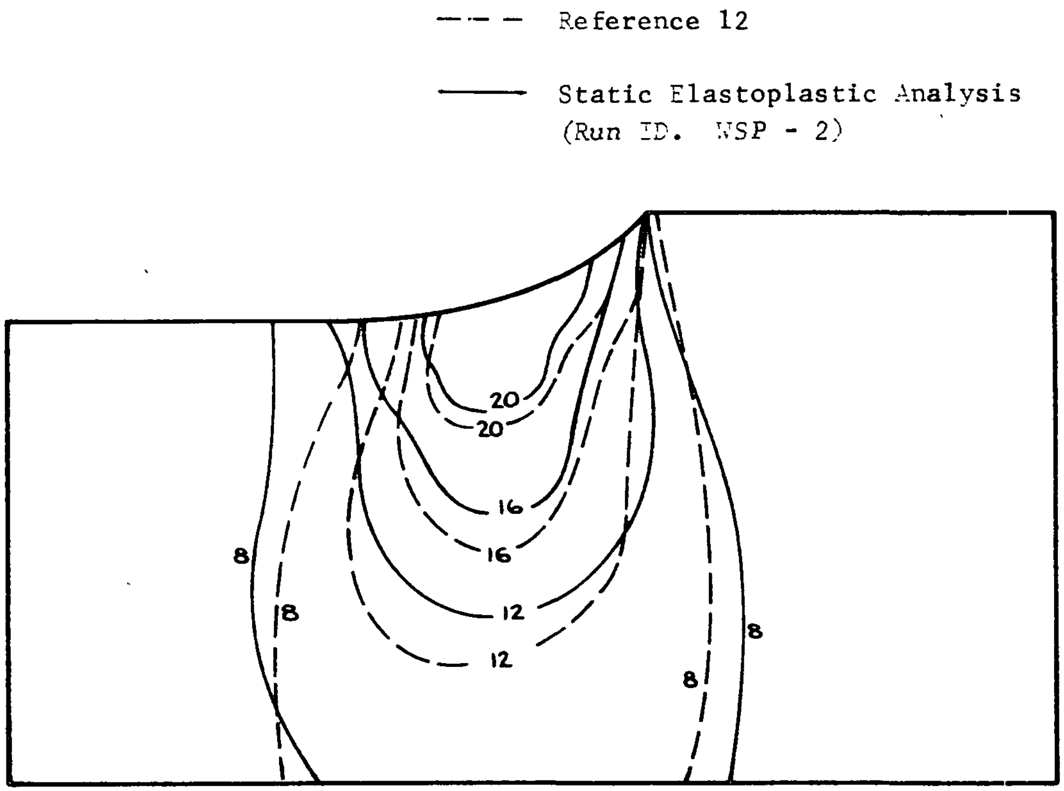


Figure 10: Isobars of Major Principal Stress (in psi) for 3.1% Slip

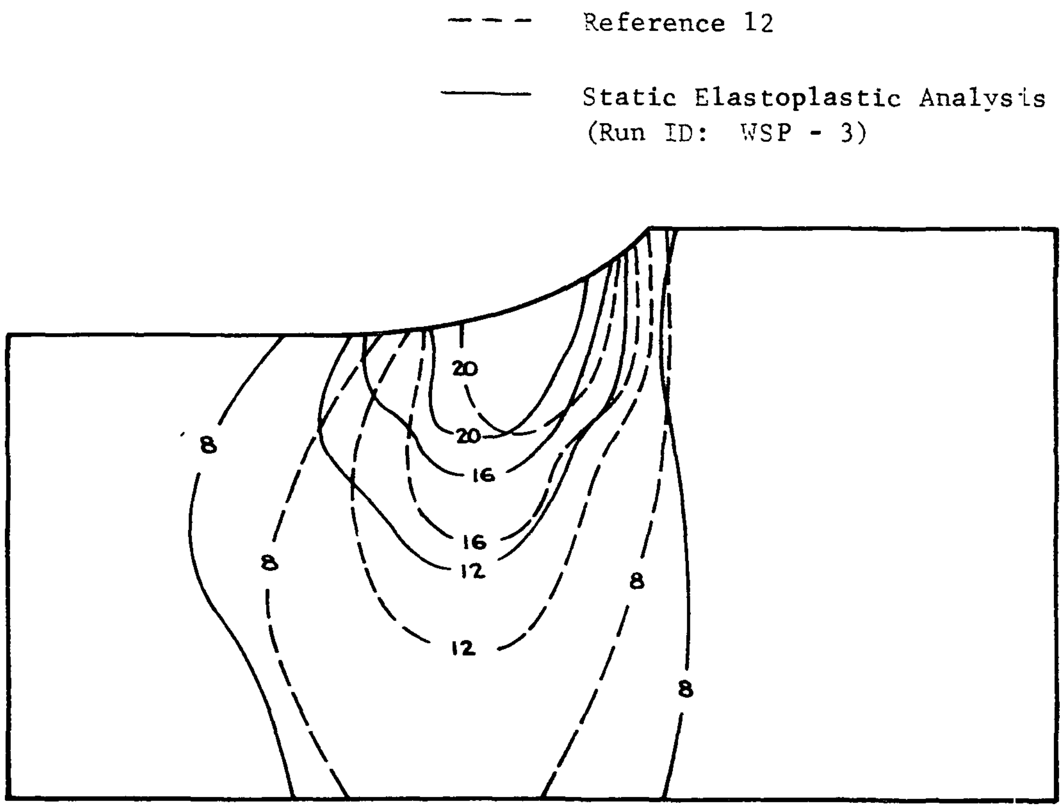


Figure 11: Isobars of Major Principal Stress (in psi) for 41.4% Slip

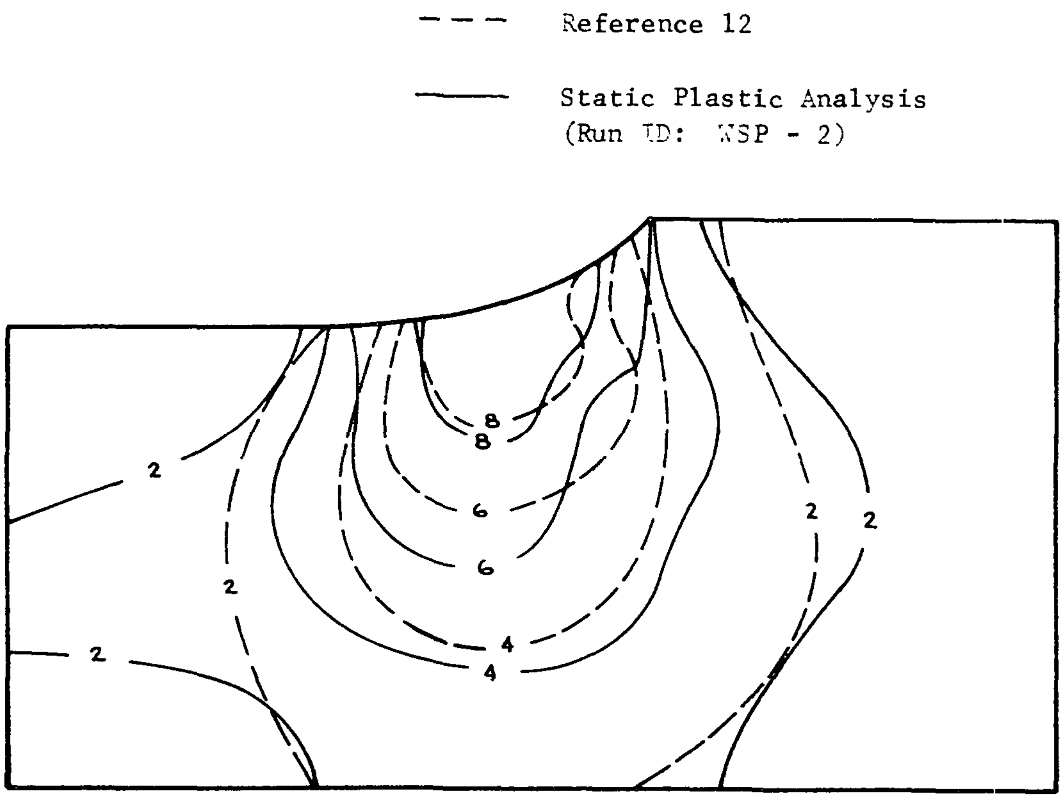


Figure 12: Isobars of Maximum Shear Stress for 3.1% Slip

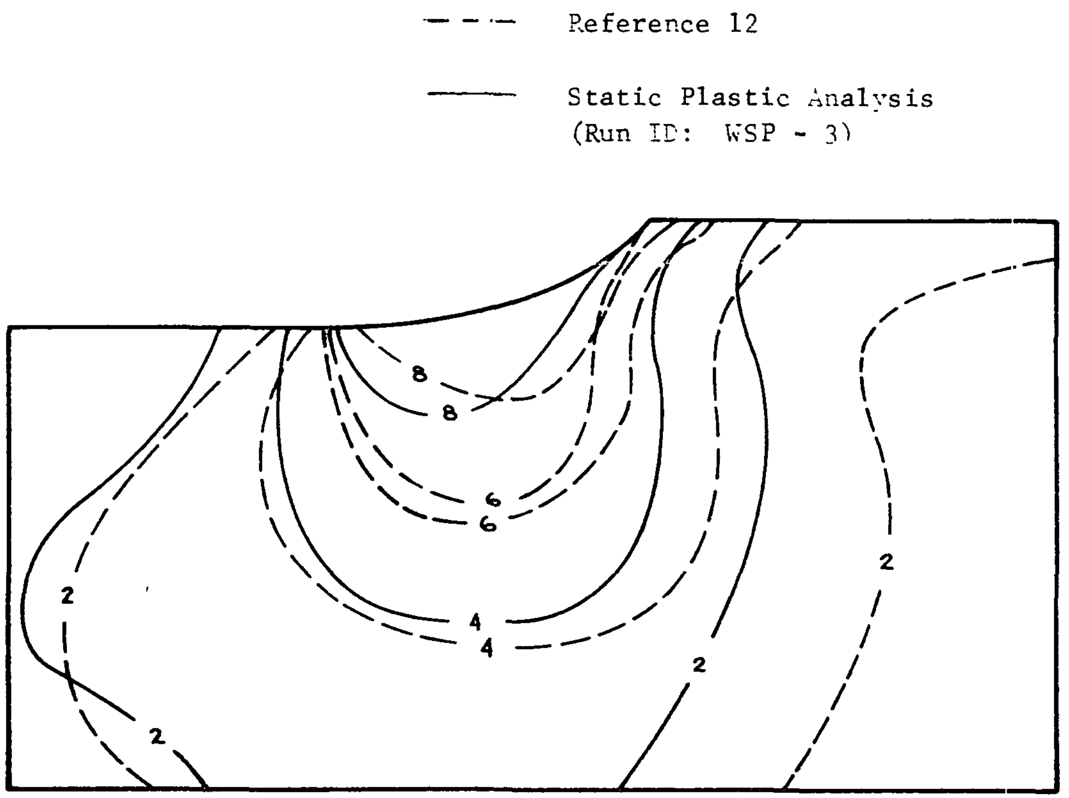


Figure 13: Isobars of Maximum Shear Stress for 41.4% Slip

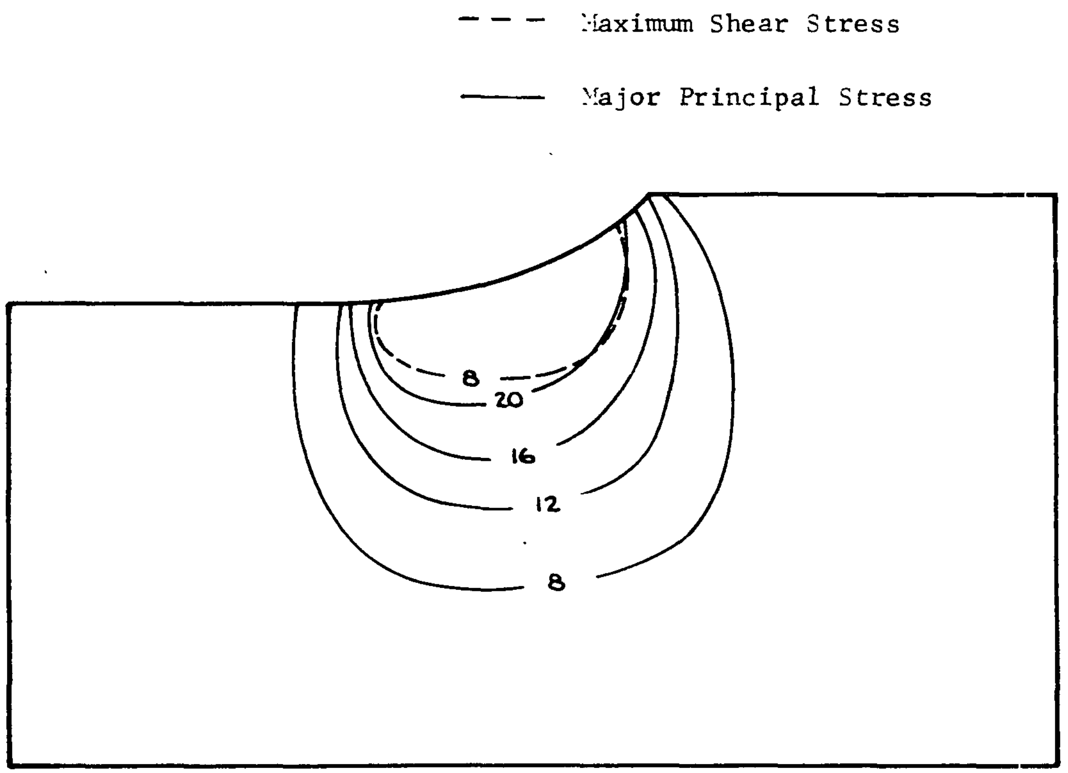


Figure 14: Isobard of Major Principal Stresses and Maximum Shear Stresses (psi) at t = 0.072 sec.

(Dynamic Elasto-plastic Analysis for 3.1% Slip)

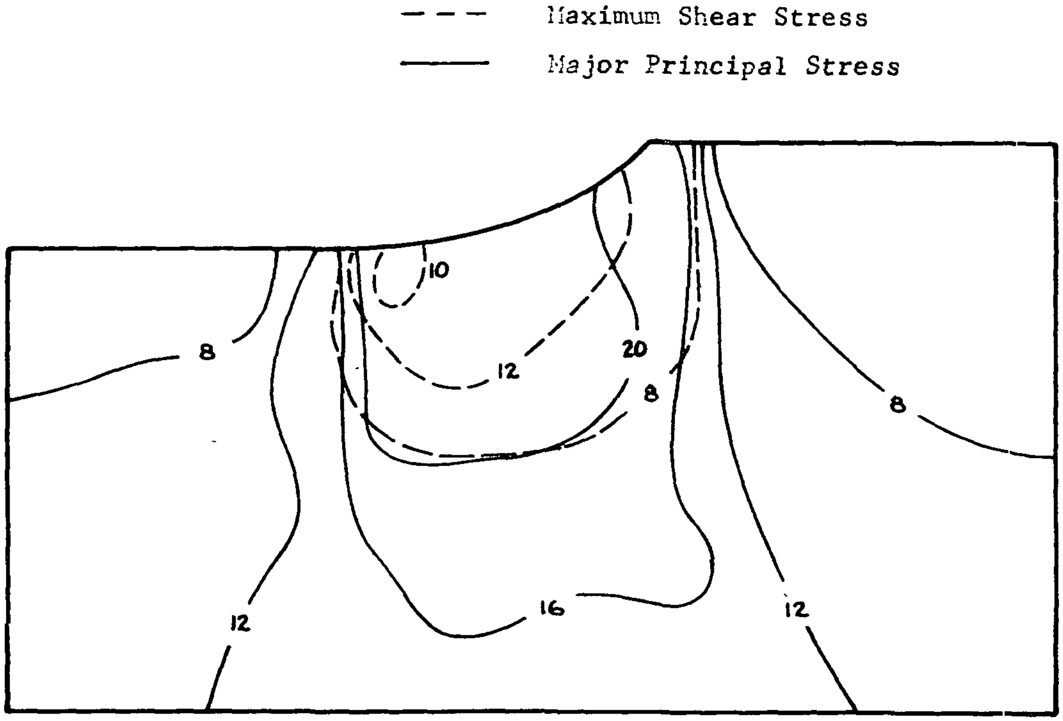


Figure 15: Isobars of Major Principal Stress and Maximum Shear Stress (psi) at t=0.15 sec. (Dynamic Elastoplastic Analysis for 3.1% Slip)

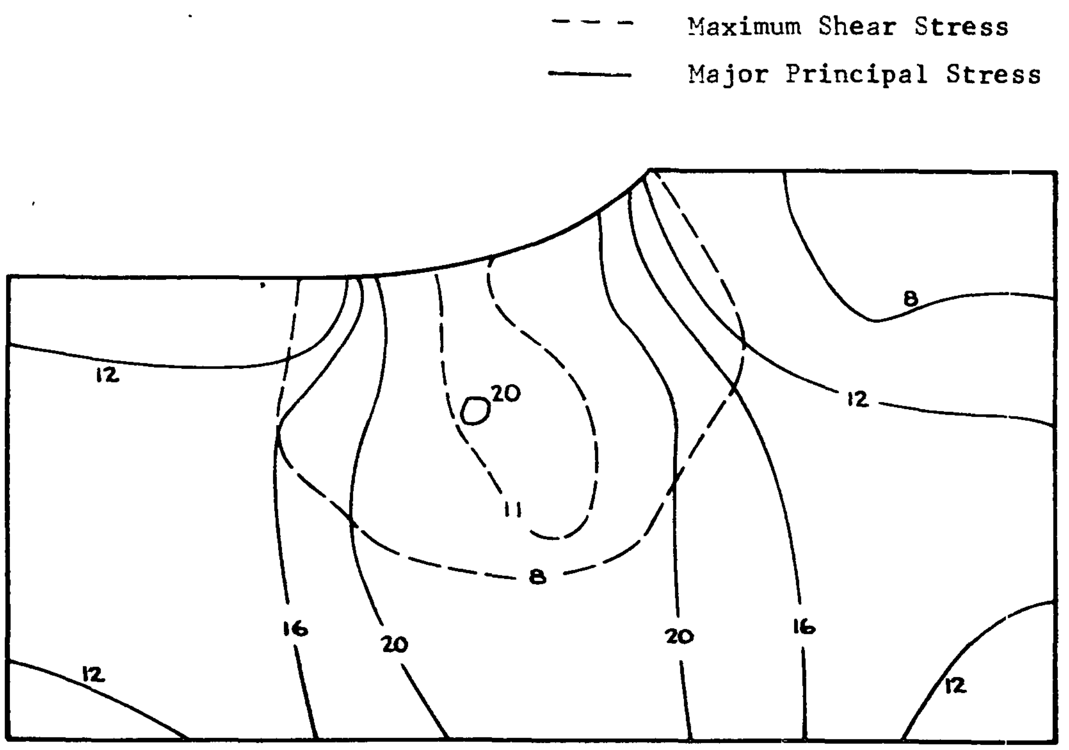


Figure 16: Isobars of Principal Stress and Maximum Shear Stress (Psi) at t=0.228 sec. (Dynamic Elastoplastic Analysis for 3.1% slip)

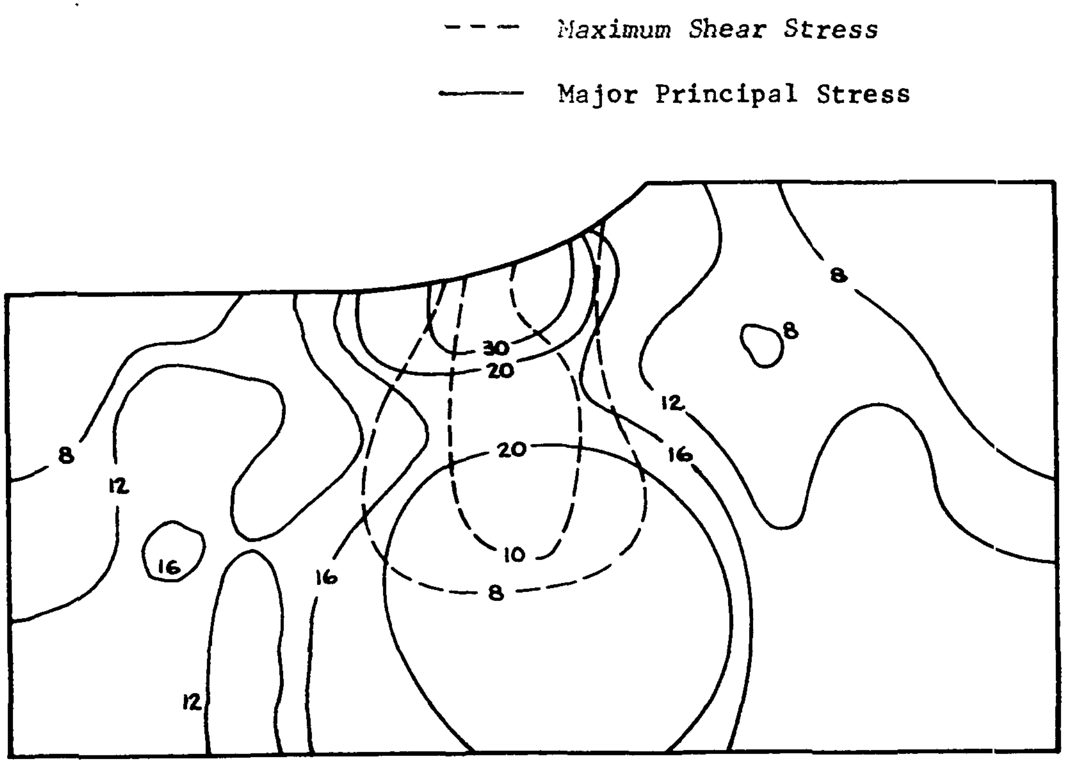


Figure 17: Isobarė of Major Principal Stress at Maximum Shear Stress (psi) at t=0.3 sec. (Dynamic Elastoplastic Analysis for 3.1% slip)

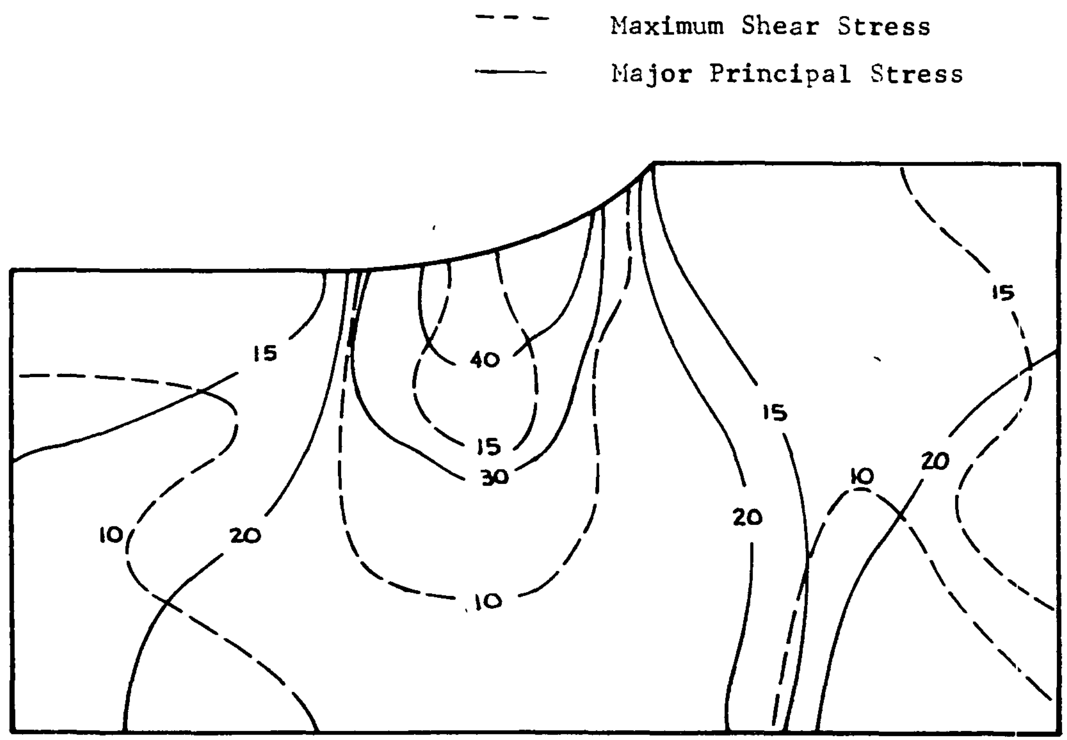


Figure 18: Isobars of Major Principal Stresses and Maximum Shear Stresses at t=0.3 sec (Dynamic Viscoelastoplastic Analysis for 3.1% Slip)

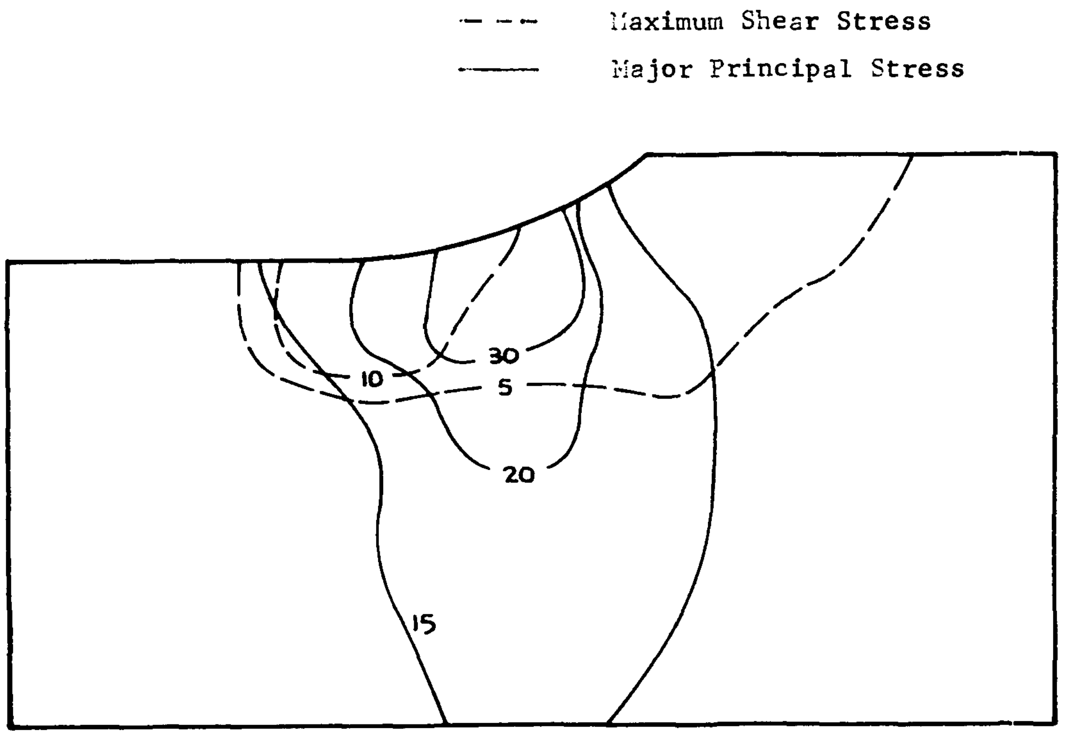


Figure 19: Isobars of Major Principal Stresses and Maximum Shear Stresses (psi) at .6 sec (Dynamic Viscoelastoplastic Analysis for 3.1% Slip)

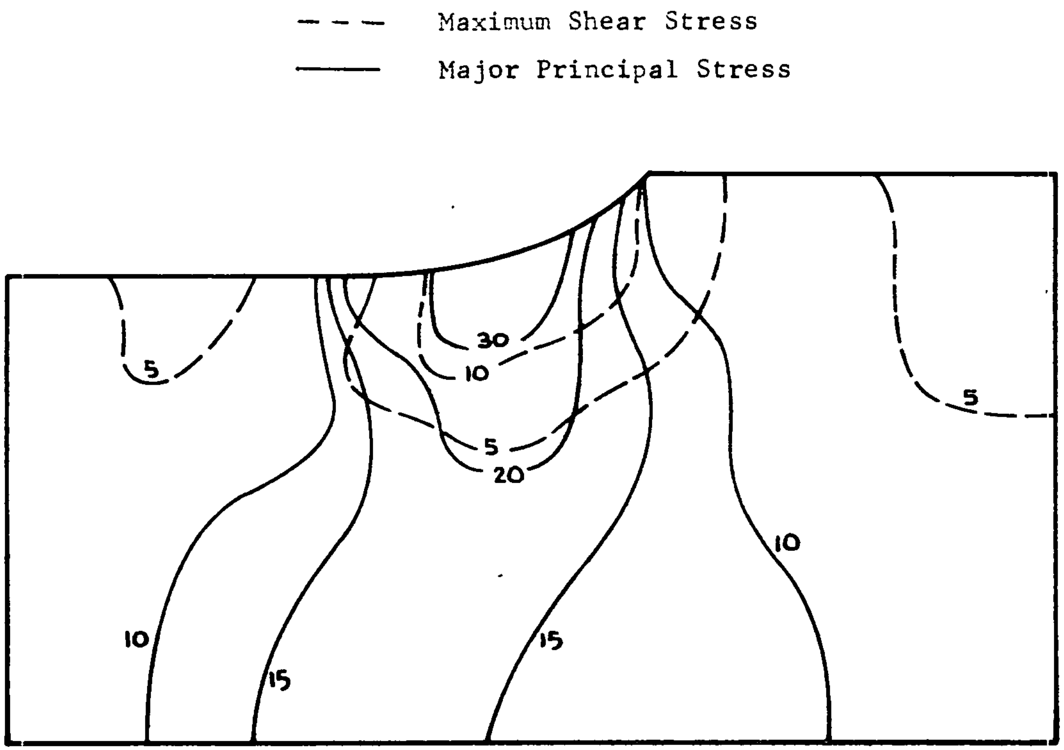


Figure 20: Isobars of Maximum Shear and Major Principal Stresses (psi) at t=0.3 sec. (Dynamic Viscoelastoplastic Analysis for 41.4% Slip)

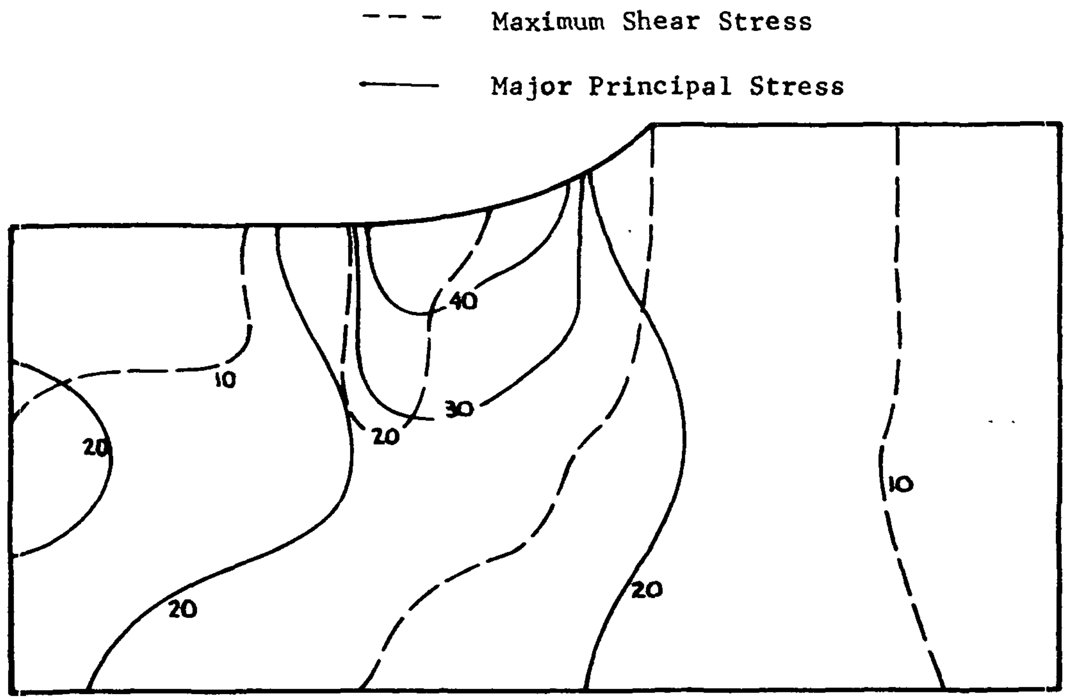


Figure 21: Isobars of Major Principal Stress and Maximum Shear Stress (Psi) at t=0.65 sec (Dynamic Viscoelastoplastic Analysis for 41.4% Slip)

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 step-stone 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 Cyclindrical 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 $\overset{(r)}{\alpha}_{ij}$

$$\overset{(r)}{\alpha}_{ij}(t) = \int_0^t \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{ij}(\tau)d\tau \qquad \text{(A-1)}$$

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

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

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

$$\overset{(r)}{\alpha}(t) = \int_0^{t-\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{ij}(\tau)d\tau + \int_{t+\Delta t}^{t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{ij}(\tau)d\tau$$

$$= \exp\left(\frac{-\Delta t}{T_{(r)}}\right) \overset{(r)}{\alpha}_{ij}(t-\Delta t) + \int_{t-\Delta t}^{t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \dot{\gamma}_{ij}(\tau)d\tau$$

$$= \exp\left(\frac{-\Delta t}{T_{(r)}}\right) \overset{(r)}{\alpha}_{ij}(t-\Delta t) + \int_{t-\Delta t}^{t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \{\dot{\gamma}_{ij}(t-\Delta t)$$

$$+ \frac{\Delta t - t + \tau}{\Delta t} [\dot{\gamma}_{ij}(t) - \dot{\gamma}_{ij}(t-\Delta t)]\}d\tau$$

$$= \exp\left(\frac{-\Delta t}{T_{(r)}}\right) {}^{(r)}\alpha_{ij}(t-\Delta t) + \int_{t-\Delta t}^{t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) \{(1-\frac{t}{\Delta t} + \frac{\tau}{\Delta t})\dot{\gamma}_{ij}(t)$$

$$+ \left(\frac{t}{\Delta t} - \frac{\tau}{\Delta t}\right) \dot{\gamma}_{ij}(t-\Delta t)\ d\tau = \exp\left(\frac{-\Delta t}{T_{(r)}}\right) {}^{(r)}\alpha_{ij}(t-\Delta t) + T_{(r)}\{[\exp\left(\frac{-(t-\tau)}{T_{(r)}}\right)$$

$$- \frac{t}{\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) + \frac{\tau}{\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) - \frac{T_{(r)}}{\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right)]_{t-\Delta t}^{t} \dot{\gamma}_{ij}(t)$$

$$+ [\frac{t}{\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) - \frac{\tau}{\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right) + \frac{T_{(r)}}{\Delta t} \exp\left(\frac{-(t-\tau)}{T_{(r)}}\right)]_{t-\Delta t}^{t} \dot{\gamma}_{ij}(t-\Delta t)\}$$

$$= \exp\left(\frac{-\Delta t}{T_{(r)}}\right) {}^{(r)}\alpha_{ij}(t-\Delta t) + T_{(r)} \Big[\{1- \exp\left(\frac{-\Delta t}{T_{(r)}}\right) - \frac{t}{\Delta t} [1- \exp\left(\frac{-\Delta t}{T_{(r)}}\right)]$$

$$+ \frac{t}{\Delta t} - \frac{t-\Delta t}{\Delta t} \exp\left(\frac{-\Delta t}{T_{(r)}}\right) - \frac{T_{(r)}}{\Delta t} + \frac{T_{(r)}}{\Delta t} \exp\left(\frac{-\Delta t}{T_{(r)}}\right)\} \dot{\gamma}_{ij}(t)$$

$$+ \{ \frac{t}{\Delta t} [1- \exp\left(\frac{-\Delta t}{T_{(r)}}\right)] - \frac{t}{\Delta t} + \frac{t-\Delta t}{\Delta t} \exp\left(\frac{-\Delta t}{T_{(r)}}\right) + \frac{T_{(r)}}{\Delta t}$$

$$- \frac{T_{(r)}}{\Delta t} \exp\left(\frac{-\Delta t}{T_{(r)}}\right) \} \dot{\gamma}_{ij}(t-\Delta t)\Big] = \exp\left(\frac{-\Delta t}{T_{(r)}}\right) {}^{(r)}\alpha_{ij}(t-\Delta t)$$

$$+ T_{(r)} \Big[\{- \exp\left(\frac{-\Delta t}{T_{(r)}}\right) + \frac{T_{(r)}}{\Delta t} [1 - \exp\left(\frac{-\Delta t}{T_{(r)}}\right) \} \dot{\gamma}_{ij}(t-\Delta t)$$

$$+ \{1 - \frac{T_{(r)}}{\Delta t} [1 - \exp\left(\frac{\Delta t}{T_{(r)}}\right) \} \dot{\gamma}_{ij}(t)\Big] = {}^{(r)}A\, {}^{(r)}\alpha_{ij}(t-\Delta t)$$

$$+ {}^{(r)}B\, \dot{\gamma}_{ij}(t-\Delta t) + {}^{(r)}C\, \dot{\gamma}_{ij}(t)$$

or

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

where

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

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

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

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

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

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

$$D = rb\{\int_{\theta_2}^{\theta_1} \tau(\theta)\cos\theta d\theta - \int_{0_2}^{\theta_1} \sigma(\theta)\sin\theta d\theta\}$$

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

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

$$\theta_M = (C_1 + C_2 i)\theta_1$$

where i is the slip (%) defined by

$$i = (1 - \frac{V}{\omega r})100$$

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

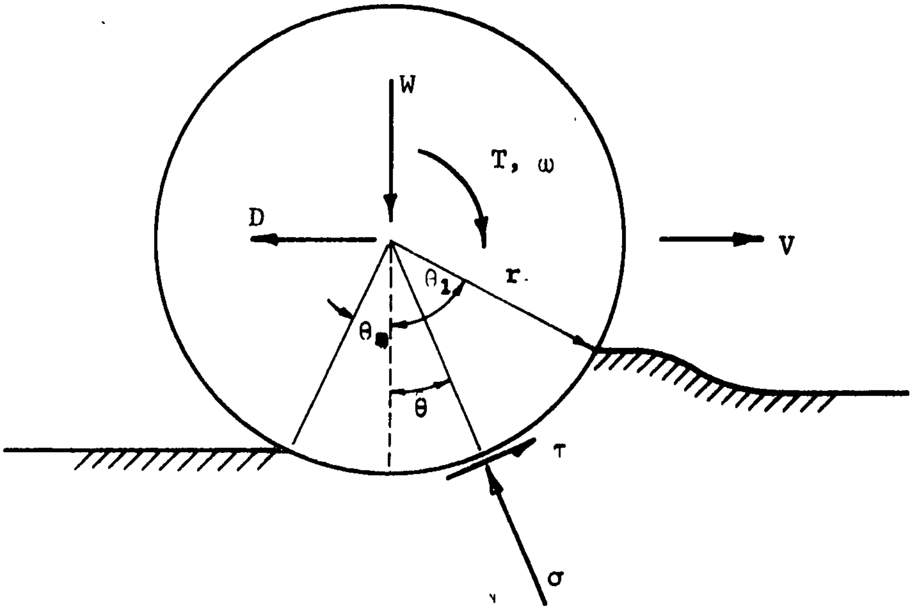


Figure 2-1: Equilibrium of a Driven Rigid Wheel on Soil

TABLE 1

Soil	Angle of Internal Friction (deg.)	Soil Cohe-sion	Density (lb/in³)	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]

$$\sigma_1(\theta) = (K_1+K_2b)(\frac{r}{6})^n(\cos\theta - \cos\theta_1)^n$$

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

$$\sigma_2(\theta) = (K_1+K_2b)(\frac{r}{6})^n\left[\cos\{\theta_1-\theta\left(\frac{(C_1+C_{21})}{C_1+C_{21}}\right)\}-\cos\theta_1\right]^n$$

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

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

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,

$$W = rb\{\int_{\theta_M}^{\theta_1} \sigma_1(\theta)\cos\theta d\theta + \int_0^{\theta_M} \sigma_2(\theta)\cos\theta d\theta$$

$$+\int_{\theta_M}^{\theta_1} \tau_1(\theta)\sin\theta d\theta + \int_0^{\theta_M} \tau_2(\theta)\sin\theta d\theta$$

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   @ASG,T 2,F/1/POS/5
000
000   C        ISW(1) = 0, STATIC ANALYSIS ONLY
000   C        ISW(1) = N, DYNAMIC ANALYSIS FOR N TIME INCREMENTS.
000   C        ISW(1) =-N, STATIC PLASTIC ANALYSIS FOR  N LOAD INCREMENT
000   C        ISW(2) =-1, VISCO-ELASTIC
000   C        ISW(2) = 0, ELASTIC
000   C        ISW(2) = 1, VISCO-ELASTO-PLASTIC
000   C        ISW(3) = M, PRINT FOR EACH M TH TIME STEP
000         PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000         COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5)
000         COMMON /BLK1/ W(4),F(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000        * HN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
000        * A0,B0,C0,IC,JC,KC,LC,NEL
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         COMMON /BLK4/ D(3,3),ATD(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),
000        * VE(8,8),AI(NELS)
000         COMMON /BLK5/ GR(NELS,3),STRAIR(NELS,3),ALPHA,BETA,GAMMA,DELT,XK,
000        * AT,XNU,E,SM,SMS,VOIDI,CAPA,RAMD,BET
000         COMMON /BLK8/ DP(NELS,3,3),PRINS(NELS,3)
000         COMMON /DYN/ CBAR(MX),XM(MX),AA(MX)
000   C
000   C
000         CALL SETUP
000   C
000         READ(5,510)           (TITLE(I),I=1,20)
000         READ(5,500) INODE,NELEM,NAPC,NBC
000         READ(5,500) ISW
000         READ(5,530) E,XNU,DENS,DEPTH,ES,XNUS
000         READ(5,530) AT,XK,DELT
000         READ(5,530) PHI,VOIDI,CAPA,RAMD
000         PHI = PHI * 3.14159 / 180.
000         SINP = SIN(PHI)
000         SM = 6.*SINP /(3.-SINP)
000         SMS = SM * SM
000         BET = RAMD - CAPA
000         WRITE(6,630)  INODE,NELEM,NAPC,NBC
000         WRITE(6,610) E,XNU,DENS,AT,XK,DELT
000         WRITE(6,640) PHI,VOIDI,CAPA,RAMD
000         II = 0
000         DO 101 I = 1,INODE
000         DO 101 J = 1,2
000          II = II + 1
000     101 ID(I,J) = II
000         WRITE(6,600) (TITLE(I),I=1,20)
000         DO 100 I = 1,INODE
000         READ(5,520) Z(I),R(I)   ,IZ,IR
000         IF(IZ.NE.0) ID(I,1) = 0
000         IF(IR.NE.0) ID(I,2) = 0
000     100 WRITE(6,620) I,Z(I),R(I) ,IZ,IR
000   C
000   C
000         CALL INIT
000         NBC = INODE
```

```
000          NFREE = INODE * 2
000          NACT = NFREE
000   C
000          WRITE(6,650)
000          READ(5,550) ((IJKL(I,J),J=1,4),I=1,NELEM)
000          WRITE(6,660) (I,(IJKL(I,J),J=1,4),I=1,NELEM)
000          DO 800 NEL = 1,NELEM
000          DO 700 K = 1,4
000          IN = IJKL(NEL,K)
000          KK(NEL,K) = ID(IN,1)
000      700 KK(NEL,K+4) = ID(IN,2)
000          NELI = ISW(5) + 1
000          E = ES
000          XNU = XNUS
000          IF(NEL.EQ.NELI) CALL INIT
000          DO 790 I = 1,3
000
000
000
000          DO 790 J = 1,3
000      790 DF(NEL,I,J) = D(I,J)
000          DO 799 I = 1,2
000          II = I + 1
000          DO 799 J = II,4
000          IDIF = IJKL(NEL,I) - IJKL(NEL,J)
000          IF(IDIF.LT.0) IDIF = -IDIF
000      799 IF(IDIF.GT.IMAX) IMAX = IDIF
000      800 CONTINUE
000   C      IMAX = MAX DIFFERENCE IN ADJACENT NODE NO.
000          IHB = (IMAX + 1) * 2
000          IHBI = IHB - 1
000          LT = IHB * IHBI / 2
000          LAST = LT + (NFREE - IHBI) * IHB
000          WRITE(6,641) IMAX,IHB,LT,LAST
000   C
000          WRITE(6,677)
000          DO 920 NC = 1,NAPC
000           READ(5,540) NODE,PZ,PR
000           WRITE(6,688)NODE,PZ,PR
000           II = (NODE-1) * 2
000            P(II+1) = PZ
000            P(II+2) = PR
000      920 CONTINUE
000          IF(ISW(1).GT.0) CALL DIAGNL(XM ,P,NFREE,IHB,IHBI,LT,LAST)
000          IF(ISW(1).LE.0) CALL DIAGNL(XKL,P,NFREE,IHB,IHBI,LT,LAST)
000   C
000          CALL STIFF1(DENS)
000   C
000          WRITE(6,693) D
000   C
000   C
000          IF(ISW(1).GT.0) GO TO 930
000   C
000          CALL ZERO(QB,NELS,3)
000          IF(ISW(1).LT.0) CALL PLASTC
000   C
000          CALL DISPL(1)
```

```
000            CALL STRAIN(1)
000            STOP ELASTC
000      C
000      C
000        930 CONTINUE
000      C
000            CALL STEP
000      C
000      C
000       1000 STOP
000      C
000        500 FORMAT(10I5)
000        510 FORMAT(20A4)
000        520 FORMAT(5X,2F10.4,2I5)
000        530 FORMAT(6F15.6)
000        540 FORMAT(I5,2F10.4)
000        550 FORMAT(5X,4I5)
000        600 FORMAT(1H1,20X,20A4///30X,'COORDINATE VALUES'//T11,'NODE',T30,'Z-C
000           *OORD',T50,'R-COORD',T65,'0,IF FREE TO Z',3X,'0,IF FREE TO R'//)
000        610 FORMAT(//5X,'E,  XNU,  DENS,  AT,  XK,  DELT,' ,6E15.7//)
000        620 FORMAT(I10,I5,T25,F10.4,T45,F10.4,T65,2(I8,5X))
000        630 FORMAT(//5X,'INODE,  NELEM,  NAPL,  NBC,',4I5)
000        640 FORMAT(5X,'PHI,  VOIDI,  CAPA,  RAMDA,',4E15.6)
000        641 FORMAT(//5X,'IMAX =',I5,'  IHB =',I5,'  LT =',I5,'  LAST =',I5)
000        650 FORMAT(//5X,'CONNECTIVITY'/)
000        660 FORMAT(5I8)
000        677 FORMAT(///10X,'APPLIED C. LOAD'//5X,'NODE',10X,'FORCE TO Z',5X,'
000           * FORCE TO R'//)
000        678 FORMAT(5X,I5,2(5X,F12.4))
000        689 FORMAT(1H1,10X,'  TOTAL DISPLACEMENT      NO. OF INCREMENTAL STEPS
000           *=',2I5//5X,'NODE',5X,'Z - DISPL',20X,'R - DISPL'//)
000        690 FORMAT(5X,I5,2E15.7)
000        691 FORMAT(///,10X,' TOTAL STRESSES'//5X,'ELEM',5X,'SIGMA - Z',10X,
000           *'SIGMA - R',10X,'TANGENTIAL',10X,'TAU - ZR'//)
000        692 FORMAT(5X,I5,3X,4F17.6)
000        693 FORMAT(3E15.6)
000            END
000      @ELT,SIH NASA*IPPS.INIT,,,165532132410
000            SUBROUTINE INIT
000            PARAMETER NET=150,NELS=150,MX= 5000,NF=NET*2
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),AI(NELS)
000            COMMON /BLK5/ GP(NELS,3),STRAIN(NELS,3),ALPHA,BETA,GAMMA,DELT,XK,
000           * AT,XNU,E,SM,SMS,VOIDI,CAPA,RAMD,BET
000            DIMENSION GD(3,3),DV(3,3),ATA(3,3)
000            CONST = E*XNU / ((1.+XNU)*(1.-XNU*2.))
000            SHEAR = E / (2.*(1.+XNU))
000            D(1,1) = CONST + SHEAR*2.
000            D(2,2) = D(1,1)
000            D(1,2) = CONST
000            D(2,1) = CONST
000            D(3,3) = SHEAR
000            TI = AT/XK
000            LSUM = 3
000            CALL ZERO(ATA,3,3)
000            CALL ZERO(DV,3,3)
000            DO 50 I = 1,3
```

```
000      50  DV(I,J) = XK                                                  00002000
000          ALPHA=0.0                                                     00002100
000          BETA =0.0                                                     00002200
000          GAMMA=0.0                                                     00002300
000          IF(ABS(TI).LT.0.000001) GO TO 20                              00002400
000          DT= -DELT/TI                                                  00002500
000          ALPHA= EXP(DT)                                                00002600
000          BETA= TI*(-ALPHA-(1.-ALPHA)/DT)                               00002700
000          GAMMA= TI*(1.+(1.-ALPHA)/DT)                                  00002800
000       20 CONTINUE                                                      00002900
000          DO 6 I=1,3                                                    00003000
000          DO 6 J=1,3                                                    00003100
000          AD(I,J)= 0.                                                   00003200
000          BD(I,J)=0.                                                    00003300
000          GD(I,J)=0.                                                    00003400
000          DO 7K=1,LSUM                                                  00003500
000          AD(I,J)= AD(I,J) +DV (I,J)*ALPHA                              00003600
000          BD(I,J)= BD(I,J)+DV (I,J)*BETA                                00003700
000        7 GD(I,J)=GD(I,J)+DV (I,J)*GAMMA                                00003800
000          DO 8 L=1,3                                                    00003900
000        8 ATD(L,L)=ATD(L,L)+GD(L,L)                                     00004000
000        6 CONTINUE                                                      00004100
000          RETURN                                                        00004200
000          END                                                           00004300
000    @ELT,SIP NASA*IPF$.STIFF1,,,165535132410
000          SUBROUTINE STIFF1(DENS)                                       00000100
000          PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2                  00000200
000          COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MAC1,ISW(5)      00000300
000          COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),  00000400
000         * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),  00000500
000         * A0,B0,C0,IC,JC,KC,LC,NEL                                     00000600
000          COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFT),Z(NFT),KK(NELS,8)  00000700
000          COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHB,IHB1,LT,LAST,NFREE       00000800
000          COMMON /BLK4/ D(3,3),ATD(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),  00000900
000         * VE(8,8),AI(NELS)                                             00001000
000          COMMON /BLK6/ BT(NELS,8,3),ARM(NELS,4),AZM(NELS,4),AOJ(NELS)  00001100
000          COMMON /BLK8/ UP(NELS,3,3),PRINS(NELS,3)                      00001200
000          COMMON /BLK9/M,N                                              00001300
000          COMMON /DYN/ CBAR(MX),XM(MX),CC(MX)                           00001400
000    C                                                                   00001500
000    C                                                                   00001600
000          DO 900 NEL = 1,NELEM                                          00001700
000    C                                                                   00001800
000          IC = IJKL(NEL,1)                                              00001900
000          JC = IJKL(NEL,2)                                              00002000
000          KC = IJKL(NEL,3)                                              00002100
000          LC = IJKL(NEL,4)                                              00002200
000    C                                                                   00002300
000          CALL AREAA(NEL,AREA)                                          00002400
000          AI(NEL) = AREA                                                00002500
000    C                                                                   00002600
000          AZ(1) = Z(LC) - Z(JC)                                         00002700
000          AZ(2) = Z(IC) - Z(KC)                                         00002800
000          AZ(3) = -AZ(1)                                                00002900
000          AZ(4) = -AZ(2)                                                00003000
000          BZ(1) = Z(KC) - Z(LC)                                         00003100
000          BZ(2) = -BZ(1)                                                00003200
```

```
000          BZ(3) = Z(JC) - Z(IC)
000          BZ(4) = -BZ(3)
000          CZ(1) = Z(JC) - Z(KC)
000          CZ(2) = Z(LC) - Z(IC)
000          CZ(3) = -CZ(2)
000          CZ(4) = -CZ(1)
000          AR(1) = R(JC) - R(LC)
000          AR(2) = R(KC) - R(IC)
000          AR(3) = -AR(1)
000          AR(4) = -AR(2)
000          BR(1) = R(LC) - R(KC)
000          BR(2) = -BR(1)
000          BR(3) = R(IC) - R(JC)
000          BR(4) = -BR(3)
000          CR(1) = R(KC) - R(JC)
000          CR(2) = R(IC) - R(LC)
000          CR(3) = -CR(2)
000          CR(4) = -CR(1)
000          DO 100 I = 1,4
000          AZ(I) = -AZ(I)
000          BZ(I) = -BZ(I)
000          CZ(I) = -CZ(I)
000          BR(I) = -BR(I)
000          CR(I) = -CR(I)
000          AR(I) = -AR(I)
000          ARM(NEL,I) = AR(I)
000          AZM(NEL,I) = AZ(I)
000      100 CONTINUE
000          AO = AR(3)*AZ(2) - AR(4)*AZ(1)
000          BO = BR(2)*BZ(4) - BR(3)*BZ(1)
000          CO = CR(3)*CZ(1) - CR(4)*CZ(2)
000          AOJ(NEL) = AO
000     C
000          DO 200 M = 1,4
000          DO 190 N = 1,4
000           CALL GAUSS(1,AA)
000          TYPEA(M,N) = AA
000           CALL GAUSS(2,AA)
000          TYPEB(M,N) = AA
000           CALL GAUSS(3,AA)
000          TYPEC(M,N) = AA
000          CALL GAUSS(4,AA)
000          TYPED(M,N) = AA
000      190 CONTINUE
000          CALL GAUSS(5,AA)
000          ET(NEL,M,1) = AA
000          ET(NEL,M+4,3) = AA / 2.
000          CALL GAUSS(6,AA)
000          ET(NEL,M+4,2) = AA
000          ET(NEL,M,3) = AA / 2.
000      200 CONTINUE
000     C
000          WRITE(2) TYPEA,TYPEB,TYPEC,TYPED
000     C
000          D11 = D(1,1)
000          DO 300 I= 1,4
000          DO 300 J = 1,4
```

```
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)*TYPEB(J,I)
000         STIFF(J+4,I) = STIFF(I,J+4)
000         STIFF(I+4,J+4) = DP(NEL,2,2)*TYPEC(I,J)+DP(NEL,3,3)*TYPEA(I,J)
000   C
000         CM(I,J) = DENS * 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(J,I)
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         II = KK(NEL,I)
000         DO 110 J = 1,8
000         JJ = KK(NEL,J)
000         IF(II.EQ.0.OR.JJ.EQ.0) GO TO 110
000         IF(II.LT.JJ) GO TO 110
000         IF(II.GT.IHBI) GO TO 104
000         L = JJ + (II-1) * II / 2
000         GO TO 105
000     104 L = JJ + LT + (II-IHB) * IHBI
000     105 XKL(L) = XKL(L) + STIFF(I,J)
000         XM(L) = XM(L) + CM(I,J)
000         CBAR(L) = CBAR(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   @ELT,SIH NASA*IPF$.DISPL,,,103543132410
000         SUBROUTINE DISPL(NI)
000         PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000         COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,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         MM = LAST + 1
000         NN = LAST + NFREE
000         WRITE(6,667) (XKL(I),I=MM,NN)
000     667 FORMAT(9E14.6)
000   C
000         CALL FACTOR(XKL,IHB,IHBI,LT,LAST,NFREE)
000   C
000         CALL SOLIN (XKL,IHB,IHBI,LT,LAST,NFREE)
000   C
```

```
000         WRITE(6,600)
000         DO 280 I = 1,NBC
000         J = (I-1)*2 + 1 + LAST
000         JJ = J + 1
000     280 WRITE(6,610) I,XKL(J),XKL(JJ)
000     600 FORMAT(/// 10X,'GLOBAL DISPLACEMENT'//5X,'NODE',
000        110X,'Z-DISPL',10X,'R-DISPL')
000     610 FORMAT(5X,I4,2E20.7)
000         RETURN
000         END
000     WELT,SIH NASA*IPF9.STRAIN,,,226151130610
000         SUBROUTINE STRAIN(NT)
000         PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000         COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5)
000         COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000        * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
000        * AO,BO,CO,IC,JC,KC,LC,NFL
000         COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFT),Z(NFT),KK(NELS,8)
000         COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHP,IHBI,LT,LAST,NFREE
000         COMMON /BLK4/DE(3,3),AID(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),
000        * VE(8,8),AI(NELS) \
000         COMMON /BLK5/ GR(NELS,3),STRAIB(NELS,3),ALPHA,BETA,GAMMA,DELT,XX,
000        * AT,XNU,E,SM,SMS,VOIDI,CAPA,RAMD,BET
000         COMMON /BLK6/ BT(NELS,8,3),ARM(NELS,4),AZM(NELS,4),AOJ(NELS)
000         COMMON /BLK8/ DP(NELS,3,3),PRINS(NELS,3)
000         DIMENSION U(4),V(4),D(3,3)
000     C
000         WRITE(6,600) NI
000     C
000         DO 900 NEL = 1,NELEM
000         DO 200 I = 1,4
000         IN = IJKL(NEL,I)
000         I1 = (IN-1) * 2 + 1  + LAST
000         V(I) = XKL(I1)
000     200 U(I) = XKL(I1+1)
000     620 FORMAT(4E15.6)
000     C
000         DO 210 I = 1,3
000         DO 210 J = 1,3
000     210 D(I,J) = DP(NEL,I,J)
000     C
000     C   COMPUTE STRESSES AT THE CTR.
000     C
000         EZ = 0.
000         ER = 0.
000         GM = 0.
000         DO 100 I = 1,4
000         EZ = EZ - ARM(NEL,I) * V(I) / AOJ(NEL)
000         ER = ER - AZM(NEL,I) * U(I) / AOJ(NEL)
000     100 GM = GM - (ARM(NEL,I) * U(I) + AZM(NEL,I) * V(I) ) / AOJ(NEL)
000         SIGMZ = D(1,1) * EZ + D(1,2) * ER
000         SIGMR = D(2,2) * ER + D(2,1) * EZ
000         SHEAR = D(3,3) * GM
000     650 FORMAT(I5,7F15.6)
000     C
000         GR(NEL,1) = SIGMZ + GR(NEL,1)
000         GR(NEL,2) = SIGMR + GR(NEL,2)
```

```
000          GP(NEL,3) = SHEAR + GB(NEL,3)
000          STRAIN(NEL,1) = SIGMZ
000          STRAIN(NEL,2) = SIGMR
000          STRAIN(NEL,3) = SHEAR
000    C
000    C        COMPUTE PRINCIPAL STRESS.( COMP. IS POSITIVE HERE)
000    C
000          SGR = (((GB(NEL,1) - GB(NEL,2)) / 2.) **2 + GB(NEL,3)**2)**.5
000          SGI =  (GB(NEL,1) + GB(NEL,2) ) / 2.
000          PRINS(NEL,1) = SGI + SGR
000          PRINS(NEL,2) = SGI - SGR
000          PRINS(NEL,3) = SGR
000      900 CONTINUE
000    C
000          WRITE(6,610) (I,(GB(I,J),J=1,3),(PRINS(I,K),K=1,3),I=1,NELEM)
000      600 FORMAT(1H1,10X,'TOTAL STRESS AT THE END OF',I4,'  LOAD INCREMENT'/
000         1 5X,'ELEM',10X,'STRESS AT CTR',30X,'PRINCIPAL STRESS'/)
000      610 FORMAT(I8,3E13.5,5X,3E13.5)
000          RETURN
000          END
000    @ELT,SIH NASA*IPFS.PLASTC,,,163544132410
000          SUBROUTINE PLASTC
000          PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000          COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5)
000          COMMON /BLK2/ IF(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          COMMON /BLK4/ D(3,3),AID(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),
000         * VE(8,8),AI(NELS)
000          COMMON /BLK7/ UDB(NF),UDDB(NF),FOV(NF),ULD(NF),DB(NF),DISP(NF),
000         * DDOT(NF)
000          COMMON /DYN/ CBAR(MX),XM(MX),AA(MX)
000    C
000          CALL ZERO(AA,MX,1)
000          NLOAD = -ISW(1)
000          SCAL = NLOAD
000          DO 100 I = 1,MACT
000          P(I) = P(I) / SCAL
000      100 XKL(LAST+I) = P(I)
000    C
000          DO 900 NI = 1,NLOAD
000    C
000          IF(NI.EQ.1) GO TO 800
000          CALL DIAGNL(XKL,P,NFREE,IHB,IHBI,LT,LAST)
000    C
000          CALL STIFF2
000    C
000      800 CALL DISPL(NI)
000          DO 120 I = 1,NFREE
000          AA(I) = AA(I) + XKL(LAST+I)
000      120 DISP(I) = AA(I)
000    C
000          WRITE(6,600) NI
000          DO 130 I = 1,INODE
000          JJ = I * 2
000          II = JJ - 1
000      130 WRITE(6,610) I,AA(II),AA(JJ)
000    C
```

```
000         CALL STRAIN(NI)
000   C
000     900 CONTINUE
000   C
000     600 FORMAT(1H1,5X,'TOT. DISPL. AT THE END OF',I4,'  LOAD INCREMENT'//
000        1 5X,'NODE',8X,'Z-DISPL',15X,'R-DISPL'/)
000     610 FORMAT(I10,2E20.7)
000         STOP
000         END
000   WELT,SIH NASA*TPF$.STIFF2,,,163551132410
000         SUBROUTINE STIFF2
000         PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000         COMMON /BLK0/ TITLE(20),INODE,NFLEM,NAPC,NHC,MACT,ISW(5)
000         COMMON /BLK1/ W(4),H(4),AR(4),PR(4),CR(4),AZ(4),BZ(4),CZ(4),
000        * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
000        * AO,BO,CO,IC,JC,KC,LC,NEL
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         COMMON /BLK4/ D(3,3),AIP(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),
000        * VE(8,8),AI(NELS)
000         COMMON /BLK5/ QB(NELS,3),STRAIB(NELS,3),ALPHA,BETA,GAMMA,DELT,XK,
000        * AT,XNU,E,SM,SMS,VOIDI,CAPA,RAND,BET
000         COMMON /BLK6/ [illegible],3),PRINS(NELS,3)
00    C
000         REWIND 2
000   C
000         XNUS = XNU * [illegible]
000         VA = XNU + 1.
000         VB = 1.+XNUS-X[illegible]
000         VC = 2.*(XNUS-X[illegible] - 1.
000   C
000         DO 900 NEL = 1,NFLEM
000         READ(2)  TYPEA,TYPEB,TYPEC,TYPED
000         IF(NEL.LE.ISW(5)) GO TO 800
000         DO 200 I = 1,3
000         DO 200 J = 1,3
000     200 DP(NEL,I,J) = D(I,J)
000   C
000         SIGZ = QB(NEL,1) + STRAIB(NEL,1)/2.
000         SIGR = QB(NEL,2) + STRAIB(NEL,2)/2.
000         TAUZ = QB(NEL,3) + STRAIB(NEL,3)/2.
000         DELP = VA * (STRAIB(NEL,1)+STRAIB(NEL,2))/3.
000         PP = VA * (SIGZ+SIGR) / 3.
000         PSQ = PP * PP
000         SZZ = 2.*VB*SIGZ + VC*SIGR
000         SRR = 2.*VB*SIGR + VC*SIGZ
000         SZR = 6.*TAUZ
000         TJ = (VB*SIGZ*SIGZ+VB*SIGR*SIGR+VC*SIGZ*SIGR)/3. + TAUZ*TAUZ
000         CALL AREAA(NEL,AREA)
000         RATIO = AREA / AI(NEL)
000         VOIDR = RATIO * (1.+VOIDI) - 1.
000         VDR = VOIDR + 1.
000         TTJ = 3.*TJ
000         ETM = 1.+TTJ/(PSQ*SMS)
000         POW = 1.-CAPA/RAND
000         PO = PP*ETM**POW
000         AA = SMS * (2.*PP-PO) / 3.
```

```
      BB = 3.*AA*VOR*SMS*PP*PO / BET
      AR(1) = SZZ + AA
      AR(2) = SRR + AA
      AR(3) = SZR
      BR(1) = SZZ
      BR(2) = SRR
      BR(3) = SZR
      DFI = 0.
      DFK = 0.
      DO 220 I = 1,3
      DFI = DFI + AR(I) * STRAIB(NEL,I)
  220 DFK = DFK + BR(I) * STRAIB(NEL,I)
      DFJ = PO * SMS * DELP
      POW = -CAPA/RAMD
      DFK = ETM **POW * (DFK-2.*TTJ/PP*DELP)
      DF = DFI - DFJ - DFK * (1.-CAPA/RAMD)
      ASQ = SMS * PP * (PO-PP)
      ETA = SQRT(TTJ) / PP
      WRITE(6,620) VOIDR,TTJ,ASQ,DF,NEL,SM,ETA
  620 FORMAT(5X,'VOIDR=',F10.5,'  3J=',E12.5,'  ASQ=',E12.5,'  DF=',
     1 E12.5, I7,'  M=',E12.5,'  ETA=',E12.5)
      IF(DF.LT.0.)  GO TO 800
      DO 300 I = 1,3
      BR(I) = 0.
      CR(I) = 0.
      DO 300 J = 1,3
      BR(I) = BR(I) + D(I,J)*AR(J)
  300 CR(I) = CR(I) + AR(J)*D(J,I)
      DEN = 0.
      DO 310 I = 1,3
  310 DEN = DEN + AR(I)*BR(I)
      DEN = DEN + BB
      DO 320 I = 1,3
      DO 320 J = 1,3
  320 DP(NEL,I,J) = D(I,J) - BR(I)*CR(J) /DEN
C
  800 CONTINUE
      DO 100 I = 1,4
      DO 100 J = 1,4
      STIFF(I,J) = DP(NEL,1,1)*TYPEA(I,J)+DP(NEL,3,3)*TYPEC(I,J)
      STIFF(I,J+4) = DP(NEL,1,2)*TYPEB(I,J)+DP(NEL,3,3)*TYPEB(J,I)
      STIFF(J+4,I) = STIFF(I,J+4)
  100 STIFF(I+4,J+4) = DP(NEL,2,2)*TYPEC(I,J)+DP(NEL,3,3)*TYPEA(I,J)
C
      DO 210 I = 1,8
      II = KK(NEL,I)
      DO 210 J = 1,8
      JJ = KK(NEL,J)
      IF(II.EQ.0.OR.JJ.EQ.0) GO TO 210
      IF(II.LT.JJ) GO TO 210
      IF(II.GT.IHBI) GO TO 214
      L = JJ + (II-1) * II / 2
      GO TO 215
  214 L = JJ + LT + (II-IHB) * IHBI
  215 XKL(L) = XKL(L) + STIFF(I,J)
  210 CONTINUE
  900 CONTINUE
```

```
000   C
000         RETURN
000         END
000   @ELT,SIH NASA*IPF3.STFP,,,163560132410
000         SUBROUTINE STFP
000         PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000         COMMON /BLK1/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5)
000         COMMON /BLK2/ XKL(MX),P(NF),IMAX,IHB,IHB1,LT,LAST,NFREE
000         COMMON /BLK5/ GB(NELS,3),STRAIB(NELS,3),ALPHA,BETA,GAMMA,DELT,XK,
000        * AT,XMU,E,SM,SMS,VOIDI,CAPA,RAMD,BET
000         COMMON /BLK7/ UDB(NF),UDDB(NF),FOV(NF),UDD(NF),DB(NF),DISP(NF),
000        * DDOT(NF)
000         COMMON /BLK8/ DP(NELS,3,3),SIGB(NELS,3)
000         COMMON /DYN/ CBAR(MX),XM(MX),AA(MX)
000         NTIME=1
000         NDELT = ISW(1)
000         N = MACT
000         NPRNT = ISW(3)
000         IKOUNT = 250
000         KOUNT = IKOUNT
000         IPLT = 1
000         CALL ZERO(SIGB,NELS,3)
000   C
000         DO 2 I = 1,LAST
000     2   AA(I) = XM(I) + CBAR(I) * DELT / 2. + XKL(I) * DELT * DELT / 4.
000         DO 150 I = 1,NFREE
000     150 AA(I+LAST) = P(I)
000   C
000         CALL FACTOR(AA,IHB,IHB1,LT,LAST,NFREE)
000         CALL SOLTN (AA,IHB,IHB1,LT,LAST,NFREE)
000   C
000         DO 3 I=1,N
000         UDB(I)=0.
000         UDDB(I)=0.
000         FOV(I)=0.
000         UDD(I) = AA(LAST+I)
000     3   DB(I)=0.
000         DO 4 I=1,N
000         DISP(I)= DB(I) + DELT*UDB(I) +(UDDB(I) +UDD(I))*DELT**2/4.
000     4   DDOT(I)= UDB(I) + (UDDB(I) +UDD(I))*DELT   /2.
000         WRITE(6,100) NTIME
000     100 FORMAT (25X,'DISPLACEMENTS FOR TIME INCREMENT',I5,5X,'LINEAR')
000         WRITE (6,6997) (DISP(I),I=1,N)
000    6997 FORMAT(8E15.6)
000     101 FORMAT(8E15.6)
000         WRITE(6,102)
000     102 FORMAT('VELOCITY')
000         WRITE(6,101) (DDOT(I),I=1,N)
000   C
000   C     INITIALIZE DB,STRAIB
000   C
000         CALL ZERO(STRAIB,NELS,3)
000         CALL ZERO(GB,NELS,3)
000   C
000         CALL STAN(NTIME)
000   C
000   C     NOW READY FOR NEW INCREMENTS..
```

```
C
   10 CONTINUE
C
      JJ = IHB1
      DO 200 I = 1,NFREE
      JJ = JJ + 1
      SUN = 0.
      SUM = 0.
      IF(I.GE.IHB) GO TO 180
      L = (I+1)*I / 2 - 1
      DO 170 J = 1,JJ
      IF(J.LE.I) L = L + 1
      IF(J.GT.I.AND.J.LE.IHB) L = L + J - 1
      IF(J.GT.IHB) L = L + IHBI
      SUN = SUN + XM(L)*DB(J)
  170 SUM = SUM + CBAR(L)*(UDB(J)+UDDB(J)*DELT/2.) + XKL(L)*(DB(J)+DELT*
     1UDB(J)+UDDB(J)*DELT*DELT/4.)
      GO TO 199
  180 II = I - IHB + 1
      L = LT + (II-1) * IHB
      IF(JJ.GT.NFREE) JJ = NFREE
      DO 190 J = II,JJ
      IF(J.LE.I) L = L + 1
      IF(J.GT.I) L = L + IHBI
      SUN = SUN + XM(L)*DB(J)
  190 SUM = SUM + CBAR(L)*(UDB(J)+UDDB(J)*DELT/2.) + XKL(L)*(DB(J)+DELT*
     1UDB(J)+UDDB(J)*DELT*DELT/4.)
  199 XM(LAST+I) = SUM
      CBAR(LAST+I) = SUN
  200 CONTINUE
C
      IF(NTIME.GT.NDELT) GO TO 113
      DO 12 I = 1,NFREE
   12 AA(I+LAST) =  P(I) - FOV(I) -XM(LAST+I) - CBAR(LAST+I)
C
  300 CALL SOLTN (AA,IHB,IHB1,LT,LAST,NFREE)
C
      DO 13 I=1,N
      FOV(I)=0.
      UDD(I) = AA(LAST+I)
      DISP(I)= DB(I) + DELT*UDB(I) +(UDDB(I) +UDD(I))*DELT**2/4.
   13 DDOT(I)= UDB(I) + (UDDB(I) +UDD(I))*DELT   /2.
C
      CALL STAN(NTIME)
C
      IF(NTIME.NE.NPRNT) GO TO 310
      NPRNT = NPRNT + ISW(3)
      WRITE(6,105) NTIME
  105 FORMAT(10X,'DISPLACEMENT AT NTIME =',I5/5X,'NODE NO      Z-DISPL
     1         R-DISPL')
      DO 106 I = 1,INODE
      JJ = I * 2
      II = JJ - 1
  106 WRITE(6,620) I,DISP(II),DISP(JJ)
  620 FORMAT(I9,2E15.6)
      II = LAST + 1
      JJ = LAST + NFREE
      WRITE(6,600)
      WRITE(6,101) (AA(I),I=II,JJ)
      WRITE(6,610)
      WRITE(6,101) (CBAR(I),I=II,JJ)
  600 FORMAT(/5X,'TOT.FORCE'/)
  610 FORMAT(/5X,'FP'/)
  310 CONTINUE
      GO TO 10
C
  113 WRITE(6,1968)
 1968 FORMAT(//2X,'END OF ANALYSIS'///)
      RETURN
      END
@ELT,SIH NASA*IPF$.STAN,,,163570132410
      SUBROUTINE STAN(NTIME)
      PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
      COMMON /BLK0/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5)
      COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFT),Z(NFT),KK(NELS,8)
      COMMON /BLK3/ XKL(MX),P(NF),IMAX,IHB,IHBI,LT,LAST,NFREE
      COMMON /BLK4/ D(3,3),ATD(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),
     * VE(8,8),AI(NELS)
      COMMON /BLK5/ GP(NELS,3),STRAIB(NELS,3),ALPHA,BETA,GAMMA,DELT,XK,
     * AT,XNU,E,SM,SMS,VOIDI,CAPA,RAND,BET
      COMMON /BLK6/ BT(NELS,8,3),ARM(NELS,4),AZM(NELS,4),AOJ(NELS)
      COMMON /BLK7/ UDB(NF),UDDB(NF),FOV(NF),UDD(NF),DB(NF),DISP(NF),
     * UDOT(NF)
      COMMON /BLK8/ DP(NELS,3,3),SIGR(NELS,3)
      COMMON /DYN/ CBAR(MX),XM(MX),AA(MX)
      DIMENSION V(4),U(4),VD(4),UD(4),EPS(3),EPSD(3),SIGE(3),SIGV(3),
     * SIGF(3), SIGT(3),FV(8)
C
      CALL ZERO(XM,MX,1)
      REWIND 2
      NN = NTIME / ISW(3)
      MM = NN * ISW(3)
      IRITE = 0
      IF(MM.EQ.NTIME) IRITE = 1
      IF(MM.EQ.NTIME) WRITE(6,600) NTIME
      DO 900 NEL = 1,NELEM
      DO 100 I = 1,4
      IN = IJKL(NEL,I)
      II = (IN-1)*2 + 1
      V(I) = DISP(II)
      U(I) = DISP(II+1)
      VD(I) = UDOT(II)
      UD(I) = UDOT(II+1)
  100 CONTINUE
      CALL ZERO(EPS,3,1)
      CALL ZERO(EPSD,3,1)
      DO 110 I = 1,4
      EPS(1) = EPS(1) + ARM(NEL,I)*V(I) / AOJ(NEL)
      EPS(2) = EPS(2) + AZM(NEL,I)*U(I) / AOJ(NEL)
      EPS(3) = EPS(3)+(ARM(NEL,I)*U(I)+AZM(NEL,I)*V(I))     /AOJ(NEL)
      EPSD(1) = EPSD(1) + ARM(NEL,I)*VD(I) / AOJ(NEL)
      EPSD(2) = EPSD(2) + AZM(NEL,I)*UD(I) / AOJ(NEL)
  110 EPSD(3) = EPSD(3)+(ARM(NEL,I)*UD(I)+AZM(NEL,I)*VD(I))     /AOJ(NEL)
      DO 120 I = 1,3
```

```
000             SIGE(I) = 0.
000             SIGV(I) = 0.
000             SIGF(I) = 0.
000             DO 119 J = 1,3
000             SIGE(I) = SIGE(I) + D(I,J)*EPS(J)
000             SIGV(I) = SIGV(I) + ATD(I,J)*EPSD(J)
000         119 SIGF(I) = SIGF(I) + BD(I,J)*STRAIB(NEL,J) + AD(I,J)*QB(NEL,J)
000         120 SIGT(I) = SIGE(I) + SIGV(I) + SIGF(I)
000       C
000       C     COMPUTE PRINCIPAL STRESSES
000       C     COMPRESSION IS POSITIVE HERE
000       C
000             SQR = (((SIGT(1)-SIGT(2))/2.)**2 + SIGT(3)**2) **.5
000             SGI =-(SIGT(1) + SIGT(2)) / 2.
000             EPS(1) = SGI + SQR
000             EPS(2) = SGI - SQR
000             EPS(3) = SQR
000             IF(MM.EQ.NTIME) WRITE(6,610) NEL,SIGE,SIGV,SIGT,EPS
000       C
000             IF(ISW(2).EQ.1) CALL PSTIFF(NEL,SIGE,IPLST,IRITF)
000             IF(ISW(2).EQ.0) GO TO 900
000       C
000       C        COMPUTE VISC. FORCE FOR EACH ELEMENT
000       C
000             DO 130 I = 1,8
000             FV(I) = 0.
000             DO 130 J = 1,3
000         130 FV(I) = FV(I) + BT(NEL,I,J) * SIGF(J)
000       C        NOW ASSEMBLE IN GLOBAL FORM
000             DO 140 I = 1,8
000             II = KK(NEL,I)
000             IF(II.EQ.0) GO TO 140
000             FOV(II) = FOV(II) + FV(I)
000             IF(ISW(2).NE.1.OR.IPLST.NE.1) GO TO 140
000             DO 139 J = 1,8
000             JJ = KK(NEL,J)
000             IF(JJ.EQ.0) GO TO 139
000             IF(II.LT.JJ) GO TO 139
000             IF(II.GT.IHBI) GO TO 137
000             L = JJ+(II-1)*II/2
000             GO TO 138
000         137 L = JJ + LT + (II-IHB)*IHBI
000         138 XM(L) = XM(L) + STIFF(I,J)
000         139 CONTINUE
000         140 CONTINUE
000       C
000       C        UPDATE (QB) AND (STRAIB). (ATRAIN RATE)
000       C
000             DO 150 I = 1,3
000             SIGB(NEL,I) = -SIGE(I)
000             QB(NEL,I) = ALPHA*QB(NEL,I)+BETA*STRAIB(NEL,I)+GAMMA*EPSD(I)
000         150 STRAIB(NEL,I) = EPSD(I)
000       C
000         900 CONTINUE
000       C
000       C        UPDATE DISPL.,VEL., AND ACCELL.
000       C
```

```
000      160 NTIME = NTIME + 1
000          DO 910 I = 1,MACT
000          DF(I) = DISP(I)
000          UDB(I) = DDOT(I)
000      910 UDDB(I) = UDD(I)
000      600 FORMAT(///10X,'NTIME =',I5/2X,'ELEM',T10,'ELASTIC STRESS',T40,
000         1'VISCOUS STRESS',T70,'TOT.STRESS',T100,'PRINCIPAL STRESS'/)
000      610 FORMAT(I5,12E10.3)
000     C
000      620 FORMAT(//10X,'PRINCIPAL STRESS.  NTIME =',I5/)
000      630 FORMAT(I10,3E13.5)
000     C
000          RETURN
000          END
000     @ELT,SIH NASA*TPF$.PSTIFF,,,163602132410
000          SUBROUTINE PSTIFF(NEL,SIGT,IPLST,IRITE)
000          PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000          COMMON /BLKO/ TITLE(20),INODE,NELEM,NAPC,NBC,MACT,ISW(5)
000          COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000         * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
000         * AO,BO,CO,IC,JC,KC,LC,MEL
000          COMMON /BL[illegible]/ D(3,3),ATD(3,3),AD(3,3),BD(3,3),STIFF(8,8),CM(8,8),
000         * VE(8,8),AI(NELS)
000          COMMON /BL[illegible]/ SB(NELS,3),STRAIB(NELS,3),ALPHA,BETA,GAMMA,DELT,XK,
000         * AT,XNU,E,SM,SMS,VOIDI,CAPA,RAMD,BET
000          COMMON /BLKP/ DP(NELS,3,3),SIGB(NELS,3)
000          DIMENSION SIGT(3)
000     C
000     C    COMPRESSION IS POSITIVE
000     C
000          CALL ZERO(STIFF,8,8)
000          READ(2) TYPEA,TYPEB,TYPEC,TYPED
000          IPLST = -1
000          IF(NEL.LE.ISW(5)) GO TO 800
000          XNUS = XNU * XNU
000          VA = XNU + 1.
000          VB = 1.+XNUS-XNU
000          VC = 2.*(XNUS-XNU) - 1.
000          DSIGZ = -SIGT(1)-SIGB(NEL,1)
000          DSIGR = -SIGT(2)-SIGB(NEL,2)
000          DTAUZ = -SIGT(3)-SIGB(NEL,3)
000          SIGZ = -SIGT(1) + DSIGZ/2.
000          SIGR = -SIGT(2) + DSIGR/2.
000          TAUZ = -SIGT(3) + DTAUZ/2.
000          DELP = VA*(DSIGZ+DSIGR) / 3.
000          PP = VA * (SIGZ+SIGR) / 3.
000          PSQ = PP * PP
000          SZZ = 2.*VB*SIGZ + VC*SIGR
000          SRR = 2.*VB*SIGR + VC*SIGZ
000          SZR = 6.*TAUZ
000          TJ = (VB*SIGZ*SIGZ+VB*SIGR*SIGR+VC*SIGZ*SIGR)/3. + TAUZ*TAUZ
000          TTJ = 3.*TJ
000          ETM = 1.+TTJ/(PSQ*SMS)
000          POW = 1.-CAPA/RAMD
000          PC = PP*ETM**POW
000          CALL AREAA(NEL,AREA)
000          RATIO = AREA / AI(NEL)
```

```
000         VOIDR = RATIO * (1.+VOIDI) - 1.
000         VDR = VOIDR + 1.
000         AA = SMS * (2.*PP-PO) / 3.
000         PB = 3.*AA*VDR*SMS*PP*PO / BET
000         AR(1) = SZZ + AA
000         AR(2) = SRR + AA
000         AR(3) = SZR
000         BR(1) = SZZ
000         BR(2) = SRR
000         BR(3) = SZR
000         DFI = 0.
000         DFK = 0.
000         DO 220 I = 1,3
000         DFI = DFI + AR(I) * STRAIB(NEL,I)
000     220 DFK = DFK + BR(I) * STRAIB(NEL,I)
000         DFJ = PO * SMS * DELP
000         POW = -CAPA/RAMD
000         DFK = ETM **POW * (DFK-2.*TTJ/PP*DELP)
000         DF = DFI - DFJ - DFK * (1.-CAPA/RAMD)
000         ASQ = SMS * PP * (PO-PP)
000         ETA = SQRT(TTJ) / PP
000         IF(DF.LT.0.)  GO TO 800
000         IPLST = 1
000         IF(IRITE.EQ.1) WRITE(6,620) VOIDR,TTJ,ASQ,DF,NEL,SM,ETA
000     620 FORMAT(5X,'VOIDR=',F10.5,'  3J=',E12.5,'  ASQ=',E12.5,'  DF=',
000        1 E12.5, I7,'  M=',E12.5,'  ETA=',E12.5)
000         DO 300 I = 1,3
000         BR(I) = 0.
000         CR(I) = 0.
000         DO 300 J = 1,3
000         BR(I) = BR(I) + D(I,J)*AR(J)
000     300 CR(I) = CR(I) + AR(J)*D(J,I)
000         DEN = 0.
000         DO 310 I = 1,3
000     310 DEN = DEN + AR(I)*BR(I)
000         DEN = DEN + BB
000         DO 320 I = 1,3
000         DO 320 J = 1,3
000     320 DP(NEL,I,J) = -BR(I) * CR(J) / DEN
000     600 FORMAT(I5,'  AREA=,  VOIDR,  PO,  DPO,,',4E15.6)
000     610 FORMAT(3E15.6)
000    C
000         DO 100 I = 1,4
000         DO 100 J = 1,4
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)*TYPEB(J,I)
000         STIFF(J+4,I) = STIFF(I,J+4)
000     100 STIFF(I+4,J+4) = DP(NEL,2,2)*TYPEC(I,J)+DP(NEL,3,3)*TYPEA(I,J)
000    C
000     800 RETURN
000         END
000    @ELT,SIH NASA*TPF$.FACTOR,,,226224130610
000         SUBROUTINE FACTOR(XK,IHB,IHBI,LT,LAST,NFREE)
000    C    THIS SUBROUTINE PERFORMS FACTORING
000         DIMENSION XK(1)
000         N = NFREE
000         IHB1 = IHBI
```

```
000         DO 8 I=1,N
000         IF(I.GT.IHB1) GO TO 2
000         K=1
000         M=K+(I-1)*I/2
000         GO TO 3
000       2 K=I-IHB1
000         M=K+LT+(I-IHB)*IHB1
000       3 J=I+IHB1
000         IF(J.GT.N) GO TO 4
000         JJ=I+IHB1
000         GO TO 5
000       4 JJ=N
000       5 B=0.0
000         LA=I-1
000         LB=I+1
000         IF(LA.EQ.0) GO TO 6
000         DO 7 L=K,LA
000         IF(L.GT.IHB1)GO TO 50
000          J = (L+1)*L/2
000         GO TO 51
000      50 J = LT + IHB*(L-IHB1)
000      51 A = XK(M)
000          B = B+A*A*XK(J)
000      7  M=M+1
000       6 A=XK(M)
000         XK(M)=A-B
000         IF(I.EQ.N) GO TO 8
000         DO 9 J=LB,JJ
000         SUM=0.0
000         IF(J.GT.IHB1) GO TO 10
000         K=1
000         MM=K+(J-1)*J/2
000         GO TO 11
000      10 K=J-IHB1
000         MM=K+LT+(J-IHB)*IHB1
000      11 IF(LA.EQ.0) GO TO 9
000         IF(K.GT.LA) GO TO 9
000         DO 12 JA=K,LA
000         L=M-I+JA
000         IF(JA.GT.IHB1) GO TO 13
000         L1=(JA+1)*JA/2
000         GO TO 14
000      13 L1=LT+IHB*(JA-IHB1)
000      14 SUM=SUM+XK(MM)*XK(L)*XK(L1)
000      12 MM=MM+1
000       9 XK(MM)=(XK(MM)-SUM)/XK(M)
000       8 CONTINUE
000         RETURN
000         END
000    @ELT,SIH NASA*IPF$.SOLTN,,,226226130610
000         SUBROUTINE SOLTN(XK,IHB,IHBI,LT,LAST,NFREE)
000    C    THIS PORTION OF SUBROUTINE PERFORMS FORWARD-SUBSTITUTION
000         DIMENSION XK(1)
000    C
000         N = NFREE
000         IHB1 = IHBI
000         NF = LAST + 1
```

```
000    C
000     14 DO 1 K = 2,N
000    C
000          IF(K.GT.IHB1) GO TO 2
000          M=0
000          MM=K-1
000          M1=MM*K/2
000          GO TO 3
000        2 M=K-IHB
000          MM=IHB1
000          M1=M*IHB1+LT
000        3 SUM=0.0
000          DO 4 L=1,MM
000          LL=L+M
000          JJ=LL+M1
000          LL=LL+NF-1
000        4 SUM=SUM+XK(JJ)*XK(LL)
000        1 XK(LL+1)=XK(LL+1)-SUM
000          J = NF+N-1
000    C     THIS PORTION OF SUBROUTINE PERFORMS BACK-SUBSTITUTION
000          NF=NF+N-1
000          XK(NF)=XK(NF)/XK(LAST)
000          DO 5 K=2,N
000          L=N-K+1
000          IF(L.GT.IHB1) GO TO 6
000          I=L+(L-1)*L/2
000          GO TO 7
000        6 I=L+(L-IHB)*IHB1+LT
000       7  IR=N-IHB
000          IF(L.GT.IR) GO TO 8
000          J=IHB1
000          GO TO 9
000        8 J=K-1
000        9 SUM=0.0
000          DO 10 M=1,J
000          MM=L+M
000          IF(MM.GT.IHB1) GO TO 11
000          NN=L+(MM-1)*MM/2
000          GO TO 12
000       11 NN=L+(MM-IHB)*IHB1+LT
000       12 MM=NF-N+MM
000       10 SUM=SUM+XK(NN)*XK(MM)
000          MM=NF-N+L
000        5 XK(MM)=XK(MM)/XK(I)-SUM
000          RETURN
000          END
000    @ELT,SIH NASA*TPF$.SETUP,,,226231130610
000          SUBROUTINE SETUP
000          COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
000         * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
000         * AO,BO,CO,IC,JC,KC,LC,NEL
000          W(1) = 0.3478548
000          W(2) = 0.6521452
000          W(3) = W(2)
000          W(4) = W(1)
000          H(1) = 0.8611363
000          H(2) = 0.3399810
```

```
      H(3) = -H(2)
      H(4) = -H(1)
      BN(1) = -1.
      BN(2) = 1.
      BN(3) = 1.
      BN(4) = -1.
      CN(1) = -1.
      CN(2) = -1.
      CN(3) = 1.
      CN(4) = 1.
      DN(1) = 1.
      DN(2) = -1.
      DN(3) = 1.
      DN(4) = -1.
      TWOPI = 1.
      RETURN
      END
@ELT,SIH NASA*IPF$.GAUSS,,,226232130610
      SUBROUTINE GAUSS (K,AA)
      COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
     * A0,B0,C0,IC,JC,KC,LC,NEL
      IPT = 4
       AA = 0.
      DO 100 I = 1,IPT
       X = H(I)
      DO 100 J = 1,IPT
       Y = H(J)
      AA = AA + W(I) * W(J) * F(K,X,Y)
  100 CONTINUE
      RETURN
      END
      FUNCTION F(K,X,Y)
      COMMON /BLK1/ W(4),H(4),AR(4),BR(4),CR(4),AZ(4),BZ(4),CZ(4),
     * BN(4),CN(4),DN(4),TYPEA(4,4),TYPEB(4,4),TYPEC(4,4),TYPED(4,4),
     * A0,B0,C0,IC,JC,KC,LC,NEL
      COMMON /BLK9/ M,N
      FC = A0 + B0*X + C0*Y
C
      GO  TO (10,20,30,40,50,60), K
   10 CONTINUE
      AMAN = AR(M)*AR(N)
      BMAN = BR(M)*AR(N) + BR(N)*AR(M)
      AMCN = AR(M)*CR(N) + AR(N)*CR(M)
      BMBN = BR(M)*BR(N)
      BMCN = BR(M)*CR(N) + CR(M)*BR(N)
      CMCN = CR(M)*CR(N)
      GO TO 100
   20 CONTINUE
      AMAN = AR(M)*AZ(N)
      BMAN = BZ(N)*AR(M) + BR(M)*AZ(N)
      AMCN = AR(M)*CZ(N) + CR(M)*AZ(N)
      BMBN = BR(M)*BZ(N)
      BMCN = BR(M) * CZ(N) + CR(M) * BZ(N)
      CMCN = CZ(N)*CR(M)
      GO TO 100
   30 CONTINUE
```

```
000         AMAN = AZ(M)*AZ(N)
000         BMAN = BZ(M)*AZ(N) + BZ(N)*AZ(M)
000         AMCN = AZ(M)*CZ(N) + AZ(N)*CZ(M)
000         BMBN = BZ(M)*BZ(N)
000         BMCN = BZ(M)*CZ(N) + CZ(M)*BZ(N)
000         CMCN = CZ(M)*CZ(N)
000         GO TO 100
000      40 CONTINUE
000         XY =  X * Y
000         FA = 1. + BN(M)*X + CN(M)*Y + DN(M)*XY
000         FB = 1. + BN(N)*X + CN(N)*Y + DN(N)*XY
000         F = FA*FB*FC/128.
000         RETURN
000      50 F = (AR(M) + BR(M)*X + CR(M)*Y) / FC
000         RETURN
000      60 F = (AZ(M) + BZ(M)*X + CZ(M)*Y) /FC
000         RETURN
000     100 FA = AMAN + BMAN*X + AMCN*Y + BMBN*X*X + BMCN*X*Y + CMCN*Y*Y
000         F = FA/FC*.125
000         RETURN
000         END
000   @ELT,SIH NASA*TPF$.DIAGNL,,,226237130610
000         SUBROUTINE DIAGNL(XK,APF,NFREE,IHB,IHBI,LT,LAST)
000         DIMENSION XK(1),APF(1)
000   C
000         DO 200 I = 1,LAST
000     200 XK(I) = 0.
000   C
000         DO 100 J = 1,NFREE
000         IF(J.GT.IHBI) GO TO 108
000         L = (J+1) * J / 2
000         GO TO 109
000     108 L = LT + IHB * (J - IHBI)
000     109 XK(L) = 1.
000     100 CONTINUE
000   C
000         DO 110 I = 1,NFREE
000     110 XK(LAST+I) = APF(I)
000   C
000         RETURN
000         END
000   @ELT,SIH NASA*TPF$.AREAA,,,163614132410
000         SUBROUTINE AREAA(NEL,AREA)
000         PARAMETER NFT=150,NELS=150,MX= 5000,NF=NFT*2
000         COMMON /BLK2/ ID(NF,2),IJKL(NELS,4),R(NFT),Z(NFT),KK(NELS,8)
000         COMMON /BLK7/ UDB(NF),UDDB(NF),FOV(NF),UDD(NF),DB(NF),DISP(NF),
000        * DDOT(NF)
000         DIMENSION RR(4),ZZ(4)
000         DO 100 I = 1,4
000         IN = IJKL(NEL,I)
000         JJ = IN * 2
000         II = JJ - 1
000         RR(I) = R(IN) + DISP(JJ)
000     100 ZZ(I) = Z(IN) + DISP(II)
000         AI = (RR(2)-RR(1))*(ZZ(4)-ZZ(1))-(RR(4)-RR(1))*(ZZ(2)-ZZ(1))
000         AJ = (RR(3)-RR(2))*(ZZ(4)-ZZ(2))-(RR(4)-RR(2))*(ZZ(3)-ZZ(2))
000         IF(AI.LT.0) AI = -AI
```

```
000          IF(AJ.LT.0) AJ = -AJ                                                                                                 00001600
000          AREA = (AI + AJ) / 2.                                                                                                00001700
000          RETURN                                                                                                               00001800
000          END                                                                                                                  00001900
000   @ELT,SIH NASA*TPF$.ZERO,,,226242130610
000          SUBROUTINE ZERO(A,N,M)                                                                                               00000100
000          DIMENSION A(1)                                                                                                       00000200
000          NM = N * M                                                                                                           00000300
000          DO 100 I = 1,NM                                                                                                      00000400
000      100 A(I) = 0.                                                                                                            00000500
000          RETURN                                                                                                               00000600
000          END                                                                                                                  00000700
000   @XQT
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
```

```
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    38    33
000        32    39    44    43    38
000        33    44    49    48    43
000        34    49    54    53    48
000        35    54    59    58    53
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    -26.6736
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 = $T_{(r)}$ x 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) VOlDI - 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

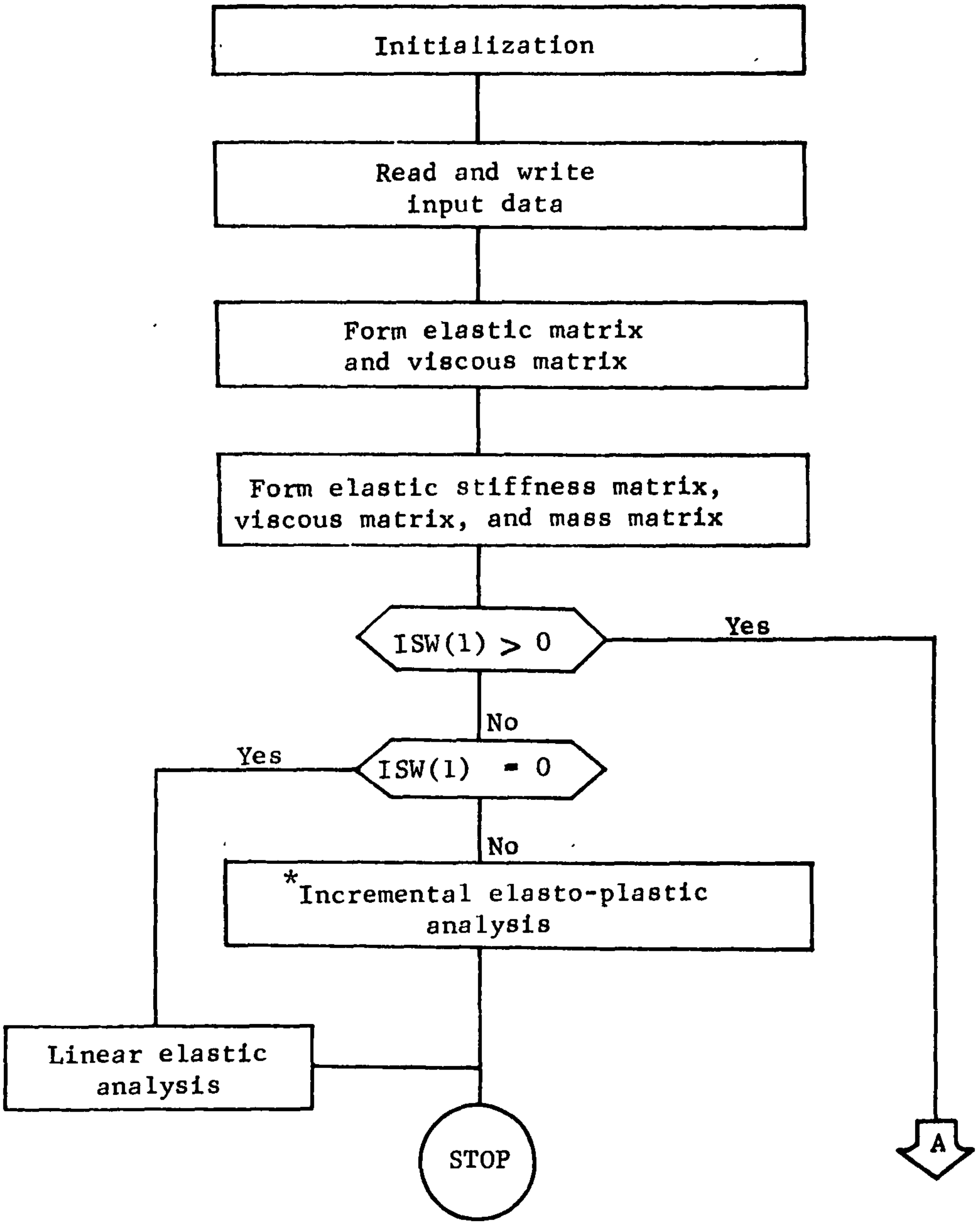
(3) PR - R-component

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

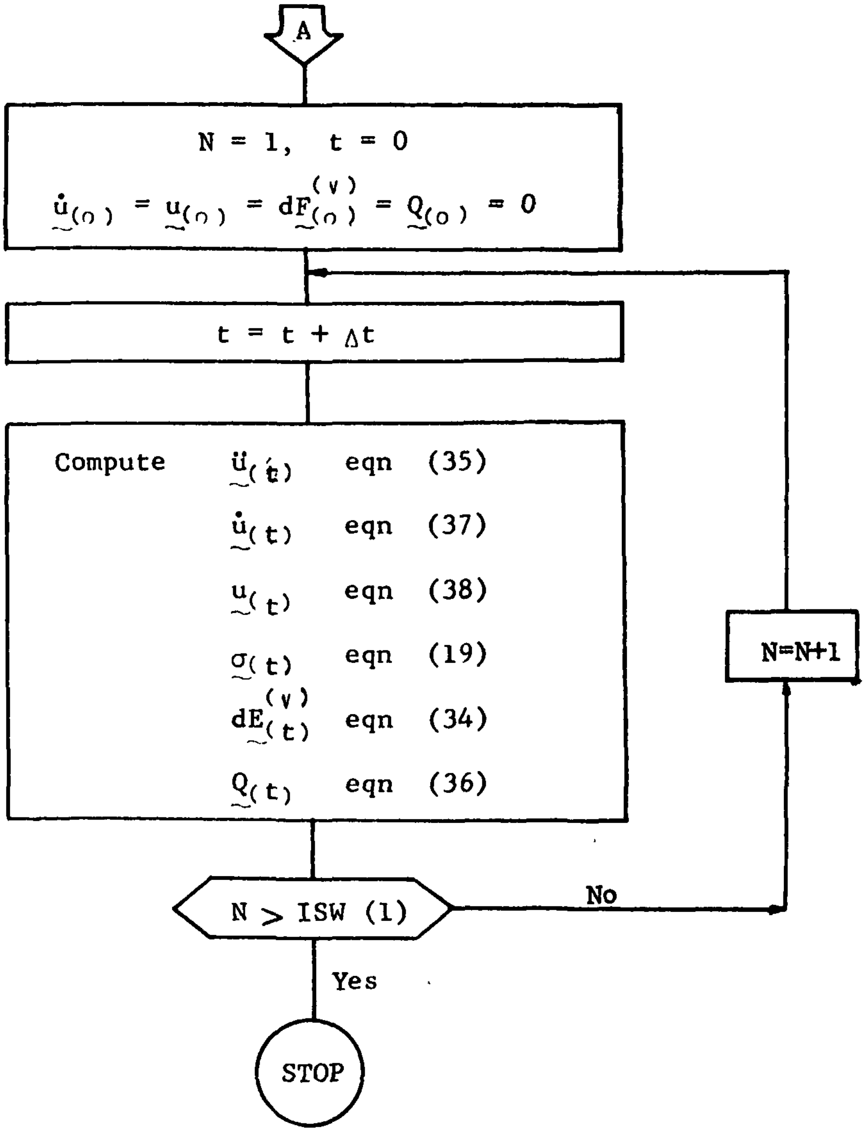
Repeat NAPC times.

APPENDIX 5

FLOW CHART

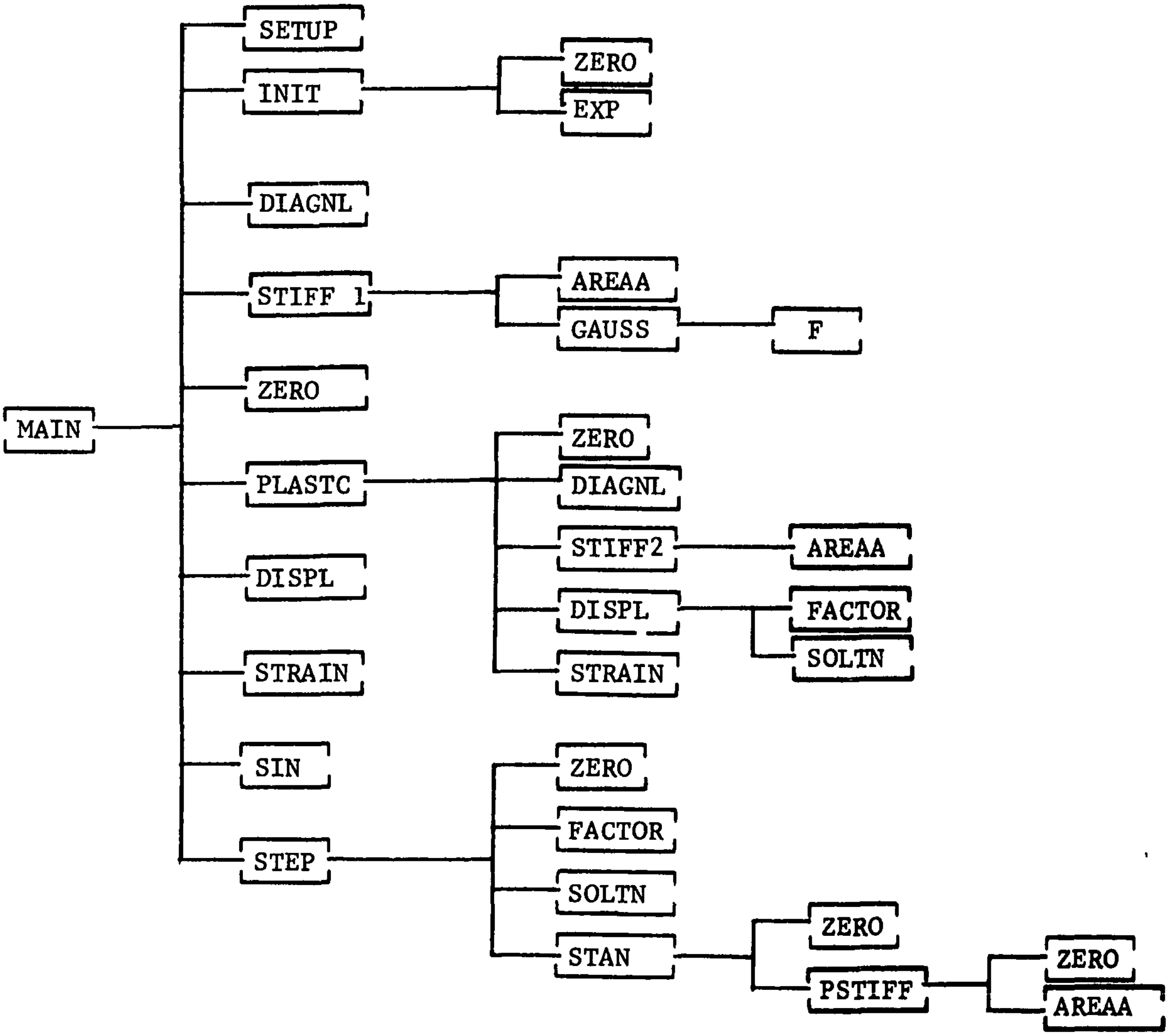

Initialization
Read and write input data
Form elastic matrix and viscous matrix
Form elastic stiffness matrix, viscous matrix, and mass matrix
ISW(1) > 0
Yes
No
ISW(1) = 0
Yes
No
*Incremental elasto-plastic analysis
Linear elastic analysis
STOP
A

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


A
$N = 1, \quad t = 0$
$\dot{\underset{\sim}{u}}_{(o)} = \underset{\sim}{u}_{(o)} = d\underset{\sim}{F}^{(v)}_{(o)} = \underset{\sim}{Q}_{(o)} = 0$
$t = t + \Delta t$
Compute $\ddot{\underset{\sim}{u}}_{(t)}$ eqn (35)
$\dot{\underset{\sim}{u}}_{(t)}$ eqn (37)
$\underset{\sim}{u}_{(t)}$ eqn (38)
$\underset{\sim}{\sigma}_{(t)}$ eqn (19)
$d\underset{\sim}{E}^{(v)}_{(t)}$ eqn (34)
$\underset{\sim}{Q}_{(t)}$ eqn (36)
$N > ISW\ (1)$
No
Yes
N=N+1
STOP

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